

BECKHOFF New Automation Technology

Manual | EN

TC1000

TwinCAT 3 ADS .NET V6

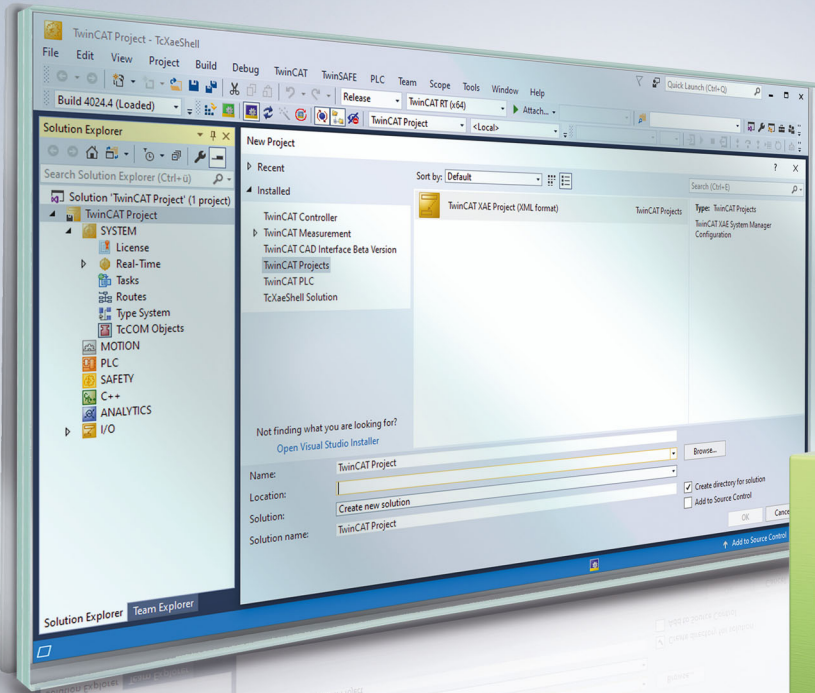


Table of contents

1	Foreword	13
1.1	Notes on the documentation	13
1.2	For your safety	13
1.3	Notes on information security.....	15
2	TwinCAT ADS .NET API Documentation	16
2.1	Prerequisites	17
2.2	Installation	17
3	Version History	19
3.1	Version 6.2	20
3.2	Version 6.1	20
3.3	Version 6.0	21
3.4	Version 5.X.....	21
3.5	Version 4.X.....	22
4	Concepts	23
4.1	Access Data via IndexGroup/IndexOffset	23
4.2	Async programming (async, await)	24
4.3	Use of ADS Notifications	26
4.4	Access Data via Symbolic path	29
4.5	Access Data via Symbol handles	30
4.6	Value marshalling with ANYTYPE concept.....	31
4.7	Access Data via Symbol Loader	35
4.8	Automatic dynamic marshalling of values	36
4.9	Reactive Extensions and ADS	38
5	HowTo Samples	42
5.1	Read/Write primitive values	42
5.2	Read/Write string types	46
5.3	Read/Write PlcOpen types (DATE, TIME ...)	49
5.4	Event driven read with ADS Notifications.....	52
5.5	Reactive Read/Write with Reactive Extensions	54
5.6	Upgrading existing ADS Application code (Version 4.X --> 6.X)	56
6	TwinCAT.Ads Namespaces	59
6.1	TwinCAT Namespace	59
6.1.1	AdsException Class	61
6.1.2	ClientNotConnectedException Class	67
6.1.3	ConnectionState Enumeration	72
6.1.4	ConnectionStateChangedEventArgs Class	74
6.1.5	ConnectionStateChangedReason Enumeration	79
6.1.6	IConnection Interface	79
6.1.7	IConnectionStateObserver Interface.....	85
6.1.8	IConnectionStateProvider Interface	89
6.1.9	ISession Interface	92
6.1.10	ISessionProvider Interface	99
6.1.11	ISessionProvider.TSession, TAddress, TSettings. Interface	102

6.1.12	ISymbolLoaderSettings Interface	104
6.1.13	ISymbolServerProvider Interface	106
6.1.14	Session Class	107
6.1.15	SessionConnectionStateChangedEventArgs Class.....	125
6.1.16	SessionException Class.....	130
6.1.17	SessionNotConnectedException Class.....	137
6.1.18	SessionProvider.TSession, TAddress, TSettings. Class	142
6.1.19	SessionProviderCapabilities Enumeration	148
6.1.20	SymbolLoaderSettings Class	148
6.1.21	SymbolsLoadMode Enumeration	159
6.1.22	ValueUpdateMode Enumeration	159
6.1.23	AdsCommunicationStatistics Class.....	160
6.1.24	RouteAccessType Enumeration.....	169
6.1.25	RoutePersistenceType Enumeration	170
6.1.26	HandleBagNotInitializedException Class	171
6.1.27	IAdsClientSettings Interface.....	175
6.1.28	IValueFactorySettings Interface	176
6.2	TwinCAT.Ads Namespace	179
6.2.1	AdsClient Class.....	183
6.2.2	AdsClientSettings Class.....	399
6.2.3	AdsCommandId Enumeration	404
6.2.4	AdsCommunicationStatistics Class.....	404
6.2.5	AdsConnection Class	412
6.2.6	AdsDataTypeArrayInfo Class.....	660
6.2.7	AdsDataTypeId Enumeration	663
6.2.8	AdsErrorCode Enumeration	664
6.2.9	AdsErrorCodeExtensions Class.....	680
6.2.10	AdsErrorException Class	682
6.2.11	AdsInvalidNotificationException Class	690
6.2.12	AdsNotificationErrorEventArgs Class.....	694
6.2.13	AdsNotificationEventArgs Class.....	695
6.2.14	AdsNotificationExEventArgs Class	699
6.2.15	AdsSession Class	701
6.2.16	AdsSessionBase Class	716
6.2.17	AdsState Enumeration	729
6.2.18	AdsStateChangedEventArgs Class	729
6.2.19	AdsStateChangedEventArgs2 Class	732
6.2.20	AdsSumCommandException Class	735
6.2.21	AdsSymbolVersionChangedEventArgs Class.....	739
6.2.22	AdsTransMode Enumeration	740
6.2.23	AdsVersion Class.....	743
6.2.24	AmsAddress Class.....	752
6.2.25	AmsNetId Class	767
6.2.26	AmsPort Enumeration	795
6.2.27	AmsRouterNotificationEventArgs Class.....	798
6.2.28	AmsRouterState Enumeration	800

6.2.29	DeviceInfo Class	801
6.2.30	IAdsAnyAccess Interface	804
6.2.31	IAdsConnectAddress Interface	835
6.2.32	IAdsConnection Interface.....	876
6.2.33	IAdsDisposableConnection Interface	915
6.2.34	IAdsHandle Interface.....	954
6.2.35	IAdsNotifications Interface	966
6.2.36	IAdsReadWrite Interface	995
6.2.37	IAdsReadWrite2 Interface	1000
6.2.38	IAdsReadWriteTimeoutAccess Interface	1005
6.2.39	IAdsRpcInvoke Interface	1012
6.2.40	IAdsSession Interface	1033
6.2.41	IAdsSessionSettings Interface	1037
6.2.42	IAdsStateControl Interface	1039
6.2.43	IAdsStateControlTimeout Interface	1048
6.2.44	IAdsStateObserver Interface.....	1054
6.2.45	IAdsStateProvider Interface	1056
6.2.46	IAdsSymbolChangedProvider Interface	1061
6.2.47	IAdsSymbolicAccess Interface.....	1065
6.2.48	IAdsSymbolTableProvider Interface.....	1093
6.2.49	INotification Interface	1096
6.2.50	INotificationSettings Interface	1098
6.2.51	IRouterNotificationProvider Interface	1099
6.2.52	Notification Class	1100
6.2.53	NotificationSettings Class	1106
6.2.54	ResultAds Class.....	1116
6.2.55	ResultAnyValue Class.....	1129
6.2.56	ResultDeviceInfo Class	1132
6.2.57	ResultHandle Class.....	1136
6.2.58	ResultRead Class	1143
6.2.59	ResultReadAdsState Class.....	1146
6.2.60	ResultReadBytes Class	1150
6.2.61	ResultReadDeviceState Class	1157
6.2.62	ResultReadWrite Class	1163
6.2.63	ResultReadWriteBytes Class	1170
6.2.64	ResultRpcMethod Class.....	1176
6.2.65	ResultValue.TValue. Class	1181
6.2.66	ResultWrite Class.....	1187
6.2.67	SessionSettings Class	1193
6.2.68	StateInfo Structure	1200
6.2.69	TaskExtensions Class.....	1209
6.2.70	TransportProtocols Enumeration	1212
6.2.71	ValueNotificationEventArgs.T. Class	1213
6.2.72	AdsClientExtensions Class	1215
6.2.73	AdsNotificationsInvalidatedEventArgs Class	1228
6.2.74	IndexGroupSymbolAccess Enumeration	1231

6.2.75	AdsStateCommand Enumeration.....	1233
6.2.76	AmsPortRange Enumeration	1233
6.2.77	ResultValue2.I, V. Class	1234
6.2.78	ResultWriteControl Class	1236
6.3	TwinCAT.Ads.Reactive Namespace	1249
6.3.1	AdsClientExtensions Class	1249
6.3.2	AnyTypeExtensions Class	1281
6.3.3	SymbolValueNotification Class	1331
6.3.4	ValueSymbolExtensions Class	1333
6.4	TwinCAT.Ads.Server Namespace	1374
6.4.1	AdsServer Class	1375
6.4.2	AdsServerException Class.....	1443
6.4.3	ErrorEventArgs Class.....	1448
6.4.4	LoopbackNotRegisteredException Class.....	1450
6.4.5	NotificationDataSample Class.....	1453
6.4.6	NotificationSamplesStamp Class	1456
6.4.7	ServerConnectionState Enumeration.....	1461
6.4.8	ServerConnectionStateChangedEventArgs Class.....	1462
6.4.9	ServerNotConnectedException Class.....	1465
6.4.10	AdsSymbolicServer Class.....	1468
6.4.11	BaseTickTrigger Class	1513
6.4.12	IAddNotificationTrigger Interface.....	1515
6.4.13	INotificationTrigger Interface	1516
6.4.14	NotificationTriggerSource Class.....	1517
6.4.15	LoopbackNotConnectedException Class.....	1523
6.5	TwinCAT.Ads.SumCommand Namespace	1525
6.5.1	ISumCommand Interface	1527
6.5.2	ResultSumCommand Class	1533
6.5.3	ResultSumHandles Class	1538
6.5.4	ResultSumHandles2 Class	1540
6.5.5	ResultSumReadRaw Class.....	1544
6.5.6	ResultSumValues Class.....	1547
6.5.7	SumCreateHandles Class.....	1550
6.5.8	SumHandleRead Class.....	1557
6.5.9	SumHandleWrite Class	1561
6.5.10	SumReleaseHandles Class	1565
6.5.11	SumSymbolRead Class	1572
6.5.12	SumSymbolWrite Class	1582
6.5.13	ISumSymbolRead Interface	1587
6.5.14	ISumCommandExtension Class	1591
6.5.15	ISumRead Interface	1596
6.5.16	ISumRead2.S. Interface.....	1601
6.5.17	ResultSumValues2.S. Class	1605
6.5.18	SumAccessMode Enumeration.....	1608
6.5.19	SumAnyTypeRead Class	1608
6.5.20	SumCommandErrorStrategy Enumeration	1614

6.5.21	SumCommandMode Enumeration.....	1615
6.5.22	SumInstancePathAnyTypeRead Class.....	1615
6.6	TwinCAT.Ads.TcpRouter Namespace.....	1619
6.6.1	AmsTcpIpRouter Class.....	1620
6.6.2	IAmsRouter Interface.....	1646
6.6.3	Route Class.....	1654
6.6.4	RouteCollection Class.....	1664
6.6.5	RouterException Class.....	1678
6.6.6	RouterNotInitializedException Class.....	1682
6.6.7	RouterNotStartedException Class.....	1684
6.6.8	RouterStatus Enumeration.....	1687
6.6.9	RouterStatusChangedEventArgs Class.....	1688
6.6.10	StaticRoutesXmlConfigurationBuilderExtension Class.....	1690
6.6.11	StaticRoutesXmlConfigurationProvider Class.....	1691
6.6.12	StaticRoutesXmlConfigurationSource Class.....	1696
6.7	TwinCAT.Ads.TypeSystem Namespace.....	1698
6.7.1	AliasType Class.....	1699
6.7.2	ArrayType Class.....	1707
6.7.3	BitMappingType Class.....	1718
6.7.4	DataType Class.....	1721
6.7.5	EnumType.T. Class.....	1738
6.7.6	Field Class.....	1750
6.7.7	IAdsSymbol Interface.....	1755
6.7.8	IAdsSymbolLoader Interface.....	1760
6.7.9	IContextMaskProvider Interface.....	1764
6.7.10	Instance Class.....	1765
6.7.11	Member Class.....	1781
6.7.12	PCCHType Class.....	1787
6.7.13	PointerType Class.....	1790
6.7.14	PrimitiveType Class.....	1796
6.7.15	PVoidType Class.....	1801
6.7.16	ReferenceType Class.....	1804
6.7.17	RpcMethod Class.....	1813
6.7.18	RpcMethodParameter Class.....	1824
6.7.19	RpcStructType Class.....	1835
6.7.20	StringType Class.....	1840
6.7.21	StructType Class.....	1846
6.7.22	SubRangeType.T. Class.....	1856
6.7.23	Symbol Class.....	1863
6.7.24	SymbolIterator Class.....	1916
6.7.25	SymbolLoaderFactory Class.....	1923
6.7.26	UnionType Class.....	1933
6.7.27	WStringType Class.....	1938
6.7.28	AlignmentCalculator Class.....	1944
6.7.29	FluentRpcMethodExtension Class.....	1947
6.7.30	FluentRpcStructTypeExtension Class.....	1952

6.7.31	FluentStructTypeExtension Class	1955
6.7.32	InterfaceType Class	1957
6.7.33	RpcStructInstance Class	1966
6.7.34	UnionInstance Class	1992
6.8	TwinCAT.Ads.ValueAccess Namespace	2002
6.8.1	ValueAccessMode Enumeration	2002
6.9	TwinCAT.Ams Namespace	2003
6.9.1	AmsConfiguration Class.....	2004
6.9.2	AmsServerErrorCode Enumeration	2008
6.9.3	AmsServerException Class.....	2009
6.9.4	IAdsHeaderDump Interface.....	2012
6.9.5	AmsPortNotAvailableException Class	2013
6.10	TwinCAT.PlcOpen Namespace	2016
6.10.1	DATE Class.....	2016
6.10.2	DateBase Class	2023
6.10.3	DT Class	2034
6.10.4	IPlcOpenTimeBase Interface	2041
6.10.5	IPlcOpenTimeBase.T1, T2. Interface.....	2043
6.10.6	LTIME Class.....	2045
6.10.7	LTimeBase Class.....	2053
6.10.8	TIME Class.....	2061
6.10.9	TimeBase Class	2068
6.10.10	TOD Class.....	2077
6.11	TwinCAT.TypeSystem Namespace	2083
6.11.1	AnySymbolSpecifier Class	2091
6.11.2	AnyTypeSpecifier Class	2094
6.11.3	CannotAccessVirtualSymbolException Class	2103
6.11.4	CannotResolveDataTypeException Class	2106
6.11.5	DataTypeCategory Enumeration.....	2111
6.11.6	DataTypeCollection Class.....	2112
6.11.7	DataTypeEventArgs Class	2118
6.11.8	DataTypeException Class.....	2121
6.11.9	DataTypeNameEventArgs Class	2127
6.11.10	Dimension Class	2130
6.11.11	DimensionCollection Class	2133
6.11.12	DynamicAliasInstance Class.....	2149
6.11.13	DynamicArrayInstance Class	2160
6.11.14	DynamicOversamplingArrayInstance Class.....	2176
6.11.15	DynamicPointerInstance Class	2186
6.11.16	DynamicPointerValue Class.....	2196
6.11.17	DynamicReferenceInstance Class	2202
6.11.18	DynamicReferenceValue Class	2217
6.11.19	DynamicRpcStructInstance Class.....	2223
6.11.20	DynamicStructInstance Class	2251
6.11.21	DynamicSymbol Class	2266
6.11.22	DynamicSymbolsCollection Class.....	2327

6.11.23	DynamicUnionInstance Class	2336
6.11.24	DynamicValue Class	2347
6.11.25	DynamicVirtualStructInstance Class	2374
6.11.26	EnumValue.T. Class	2384
6.11.27	EnumValueCollection Class.....	2391
6.11.28	EnumValueCollection.T. Class.....	2407
6.11.29	FieldCollection Class.....	2427
6.11.30	IAliasInstance Interface.....	2433
6.11.31	IAliasType Interface	2436
6.11.32	IAnyTypeMarshaler Interface	2441
6.11.33	IArrayInstance Interface	2453
6.11.34	IArrayType Interface.....	2459
6.11.35	IArrayValue Interface	2465
6.11.36	IAttributedInstance Interface	2469
6.11.37	IBitSize Interface	2472
6.11.38	IDataType Interface.....	2475
6.11.39	IDataTypeCollection Interface.....	2484
6.11.40	IDataTypeCollection.T. Interface.....	2486
6.11.41	IDimension Interface	2490
6.11.42	IDimensionCollection Interface	2492
6.11.43	IDynamicSymbol Interface	2496
6.11.44	IDynamicSymbolLoader Interface.....	2500
6.11.45	IDynamicSymbolsCollection Interface.....	2503
6.11.46	IDynamicValue Interface	2504
6.11.47	IEnumType Interface.....	2507
6.11.48	IEnumType.T. Interface.....	2516
6.11.49	IEnumValue Interface.....	2524
6.11.50	IEnumValueCollection Interface.....	2527
6.11.51	IEnumValueCollection.TEnumValue, TValue. Interface.....	2529
6.11.52	IField Interface	2535
6.11.53	IFieldCollection Interface.....	2538
6.11.54	IGenericTypeMarshaler Interface.....	2542
6.11.55	IHierarchicalSymbol Interface	2545
6.11.56	IInstance Interface.....	2549
6.11.57	IInstanceCollection.T. Interface.....	2554
6.11.58	IMember Interface	2561
6.11.59	IMemberCollection Interface	2565
6.11.60	INamespaceCollection Interface	2570
6.11.61	INamespaceCollection.T. Interface	2571
6.11.62	InstanceCollectionMode Enumeration	2573
6.11.63	InsufficientAccessRightsException Class	2573
6.11.64	IOversamplingArrayInstance Interface.....	2576
6.11.65	IPointerInstance Interface	2580
6.11.66	IPointerType Interface.....	2584
6.11.67	IPrimitiveType Interface	2588
6.11.68	IProcessImageAddress Interface.....	2593

6.11.69	IReferenceInstance Interface	2595
6.11.70	IReferenceType Interface.....	2600
6.11.71	IRpcCallableInstance Interface	2606
6.11.72	IRpcCallableType Interface.....	2624
6.11.73	IRpcMethod Interface.....	2625
6.11.74	IRpcMethodCollection Interface	2629
6.11.75	IRpcMethodParameter Interface.....	2635
6.11.76	IRpcMethodParameterCollection Interface	2639
6.11.77	IRpcStructInstance Interface.....	2642
6.11.78	IStringInstance Interface	2646
6.11.79	IStringMarshaler Interface.....	2650
6.11.80	IStringType Interface.....	2661
6.11.81	IStructInstance Interface	2666
6.11.82	IStructType Interface.....	2671
6.11.83	IStructValue Interface.....	2678
6.11.84	ISubRangeType Interface	2681
6.11.85	ISubRangeType.T. Interface.....	2686
6.11.86	ISymbol Interface	2691
6.11.87	ISymbolCollection Interface	2697
6.11.88	ISymbolCollection.T. Interface	2700
6.11.89	ISymbolFactory Interface	2702
6.11.90	ISymbolFactoryServiceProvider Interface	2711
6.11.91	ISymbolInfo Interface	2712
6.11.92	ISymbolLoader Interface.....	2714
6.11.93	ISymbolProvider Interface.....	2717
6.11.94	ISymbolServer Interface.....	2719
6.11.95	ITypeAttribute Interface.....	2725
6.11.96	ITypeAttributeCollection Interface	2726
6.11.97	ITypeMarshaler Interface	2731
6.11.98	IUnionInstance Interface	2737
6.11.99	IUnionType Interface.....	2740
6.11.100	IValue Interface.....	2745
6.11.101	IValueAccessorProvider Interface.....	2752
6.11.102	IValueAnySymbol Interface.....	2753
6.11.103	IValueRawSymbol Interface.....	2765
6.11.104	IValueSymbol Interface.....	2775
6.11.105	IVirtualStructInstance Interface	2790
6.11.106	MarshalException Class.....	2795
6.11.107	MemberCollection Class	2802
6.11.108	MethodParamFlags Enumeration	2812
6.11.109	PrimitiveTypeFlags Enumeration	2812
6.11.110	RawValueChangedEventArgs Class.....	2813
6.11.111	ReadOnlyDataValueCollection Class.....	2815
6.11.112	ReadOnlyDimensionCollection Class	2818
6.11.113	ReadOnlyEnumValueCollection Class.....	2824
6.11.114	ReadOnlyEnumValueCollection.T. Class.....	2833

6.11.115	ReadOnlyFieldCollection Class.....	2841
6.11.116	ReadOnlyMemberCollection Class	2845
6.11.117	ReadOnlyMethodParameterCollection Class.....	2852
6.11.118	ReadOnlyRpcMethodCollection Class	2855
6.11.119	ReadOnlySymbolCollection Class	2861
6.11.120	ReadOnlyTypeAttributeCollection Class	2866
6.11.121	ResultDataTypes Class.....	2872
6.11.122	ResultDynamicSymbols Class	2876
6.11.123	ResultSymbols Class	2880
6.11.124	ResultSymbols.T. Class	2884
6.11.125	RpcInvokeException Class	2887
6.11.126	RpcMethodCollection Class	2894
6.11.127	RpcMethodNotSupportedException Class.....	2910
6.11.128	RpcMethodParameterCollection Class	2915
6.11.129	StringConvertMode Enumeration.....	2927
6.11.130	SymbolAccessRights Enumeration	2928
6.11.131	SymbolCollection Class	2928
6.11.132	SymbolException Class	2933
6.11.133	TypeAttribute Class.....	2946
6.11.134	TypeAttributeCollection Class	2952
6.11.135	ValueChangedBaseEventArgs Class.....	2968
6.11.136	ValueChangedEventArgs Class.....	2971
6.11.137	AlignedMemberCollection Class	2973
6.11.138	FluentAlignedMemberCollectionExtension Class	2980
6.11.139	IDimensionCollectionExtension Class.....	2981
6.11.140	IInterfaceType Interface	2983
6.11.141	IRpcStructType Interface	2990
6.11.142	DynamicInterfaceInstance Class.....	2995
6.11.143	IInterfaceInstance Interface.....	3014
6.11.144	SymbolIterationMask Enumeration	3019
6.11.145	AnyTypeValueCreator Class.....	3020
6.11.146	DataTypeExtension Class.....	3021
6.11.147	DataTypeResolveStrategy Enumeration.....	3042
6.11.148	DynamicArrayValue Class	3042
6.11.149	DynamicEnumValue Class.....	3048
6.11.150	UInt32Ptr Structure	3054
6.11.151	UInt64Ptr Structure	3070
6.12	TwinCAT.TypeSystem.Generic Namespace.....	3086
6.12.1	DataTypeCollection.T. Class.....	3087
6.12.2	INamespace.TType. Interface.....	3102
6.12.3	InstanceCollection.T. Class.....	3104
6.12.4	ISymbolProvider.TNamespace, TDataType, TSymbol. Interface	3122
6.12.5	NamespaceCollection.T. Class	3126
6.12.6	ReadOnlyDataTypeCollection.T. Class.....	3139
6.12.7	ReadOnlyInstanceCollection.T. Class.....	3145
6.12.8	ReadOnlyNamespaceCollection.T. Class	3154

6.12.9	ReadOnlySymbolCollection.T. Class	3161
6.12.10	SymbolCollection.T. Class	3165
6.12.11	SymbolIterationMask Enumeration	3171
6.12.12	SymbolIterator.T. Class.....	3172
6.13	TwinCAT.ValueAccess Namespace	3179
6.13.1	CannotAccessException Class	3181
6.13.2	IAccessorRawValue Interface	3184
6.13.3	IAccessorValueFactory Interface	3192
6.13.4	ResultAccess Class	3197
6.13.5	ResultReadDynamicValueAccess Class.....	3204
6.13.6	ResultReadRawAccess Class.....	3206
6.13.7	ResultReadValueAccess Class.....	3209
6.13.8	ResultReadValueAccess.T. Class	3211
6.13.9	ResultRpcMethodAccess Class	3213
6.13.10	ResultWriteAccess Class	3217
6.13.11	SymbolNotificationTypes Enumeration	3222
6.13.12	ValueCreationModes Enumeration	3223
6.13.13	ResultHandleAccess Class	3224
6.13.14	ResultReadValueAccess2.I, V. Class	3226
6.13.15	ValueAccessMode Enumeration	3228
6.13.16	ValueUpdateMode Enumeration.....	3229
6.14	TwinCAT.Ads.Server.TypeSystem Namespace	3230
6.14.1	DataArea Class	3230
6.14.2	IAddSymbolicInformation Interface	3234
6.14.3	ServerSymbolFactory Class.....	3242
6.15	TwinCAT.Router Namespace	3255
6.15.1	RouterPort Class.....	3255
6.15.2	RouterPortInfo Class.....	3264
6.16	TwinCAT.SystemService Namespace	3272
6.16.1	SystemServiceIndexGroup Enumeration	3273
6.16.2	TargetInfoOffset Enumeration.....	3275
6.16.3	AdsSysServState Structure.....	3275
6.16.4	SystemServiceExtension Class	3281
6.17	TwinCAT.TIs Namespace	3288
6.17.1	TIsException Class.....	3288
6.18	TwinCAT.Ads.Extensions Namespace	3292
6.18.1	ConnectionStateExtension Class.....	3292

1 Foreword

1.1 Notes on the documentation

This description is intended exclusively for trained specialists in control and automation technology who are familiar with the applicable national standards.

For installation and commissioning of the components, it is absolutely necessary to observe the documentation and the following notes and explanations.

The qualified personnel is obliged to always use the currently valid documentation.

The responsible staff must ensure that the application or use of the products described satisfies all requirements for safety, including all the relevant laws, regulations, guidelines, and standards.

Disclaimer

The documentation has been prepared with care. The products described are, however, constantly under development.

We reserve the right to revise and change the documentation at any time and without notice.

No claims to modify products that have already been supplied may be made on the basis of the data, diagrams, and descriptions in this documentation.

Trademarks

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® and XPlanar® are registered and licensed trademarks of Beckhoff Automation GmbH.

If third parties make use of designations or trademarks used in this publication for their own purposes, this could infringe upon the rights of the owners of the said designations.

Patents

The EtherCAT Technology is covered, including but not limited to the following patent applications and patents:

EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702
and similar applications and registrations in several other countries.

EtherCAT®

EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany

Copyright

© Beckhoff Automation GmbH & Co. KG, Germany.

The distribution and reproduction of this document as well as the use and communication of its contents without express authorization are prohibited.

Offenders will be held liable for the payment of damages. All rights reserved in the event that a patent, utility model, or design are registered.

1.2 For your safety

Safety regulations

Read the following explanations for your safety.

Always observe and follow product-specific safety instructions, which you may find at the appropriate places in this document.

Exclusion of liability

All the components are supplied in particular hardware and software configurations which are appropriate for the application. Modifications to hardware or software configurations other than those described in the documentation are not permitted, and nullify the liability of Beckhoff Automation GmbH & Co. KG.

Personnel qualification

This description is only intended for trained specialists in control, automation, and drive technology who are familiar with the applicable national standards.

Signal words

The signal words used in the documentation are classified below. In order to prevent injury and damage to persons and property, read and follow the safety and warning notices.

Personal injury warnings**⚠ DANGER**

Hazard with high risk of death or serious injury.

⚠ WARNING

Hazard with medium risk of death or serious injury.

⚠ CAUTION

There is a low-risk hazard that could result in medium or minor injury.

Warning of damage to property or environment**NOTICE**

The environment, equipment, or data may be damaged.

Information on handling the product

This information includes, for example:
recommendations for action, assistance or further information on the product.

1.3 Notes on information security

The products of Beckhoff Automation GmbH & Co. KG (Beckhoff), insofar as they can be accessed online, are equipped with security functions that support the secure operation of plants, systems, machines and networks. Despite the security functions, the creation, implementation and constant updating of a holistic security concept for the operation are necessary to protect the respective plant, system, machine and networks against cyber threats. The products sold by Beckhoff are only part of the overall security concept. The customer is responsible for preventing unauthorized access by third parties to its equipment, systems, machines and networks. The latter should be connected to the corporate network or the Internet only if appropriate protective measures have been set up.

In addition, the recommendations from Beckhoff regarding appropriate protective measures should be observed. Further information regarding information security and industrial security can be found in our <https://www.beckhoff.com/secguide>.

Beckhoff products and solutions undergo continuous further development. This also applies to security functions. In light of this continuous further development, Beckhoff expressly recommends that the products are kept up to date at all times and that updates are installed for the products once they have been made available. Using outdated or unsupported product versions can increase the risk of cyber threats.

To stay informed about information security for Beckhoff products, subscribe to the RSS feed at <https://www.beckhoff.com/secinfo>.

2 TwinCAT ADS .NET API Documentation

The TwinCAT .NET API implements support for the TwinCAT Automation Device specification (ADS). It can be used within .NET Framework programming languages and support ADS Client implementations. The ADS API interface permits:

- The Implementation of ADS Clients
- Browsing of (ADS) server side symbolic information.
- Reading and writing ProcessImage information in Raw or in type safe manner.
- Receiving ADS Notifications as events.

Getting Started

For getting started please have a look at the following documents:

- [Prerequisites \[▶ 17\]](#)
- [Installation \[▶ 17\]](#)
- [Concepts \[▶ 23\]](#)
- [HowTo Samples \[▶ 42\]](#)

Background information about the TwinCAT ADS protocol can be found here:

- [ADS Introduction](#)
- [TwinCAT ADS Device concept](#)

for common ADS information.

From the conceptual standpoint within this ADS .NET API reference documentation, the most important starting points for reading are the following Classes/Methods:

Main documentation entry points

Description	Link
Addressing ADS Devices via AmsNetId Address information.	AmsNetId [▶ 767]
Communicate to ADS Devices via the AdsClient class.	AdsClient [▶ 183]
Implement your own (virtual) ADS Server	AdsServer [▶ 1375]
Browse target system symbolic information. Creation of the SymbolLoader via Factory class.	SymbolLoaderFactory.Create(IConnection,ISymbolLoaderSettings) [▶ 1924]
Session and Connection management on top of the ADS communication channel established by the AdsClient.	AdsSession [▶ 701] , AdsConnection [▶ 412]
Usage of Ads.Rx (Reactive extensions) to write reactive code.	AdsClientExtensions [▶ 1249] , AdsClientExtensions [▶ 1249] The ADS reactive extensions are only available for usage, when the corresponding Nuget package is downloaded from Nuget.org. Beckhoff.TwinCAT.Ads.Reactive package on Nuget

Other Resources

[HowTo Samples \[▶ 42\]](#)

[Version History \[▶ 19\]](#)

2.1 Prerequisites

Preconditions for installing/Using the TwinCAT .NET ADS Communication API Version 6.X

To develop an application the TwinCAT .NET ADS Communication API the following software infrastructure must be available on the development system.

- An ADS/AMS Router (e.g. TwinCAT >= 4024.10) that is capable to Route the ADS/AMS Frames between systems.
- The .NET SDK to develop the application software.
- ADS Nuget packages referenced within the application project to develop TwinCAT ADS data exchange / communication.

ADS/AMS Router

- A TwinCAT installation Version >= 4024.10.

Because the Beckhoff.TwinCAT.Ads Version 6.X uses internal interfaces that are available only from TwinCAT 4024.10 on, an appropriate version must be installed locally.

The package doesn't work with older installations - the actually only alternative is the AdsRouterConsole.

- Or alternatively a running instance of the '[Beckhoff.TwinCAT.Ads.TcpRouter](#)' for customized router implementations or the 'ready-out-of-the-box' router package '[Beckhoff.TwinCAT.Ads.AdsRouterConsole](#)' for use on system without TwinCAT.

A Software development kit for .NET (SDK)

At least one of the following SDKs and target frameworks is necessary:

- .NET 5.0 or later
- .NET Core 3.1 or later
- .NET Framework 4.61 or later
- .NET Standard 2.0 compatible SDK or later

The Beckhoff.TwinCAT.Ads components for the application.

The following Nuget packages must be added to the application project as package references:

- The '[Beckhoff.TwinCAT.Ads Nuget](#)' package.
- and optionally the '[Beckhoff.TwinCAT.Ads.Reactive](#)' package.

2.2 Installation

The now preferred way to install the TwinCAT ADS .NET Communication API is to use the NuGet.org package manager.

Beckhoff.TwinCAT.Ads package from Nuget.org repository.

This is the main package implementing the ADS client functionality. This is needed to establish ADS connections to local and remote devices.

Please follow the install instructions on the Nuget.org site.

[Beckhoff.TwinCAT.Ads Nuget Package](#)

Beckhoff.TwinCAT.Ads.Reactive package from Nuget.org repository.

This package installs Reactive extensions on top of the [Beckhoff.TwinCAT.Ads Nuget Package](#) and installs additional extensions to map ADSNotifications to observable events.

Please follow the install instructions on the Nuget.org site.

[Beckhoff.TwinCAT.Ads.Reactive Nuget Package](#)

3 Version History

This chapter documents the different active development lines of the **Beckhoff.TwinCAT.ADS .NET API** and its package Versions.

Several flavours of the .NET Framework exist (both actually still supported and outdated). Therefore different versions of the packages exist that support the different .NET lines and releases. Background information can be read here: [Fundamentals about .NET Full Framework, .NET Core and the supported Operating Systems](#)

Here we highlight the different version lines of the **Beckhoff.TwinCAT.ADS .NET API** package and their link to the .NET Framework.

Version support lifecycle

Package	Description	.NET Framework	TwinCAT	Active Support
Version 6.2 [▶ 20]	Frameworks support of .NET 8.0 and below (.NET Core)	net8.0 (LTS), net6.0 (LTS), netstandard2.0	>= 3.1.4024.10 Requirement on the Host system. No version limitation in remote system communication.	X
Version 6.1 [▶ 20]	Framework support of .NET 7.0 [1] and below (.NET Core)	net7.0, net6.0 (LTS), netstandard2.0	>= 3.1.4024.10 Requirement on the Host system. No version limitation in remote system communication.	X [1]
Version 6.0 [▶ 21]	Framework support of .NET 6.0 and below (.NET Core)	net6.0 (LTS), netcoreapp3.1, netstandard2.0, net461	>= 3.1.4024.10 Requirement on the Host system. No version limitation in remote system communication.	X
Version 5.X [▶ 21]	Framework support of .NET 5.0 [2] and below (.NET Core)	net5.0, netcoreapp3.1, netstandard2.0, net461	>= 3.1.4024.10 Requirement on the Host system. No version limitation in remote system communication.	O [2]
Version 4.X [▶ 22]	Package basing on .NET Framework 4.0 (Full Framework)	net4	All	X

[1]: Microsoft support for .NET7 ends with May 14, 2024.

[2]: Microsoft support for .NET5 ended with May 8, 2022. Therefore it is recommended to update **Beckhoff.TwinCAT** packages from Version 5 to Version 6.

[Migrate from ASP.NET Core 5.0 to 6.0.](#)

[Migrating to the latest .NET.](#)

[Microsoft .NET support lifecycle.](#)

Other Resources

[TwinCAT ADS .NET API Documentation \[▶ 16\]](#)

3.1 Version 6.2

[Beckhoff.TwinCAT.Ads Package Version 6.2](#)

It supports **TwinCAT version** $\geq 3.1.4024.10$ running on the same Device. **TwinCAT** versions $<3.1.4024.10$ are *not* supported! There are no limitations with remote TwinCAT Systems. For accessing TwinCAT2 locally, please use Version 4.X.

Because this package is basing on .NET Core, the support of different Operating Systems like Linux or MacOS beneath windows is given.

This package contains multi-platform binaries for **net70**, **net60**, **netstandard2.0** so that many flavors of applications and Operation systems can be covered. For more information please see the Microsoft documentation.

The Full .NET Framework up to .NET482 is covered by **netstandard2.0**.

Key characteristics

- .NET 6 and above supports a wide range of operating systems.
[Supported OperatingSystems \(net80\)](#)
- Basing on .NET Core (Verison 8.0)
Target support for **net80 (LTS)**, **net60 (LTS)**, **netstandard2.0**
- Asynchronous programming model (support of the `async/await` statements, see also [Concept of async operation \[▶ 24\]](#))
- Integrated support for implementing customized AdsServers ([AdsServer \[▶ 1375\]](#) and [AdsSymbolicServer \[▶ 1468\]](#))
- Optimized interfaces enhancing scalability and performance by using Span.T. and Memory.T..
- Reactive Extension Functions for PollingValues, AdsStates, ...

Other Resources

[Version History \[▶ 19\]](#)

3.2 Version 6.1

[Beckhoff.TwinCAT.Ads Package Version 6.1](#)

It supports **TwinCAT version** $\geq 3.1.4024.10$ running on the same Device. **TwinCAT** versions $<3.1.4024.10$ are *not* supported! There are no limitations with remote TwinCAT Systems. For accessing TwinCAT2 locally, please use Version 4.X.

Because this package is basing on .NET Core, the support of different Operating Systems like Linux or MacOS beneath windows is given.

This package contains multi-platform binaries for **net70**, **net60**, **netstandard2.0** so that many flavors of applications and Operation systems can be covered. For more information please see the Microsoft documentation.

The Full .NET Framework up to .NET482 is covered by **netstandard2.0**.

Key characteristics

- Basing on .NET Core (Verison 7.0)
.NET 7 and below support a wide range of operating systems ([Supported OperatingSystems \(net7.0\)](#)).

- Target support for **net70**, **net60 (LTS)**, **netstandard2.0**
- Asynchronous programming model (support of the `async/await` statements, see also [Concept of async operation](#) [[▶ 24](#)])
- Integrated support for implementing customized `AdsServers` ([AdsServer](#) [[▶ 1375](#)] and [AdsSymbolicServer](#) [[▶ 1468](#)])
- Optimized interfaces enhancing scalability and performance by using `Span.T.` and `Memory.T.`
- Reactive Extension Functions for `PollingValues`, `AdsStates`, ...

Other Resources

[Version History](#) [[▶ 19](#)]

3.3 Version 6.0

[Beckhoff.TwinCAT.Ads](#) Package Version 6.0

It supports **TwinCAT version** $\geq 3.1.4024.10$ running on the same Device. **TwinCAT** versions $<3.1.4024.10$ are *not* supported! There are no limitations with remote TwinCAT Systems. For accessing TwinCAT2 locally, please use Version 4.X.

Another reason for usage of this package is the support of different Operating Systems like Linux or MacOS beneath windows.

This package contains multi-platform binaries for **net60**, **netstandard2.0**, **net461**, **netcoreapp3.1** so that many flavors of applications and Operation systems can be covered. For more information please see the Microsoft documentation.

Key characteristics

- Basing on .NET Core (Long Term Version 6.0)
.NET 6 (LTS) supports a wide range of operating systems ([Supported OperatingSystems \(net6.0\)](#)).
- Target support for **net60 (LTS)**, **netstandard2.0**, **net461**, **netcoreapp3.1**
- Asynchronous programming model (support of the `async/await` statements, see also [Concept of async operation](#) [[▶ 24](#)])
- Integrated support for implementing customized `AdsServers` ([AdsServer](#) [[▶ 1375](#)] and [AdsSymbolicServer](#) [[▶ 1468](#)])
- Optimized interfaces enhancing scalability and performance by using `Span.T.` and `Memory.T.`

Other Resources

[Version History](#) [[▶ 19](#)]

3.4 Version 5.X

[Beckhoff.TwinCAT.Ads](#) Package Version 5.0

Microsoft support for .NET5 ends with May 8, 2022. Therefore it is recommended to update Beckhoff.TwinCAT packages from Version 5 to Version 6 because this version is outdated.

Therefore, although `.net50` is still operational and can still downloaded from Microsoft, it is *not* recommended to use the Series 5.X package for new application releases anymore. **This package is not under active development anymore and should be replaced by the [Version 6.X series](#) [[▶ 21](#)] as soon as possible.**

It supports **TwinCAT version** $\geq 3.1.4024.10$ running on the same Device. **TwinCAT** versions $<3.1.4024.10$ are *not* supported! There are no limitations with remote TwinCAT Systems. For accessing TwinCAT2 locally, please use Version 4.X.

Different Operating Systems like Linux or MacOS are supported beneath windows.

This package contains multi-platform binaries for **net50**, **netstandard2.0**, **net461**, **netcoreapp3.1** so that many flavors of applications and Operation systems can be covered. For more information please see the Microsoft documentation.

Key characteristics

- Basing on .NET Core.
.NET 5 supports a wide range of operating systems ([Supported OperatingSystems](#)).
- Target support for **net50**, **netstandard2.0**, **net461**, **netcoreapp3.1**
- Asynchronous programming model (support of the async/await statements, see also [Concept of async operation \[► 24\]](#))
- Integrated support for implementing customized AdServers ([AdsServer \[► 1375\]](#))
- Optimized interfaces enhancing scalability and performance by using Span.T. and Memory.T..

Hints for upgrading packages to version 6.0

[Migrate from ASP.NET Core 5.0 to 6.0.](#)

[Migrating to the latest .NET.](#)

[Microsoft .NET support lifecycle.](#)

Other Resources

[Version History \[► 19\]](#)

3.5 Version 4.X

[Beckhoff.TwinCAT.Ads 4.X Series Package](#)

The 4.X package series is recommended to use when the support of **TwinCAT version < 3.1.4024.10** (including TwinCAT 2) is a necessity on the same system. Although it is based on the Full .NET Framework (net4) it is useable in newer .NET Core Generations (net8.0, net6.0 but not in standard2.0 applications). However, the use then is limited to **Windows only**.

This version is still under active support.

Key characteristics

- Windows only
- Basing on .NET Framework 4.0 (**net40**, .NET Full Framework)
- Supports all versions of TwinCAT (including TwinCAT 2 and TwinCAT 3 versions) running on the local system.
- Synchronous communication
- Asynchronous Read/Write requestes (with async-Statement) are not supported.

Other Resources

[Version History \[► 19\]](#)

4 Concepts

Concepts introduction

Concepts

Concepts discussed

Name	Description
IndexGroup / IndexOffset	Read/Write values by IndexGroup / IndexOffset [▶ 23]
Symbolic access	Read/Write values by symbolic instance path [▶ 29]
Access via symbol handle	Read/Write values by symbol handles. [▶ 30]
Asynchronous programming (async, await)	Concept of async operation [▶ 24]
ADS Notifications	Concept of ADS Notifications [▶ 26]
Symbol loader access	Access symbolic information by Symbol loader [▶ 35]
Marshalling values via ANYTYPE concept	ANY Type blittable type marshalling [▶ 31]
Dynamic automatic marshalling via Dynamic Language Runtime	Use of the .NET Framework Dynamic Language Runtime (DLR) [▶ 36]
Access via reactive extensions	Observer value changes by ADS Reactive Extensions [▶ 38]

Other Resources

[TwinCAT ADS .NET API Documentation \[▶ 16\]](#)

4.1 Access Data via IndexGroup/IndexOffset

Reading/Writing values by Index/Group index offset are the most basic way to access data via ADS. This address combination directly link into the process image of virtual ADS Devices.

As long the process image is static this is unproblematic and a system near access, but if the content is more dynamic and the address changes over time the IndexGroup/IndexOffset can get invalid.

Examples about moving addresses could be:

- Changed Parametrization of IO (and Re-activation)
- The PLC Online change
- New Plc Downloads

In that case other access methods could be advantageous.

Another important point is that the data access is not type safe. The values are read or written to or from byte buffers and the proper marshalling/unmarshalling is the task of the application code.

Asynchronous access

Access ProcessImage Data by IndexGroup/IndexOffset

```
CancellationToken cancel = CancellationToken.None;
```

```
using (AdsClient client = new AdsClient())
{
    UInt32 valueToRead = 0;
    UInt32 valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);
```

```

byte[] writeData = new byte[sizeof(uint)];

// Write an UINT32 Value
MemoryStream writeStream = new MemoryStream(writeData);
BinaryWriter writer = new BinaryWriter(writeStream);
writer.Write(valueToWrite);
ResultWrite resultWrite = await client.WriteAsync(0x4020, 0x0, writeData.AsMemory(),cancel);

// Read an UINT32 Value
byte[] readData = new byte[sizeof(uint)];
ResultRead resultRead = await client.ReadAsync(0x4020, 0x0, readData.AsMemory(),cancel);

MemoryStream readStream = new MemoryStream(readData);
BinaryReader reader = new BinaryReader(readStream);
valueToRead = reader.ReadUInt32();
}

```

Synchronous access

Access ProcessImage Data by IndexGroup/IndexOffset

```

using (AdsClient client = new AdsClient())
{
    UInt32 valueToRead = 0;
    UInt32 valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);

    // Write an UINT32 Value
    byte[] writeData = new byte[sizeof(uint)];
    MemoryStream writeStream = new MemoryStream(writeData);
    BinaryWriter writer = new BinaryWriter(writeStream);
    writer.Write(valueToWrite);
    client.Write(0x4020, 0x0, writeData);

    // Read an UINT32 Value
    byte[] readData = new byte[sizeof(uint)];
    int readBytes = client.Read(0x4020, 0x0, readData);

    MemoryStream readStream = new MemoryStream(readData);
    readStream.Position = 0;
    BinaryReader reader = new BinaryReader(readStream);
    valueToRead = reader.ReadUInt32();
}

```

4.2 Async programming (async, await)

Since .NET Version 4.0, the .NET API supports concurrent operation in form of the compiler support of the async/await statements. This is a special code generation supported flavor of concurrency that uses so called 'futures'. A 'future' (or promise) is a type that represents an operation that will be completed in the future and is represented by the .NET type `Task` or `Task.TRESULT` type. This ensures that the called asynchronous method is started on call and delivers its result later on, without blocking the calling thread. As consequence the calling thread is able to process other work in the meanwhile. The deep support level in the .NET framework and the underlying code generation makes asynchronous programming almost as easy as synchronous programming.

From version 5.0.0 on, the TwinCAT.Ads API also supports the async programming model.

The main advantages are:

- Remains responsiveness of GUI applications during ADS communication
- Asynchronous programming enables scalability (mainly on the Server side)
- Easy synchronization between threads, nearly as simple as asynchronous programming.
- Functional, stateless programming style seamlessly integrating with other concurrency techniques like parallel programming ([.NET Task Parallel Library](#)) or reactive programming ([Reactive Extensions \[▶ 381\]](#)).

More about asynchronous programming can be read here: [Asynchronous programming](#)

and here: [The Task asynchronous programming model in C#](#)

Example

Read/Write AnyType by IndexGroup/IndexOffset (asynchronously)

```
using (AdsClient client = new AdsClient())
{
    CancellationToken cancel = CancellationToken.None;

    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);
    ResultWrite resultWrite = await client.WriteAnyAsync(0x4020, 0x0, valueToWrite, cancel);
    bool succeeded = resultWrite.Succeeded;

    ResultValue<uint> resultRead = await client.ReadAnyAsync<uint>(0x4020, 0x0, cancel);

    if (resultRead.Succeeded)
    {
        valueToRead = (uint)resultRead.Value;
    }
}
```

Read/Write AnyType by variable handle (asynchronously)

```
using (AdsClient client = new AdsClient())
{
    CancellationToken cancel = CancellationToken.None;
    uint varHandle = 0;
    client.Connect(AmsNetId.Local, 851);

    uint valueToRead = 0;
    uint valueToWrite = 42;

    ResultHandle resultHandle = await client.CreateVariableHandleAsync("MAIN.nCounter", cancel);
    varHandle = resultHandle.Handle;

    if (resultHandle.Succeeded)
    {
        try
        {
            ResultWrite resultWrite = await client.WriteAnyAsync(varHandle, valueToWrite, cancel);
            ResultValue<uint> resultRead = await client.ReadAnyAsync<uint>(varHandle, cancel);

            if (resultRead.Succeeded)
                valueToRead = resultRead.Value;
        }
        finally
        {
            // Unregister VarHandle after Use
            ResultAds result = await client.DeleteVariableHandleAsync(varHandle, cancel);
        }
    }
}
```

Read/Write AnyType by SymbolBrowser (asynchronously)

```
using (AdsClient client = new AdsClient())
{
    CancellationToken cancel = CancellationToken.None;

    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);

    // Load all Symbols + DataTypes
    ISymbolLoader loader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);

    ResultSymbols resultSymbols = await loader.GetSymbolsAsync(cancel);

    if (resultSymbols.Succeeded)
    {
        Symbol symbol = (Symbol)resultSymbols.Symbols["MAIN.nCounter"];

        // Works for ALL Primitive 'ANY TYPES' Symbols
        ResultWriteAccess resultWrite = await symbol.WriteValueAsync(valueToWrite, cancel);
        ResultReadValueAccess resultRead = await symbol.ReadValueAsync(cancel);
    }
}
```



```

        if (resultRead.Succeeded)
            valueToRead = (uint)resultRead.Value;

        // Simple filtering of Symbols
        Regex filterExpression = new Regex(pattern: @"^MAIN.*"); // Everything that starts with "MAIN"

        // FilterFunction that filters for the InstancePath
        Func<ISymbol, bool> filter = s => filterExpression.IsMatch(s.InstancePath);
        SymbolIterator iterator = new SymbolIterator(symbols: resultSymbols.Symbols, recurse: true,
selector: filter);

        foreach (ISymbol filteredSymbol in iterator)
        {
            Console.WriteLine(filteredSymbol.InstancePath);
        }
    }
}

```

4.3 Use of ADS Notifications

If values from a PLC or NC are to be displayed continuously on a user interface, then it is very inefficient to use asynchronous read access, since this function must be called cyclically (polled triggered by a timer). Instead of using a pulling (read) model, ADS Notifications are implementing a push model. That means ADS Notifications are triggered by the sender and form a single or series of ADS messages/events. Together with these ADS Notifications, values can be transmitted. A distinction is drawn between whether the TwinCAT server is to transmit the values cyclically, or only when the values change.

In principle (raw mode) a notification is begun with the registration of the notification [AddDeviceNotificationAsync \[▶ 235\]](#) (asynchronous) or [AddDeviceNotification \[▶ 232\]](#) (synchronous). After this, events are automatically fired by TwinCAT. [DeleteDeviceNotificationAsync \[▶ 254\]](#) (asynchronous) or [DeleteDeviceNotification \[▶ 253\]](#) (synchronous) is used to halt the notification again. Since the number of notifications is limited, you should ensure the notifications no longer required by your program are unregistered/deleted.

There exist several 'modes' for different type of ADS Notification triggers. For a complete list please consult [AdsTransMode \[▶ 740\]](#).

All the following examples demonstrate how to receive ADS Notifications. The .NET ADS API supports different information layers which different levels of ADS Notification support. All are using a PLC variable in the PLC and each time the value of the PLC variable changes, an ADS Notification message is sent and the registered callback method is invoked with event arguments that contain all the necessary information (value, time stamp, ...).

Hint: Don't use time intensive executions or ADS commands inside of your callback (not more than approx. 500). Remind to sync your callback in your main thread (typically the UI thread) if necessary, because the ADS Notifications appear on a background thread.

Using ADS Notifications with Symbolic information

C#

```

private void SymbolValueChanged()
{
    using (AdsClient client = new AdsClient())
    {
        // Connect to target
        client.Connect(AmsNetId.Local, 851);
        Symbol symbol = null;

        try
        {
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
            // DINT Type (UINT32)
            symbol = (Symbol)loader.Symbols["MAIN.nCounter"];

            // Set the Notification Settings of the Symbol if NotificationSettings.Default is not appropriate
            // Check for change every 500 ms
            symbol.NotificationSettings = new NotificationSettings(AdsTransMode.OnChange, 500, 0);
        }
    }
}

```

```

        symbol.ValueChanged += Symbol_ValueChanged; // Registers the notification
        Thread.Sleep(5000); // Sleep the main thread to get some (asynchronous Notifications)
    }
    finally
    {
        // Unregister the Event and the underlying Handle
        symbol.ValueChanged -= Symbol_ValueChanged; // Unregisters the notification
    }
}

private void Symbol_ValueChanged(object sender, ValueChangedEventArgs e)
{
    Symbol symbol = (Symbol)e.Symbol;

    // Object Value can be cast to int automatically, because it is an Primitive Value (DINT --
    > Int32).
    // The Symbol information is used internally to cast the value to its appropriate .NET Type.
    int iVal = (int)e.Value;

    // If Synchronization is needed (e.g. in Windows.Forms or WPF applications)
    // we could synchronize via SynchronizationContext into the UI Thread

    /*SynchronizationContext syncContext = SynchronizationContext.Current;
    _context.Post(status => someLabel.Text = iVal.ToString(), null); // Non-blocking post */
}

```

Using ADS Notifications in 'ANYTYPE' style

C#

```

//AdsStream readStream = new AdsStream(sizeof(UInt32));

private void ReceiveNotifications()
{
    using (AdsClient client = new AdsClient())
    {
        // Add the Notification event 'Ex' handler
        client.AdsNotificationEx += Client_AdsNotification;

        // Connect to target
        client.Connect(AmsNetId.Local, 851);
        uint notificationHandle = 0;

        try
        {
            // Notification to a ZDINT Type (UINT32)
            // Check for change every 200 ms
            notificationHandle = client.AddDeviceNotificationEx("MAIN.nCounter", new NotificationSettings
(AdsTransMode.OnChange, 200, 0), null, typeof(uint));
            Thread.Sleep(5000); // Sleep the main thread to get some (asynchronous Notifications)
        }
        finally
        {
            // Unregister the Event / Handle
            client.DeleteDeviceNotification(notificationHandle);
            client.AdsNotificationEx -= Client_AdsNotification;
        }
    }
}

private void Client_AdsNotification(object sender, AdsNotificationExEventArgs e)
{
    // Or here we know about UDINT type --> can be marshalled as UINT32
    uint nCounter = (uint)e.Value;

    // If Synchronization is needed (e.g. in Windows.Forms or WPF applications)
    // we could synchronize via SynchronizationContext into the UI Thread

    /*SynchronizationContext syncContext = SynchronizationContext.Current;
    _context.Post(status => someLabel.Text = nCounter.ToString(), null); // Non-blocking post */
}

```

Asynchronous registering of Notifications

Trigger on changed values by ADS Notifications

```
private async Task RegisterNotificationsAsync()
{
    CancellationToken cancel = CancellationToken.None;

    using (AdsClient client = new AdsClient())
    {
        // Add the Notification event handler
        client.AdsNotification += Client_AdsNotification2;

        // Connect to target
        client.Connect(AmsNetId.Local, 851);
        uint notificationHandle = 0;

        // Notification to a DINT Type (UINT32)
        // Check for change every 200 ms

        //byte[] notificationBuffer = new byte[sizeof(UINT32)];
        int size = sizeof(UInt32);

        ResultHandle result = await client.AddDeviceNotificationAsync("MAIN.nCounter", size, new NotificationSettings(AdsTransMode.OnChange, 200, 0), null, cancel);

        if (result.Succeeded)
        {
            notificationHandle = result.Handle;
            await Task.Delay(5000); // Wait asynchronously without blocking the UI Thread.
            // Unregister the Event / Handle
            ResultAds result2 = await client.DeleteDeviceNotificationAsync(notificationHandle, cancel);
        }
        client.AdsNotification -= Client_AdsNotification2;
    }
}

private void Client_AdsNotification2(object sender, AdsNotificationEventArgs e)
{
    // Or here we know about UDINT type --> can be marshalled as UINT32
    uint nCounter = BinaryPrimitives.ReadUInt32LittleEndian(e.Data.Span);

    // If Synchronization is needed (e.g. in Windows.Forms or WPF applications)
    // we could synchronize via SynchronizationContext into the UI Thread

    /*SynchronizationContext syncContext = SynchronizationContext.Current;
    _context.Post(status => someLabel.Text = nCounter.ToString(), null); // Non-blocking post */
}
```

Synchronous registering of Notifications

Trigger on changed values by ADS Notifications

```
private void RegisterNotifications()
{
    using (AdsClient client = new AdsClient())
    {
        // Add the Notification event handler
        client.AdsNotification += Client_AdsNotification;

        // Connect to target
        client.Connect(AmsNetId.Local, 851);
        uint notificationHandle = 0;

        try
        {
            // Notification to a DINT Type (UINT32)
            // Check for change every 200 ms

            int size = sizeof(UInt32);
            //byte[] notificationBuffer = new byte[sizeof(UInt32)];

            notificationHandle = client.AddDeviceNotification("MAIN.nCounter", size, new NotificationSettings(AdsTransMode.OnChange, 200, 0), null);
            Thread.Sleep(5000); // Sleep the main thread to get some (asynchronous Notifications)
        }
        finally
        {

```

```

        // Unregister the Event / Handle
        client.DeleteDeviceNotification(notificationHandle);
        client.AdsNotification -= Client_AdsNotification;
    }
}

private void Client_AdsNotification(object sender, AdsNotificationEventArgs e)
{
    // Or here we know about UDINT type --> can be marshalled as UInt32
    uint nCounter = BinaryPrimitives.ReadUInt32LittleEndian(e.Data.Span);

    // If Synchronization is needed (e.g. in Windows.Forms or WPF applications)
    // we could synchronize via SynchronizationContext into the UI Thread

    /*SynchronizationContext syncContext = SynchronizationContext.Current;
    _context.Post(status => someLabel.Text = nCounter.ToString(), null); // Non-blocking post */
}

```

Using reactive ADS Notifications

C#

```

// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];

    // Reactive Notification Handler
    var valueObserver = Observer.Create<object>(val =>
    {
        Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToString()));
    });

    cycleCount.NotificationSettings = new NotificationSettings(AdsTransMode.OnChange, 500, 5000); // optional: Change NotificationSettings on Symbol

    // Turning ADS Notifications into sequences of Value Objects (Taking 20 Values)
    // and subscribe to them.
    IDisposable subscription = cycleCount.WhenValueChanged().Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}

```

4.4 Access Data via Symbolic path

The Read/Write Access by symbol path solves the issue of directly accessing the process image. With specifying the access path to the symbol, the symbol address can be found by a binary search (internally) and reading / writing symbols is independent of the location within the process image.

This access method can only be used, when the ADS device is supporting symbolic information like the TwinCAT PLC.

Because its indirect access, the performance is slightly worse than the direct access via IndexGroup/ IndexOffset. However there are internal optimizations to cache handles to the already used symbols to accelerate repeated access.

Asynchronous access

Access symbolic data by instance/symbol path

```

using (AdsClient client = new AdsClient())
{
    CancellationToken cancel = CancellationToken.None;
    uint valueToRead = 0;
}

```

```

uint valueToWrite = 42;

client.Connect(AmsNetId.Local, 851);

ResultWrite resultWrite = await client.WriteValueAsync("MAIN.nCounter", valueToWrite, cancel);
ResultValue<uint> resultRead = await client.ReadValueAsync<uint>("MAIN.nCounter",cancel);

if (resultRead.Succeeded)
    valueToRead = resultRead.Value;
}

```

Synchronous access

Access symbolic data by instance/symbol path

```

using (AdsClient client = new AdsClient())
{
    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);
    client.WriteValue("MAIN.nCounter", valueToWrite);
    valueToRead = (uint)client.ReadValue("MAIN.nCounter", typeof(uint));
}

```

4.5 Access Data via Symbol handles

The Read/Write Access by handle solves the issue of directly accessing the process image like the symbol path access. Because the address is accessed indirectly by the symbol path creating a variable handle, the read/write works also when the data object has changed its position within the process image.

However the cost for this are two extra ADS communication roundtrips by the 'CreateVariableHandle' and 'DeleteVariableHandle' calls compared to the IndexGroup/IndexOffset access methods. It is the responsibility of the application code to optimize these accesses.

Asynchronous access

Access symbolic values by handle

```

CancellationToken cancelT = CancellationToken.None;
using (AdsClient client = new AdsClient())
{
    client.Connect(AmsNetId.Local, 851);

    ushort valueToRead = 0; // System.UInt16
    ushort valueToWrite = 42; // System.UInt16

    // Create the Variable Handle
    ResultHandle resultHandle = await client.CreateVariableHandleAsync("MAIN.testVar", cancel); //
Test Var is defined as PLC INT

    if (resultHandle.Succeeded)
    {
        uint varHandle = 0;

        try
        {
            // Write an UINT16 Value
            byte[] writeData = new byte[sizeof(ushort)];

            MemoryStream writeStream = new MemoryStream(writeData);
            BinaryWriter writer = new BinaryWriter(writeStream);
            writer.Write(valueToWrite); // Marshal the Value
            ResultWrite resultWrite = await client.WriteAsync(varHandle, writeData.AsMemory(), cancelT);

            bool succeeded = resultWrite.Succeeded;

            // Read an UINT16 Value
            byte[] readData = new byte[sizeof(ushort)];
            ResultRead resultRead = await client.ReadAsync(varHandle, readData.AsMemory(), cancel);

            if (resultRead.Succeeded)
            {

```

```

        MemoryStream readStream = new MemoryStream(readData);
        BinaryReader reader = new BinaryReader(readStream);
        valueToRead = reader.ReadUInt16(); // Unmarshal the Value
    }
}
finally
{
    // Unregister VarHandle after Use
    ResultAds result = await client.DeleteVariableHandleAsync(varHandle, cancel);
}
}
}

```

Synchronous access

Access symbolic values by handle

```

using (AdsClient client = new AdsClient())
{
    uint varHandle = 0;
    client.Connect(AmsNetId.Local, 851);
    try
    {
        UInt16 valueToRead = 0;
        UInt16 valueToWrite = 42;

        // Create the Variable Handle
        varHandle = client.CreateVariableHandle("MAIN.testVar"); //Test Var is defined as PLC INT

        // Write an UINT16 Value
        byte[] writeData = new byte[sizeof(ushort)];

        MemoryStream writeStream = new MemoryStream(writeData);
        BinaryWriter writer = new BinaryWriter(writeStream);
        writer.Write(valueToWrite); // Marshal the Value
        client.Write(varHandle, writeData.AsMemory());

        // Read an UINT16 Value
        byte[] readData = new byte[sizeof(ushort)];

        MemoryStream readStream = new MemoryStream(readData);
        client.Read(varHandle, readData.AsMemory());
        BinaryReader reader = new BinaryReader(readStream);
        valueToRead = reader.ReadUInt16(); // Unmarshal the Value
    }
    finally
    {
        // Unregister VarHandle after Use
        client.DeleteVariableHandle(varHandle);
    }
}

```

4.6 Value marshalling with ANYTYPE concept

This topic describes reading and writing variables/symbols of 'any' type with the help of the `ReadAny` and `WriteAny` (`ReadSymbol`, `WriteSymbol`) methods. The value will be marshalled / cast directly from/to its appropriate .NET type, what eases the value access.

'Any' types in this context are all types that are 'blittable' to the process image - what means that the memory layout on both sides of the data transfer is equal (e.g some primitive types) or can be marshalled by the marshalling mechanisms of .NET (see 'PlcStruct' in the example below). The memory layout specification can be customized with the 'System.Runtime.InteropServices.StructLayoutAttribute' on the .NET side (see MSDN) and the 'pack_mode' attribute on the TwinCAT PLC Side (TwinCAT 3). TwinCAT 2 only supports a memory layout of `PACK = 1`.

The appropriate .NET type must be known during compile time and is passed to the methods as parameter. In case of a `ReadAny` call, the read data will be returned as a object. The type of the object is marshalled to the type specified as parameter type. Because the data size and the memory alignment is taken from this type specification, it is so important that this specification fits to the memory representation in the ADS device (e.g. the PLC).

Because some data types (arrays and strings) need additional information, an overload of the method `ReadAny` exists, that takes an additional parameter `args`. A Full list of supported types can be found in the documentation of the overloaded method.

Reading and writing of structures

To be able to read or write PLC structures the memory layout of the structure or class in .NET must be the same as in the PLC. The layout of a structure or class can be specified with the attribute `StructLayoutAttribute`. The `LayoutKind` must be set to `LayoutKind.Sequential` and the `pack` must be set to 1

If arrays, strings or boolean values are define the class, one has to specify how these fields should be marshalled. This is accomplished with help of the `MarshalAs` attribute. Because arrays and strings do not have a fixed length in .NET, the property `SizeConst` is necessary for arrays and strings.

Marshalling values with 'ANY_TYPES' asynchronously

```
CancellationToken cancel = CancellationToken.None;

using (AdsClient client = new AdsClient())
{
    client.Connect(AmsNetId.Local, 851);

    // Bool value
    bool boolValue = false;
    ResultAnyValue resultBoolValue = await client.ReadValueAsync("MAIN.bool1", typeof(bool), cancel);
    boolValue = (bool)resultBoolValue.Value;
    ResultWrite resultWrite = await client.WriteValueAsync("MAIN.bool1", boolValue, cancel);

    // or
    ResultHandle resultHandleBool = await client.CreateVariableHandleAsync("MAIN.bool1", cancel); //
    BOOL
    //resultHandleBool.ThrowOnError(); // or

    if (resultHandleBool.Succeeded)
    {
        ResultAnyValue resultReadBool = await client.ReadAnyAsync(resultHandleBool.Handle, typeof(bo
ol), cancel);
        boolValue = (bool)resultReadBool.Value;
        ResultWrite resultWriteBool = await client.WriteAnyAsync(resultHandleBool.Handle, boolValue,
cancel);
        ResultAds resultHandleDeleteBool = await client.DeleteVariableHandleAsync(resultHandleBool.H
andle, cancel);
    }

    // RealValue
    ResultHandle resultHandleReal = await client.CreateVariableHandleAsync("MAIN.real1", cancel); //
    REAL

    if (resultHandleReal.Succeeded)
    {
        ResultAnyValue resultReadReal = await client.ReadAnyAsync(resultHandleReal.Handle, typeof(fl
oat), cancel); // REAL
        ResultWrite resultWriteReal = await client.WriteAnyAsync(resultHandleReal.Handle, resultRead
Real.Value, cancel);
        ResultAds resultHandleDeleteReal = await client.DeleteVariableHandleAsync(resultHandleReal.H
andle, cancel);
    }

    // String
    ResultHandle resultHandleString = await client.CreateVariableHandleAsync("MAIN.string1", cancel);
    // STRING[80]

    if (resultHandleString.Succeeded)
    {
        ResultAnyValue resultReadString = await client.ReadAnyAsync(resultHandleString.Handle, typeo
f(string), new int[] { 80 }, cancel); // Needs additional para for strlen
        ResultWrite resultWriteString = await client.WriteAnyAsync(resultHandleString.Handle, result
ReadString.Value, new int[] { 80 }, cancel);
        ResultAds resultHandleDeleteString = await client.DeleteVariableHandleAsync(resultHandleStri
ng.Handle, cancel);
    }

    // ushort[]
    ResultHandle resultHandleArray = await client.CreateVariableHandleAsync("MAIN.uint1Arr", cancel)
; // ARRAY [0..9] OF UINT
}
```



```

    if (resultHandleArray.Succeeded)
    {
        ResultAnyValue resultReadArray = await client.ReadAnyAsync(resultHandleArray.Handle, typeof(
ushort[]), new int[] { 10 }, cancel);
        ushort[] arrayValue = (ushort[])resultReadArray.Value;
        ResultWrite resultWriteArray = await client.WriteAnyAsync(resultHandleArray.Handle, arrayVal
ue, new int[] { 10 }, cancel);
        ResultAds resultHandleDeleteArray = await client.DeleteVariableHandleAsync(resultHandleArray
.Handle, cancel);
    }

    // Complex Struct Type
    // Take care the the corresponding .NET Type is blittable / marshallable to the PLC type
    ResultHandle resultHandleStruct = await client.CreateVariableHandleAsync("MAIN.struct",cancel);

    if (resultHandleStruct.Succeeded)
    {
        ResultAnyValue resultReadStruct = await client.ReadAnyAsync(resultHandleStruct.Handle, typeo
f(PlcStruct), cancel);
        PlcStruct structValue = (PlcStruct)resultReadStruct.Value;
        ResultWrite resultWriteStruct = await client.WriteAnyAsync(resultHandleStruct.Handle, struct
Value, cancel);
        ResultAds resultHandleDeleteStruct = await client.DeleteVariableHandleAsync(resultHandleStru
ct.Handle, cancel);
    }

    // ARRAY [0..9] OF STRING[80]
    // args[0] --> Number of Characters
    // args[1] --> Number of Array Elements
    // Needs additional para for strlen and number of Elements in Array
    ResultHandle resultHandleStringArray = await client.CreateVariableHandleAsync("MAIN.stringArr",c
ancel); // ARRAY [0..9] OF STRING[80]

    if (resultHandleStringArray.Succeeded)
    {
        ResultAnyValue resultReadStringArray = await client.ReadAnyAsync(resultHandleStringArray.Han
dle, typeof(string[]), new int[] { 80, 10 }, cancel);
        string[] stringArrValue = (string[])resultReadStringArray.Value;
        ResultWrite resultWriteStringArray = await client.WriteAnyAsync(resultHandleStringArray.Hand
le, stringArrValue, new int[] { 80, 10 }, cancel);
        ResultAds resultHandleDeleteStringArray = await client.DeleteVariableHandleAsync(resultHandl
eStringArray.Handle, cancel);
    }
}

```

Defining Memory layout of struct type.

```

// Attention: Dependent of the System where the PLC runs, the StructLayout of the exchanged
// Structures must match. With the ANY_TYPE concept this is realized with 'blittable' objects,
// that match on .NET and PLC side.

// Default Pack Modes:

// TC3 I64/x86: Normal, in this case Pack = 8
// TC2 x86:      Pack = 1

// On TC3 PLC side we can force the packing of structures with the attribute
// {attribute 'pack_mode' := '1'}, see also 'pack_mode' attribute in Beckhoff InfoSystem
// For TC2 is the Pack setting Pack = 1 the only possible way, because it is not selectable.

// We have to ensure that the pack mode on both sides is equal!

[StructLayout(LayoutKind.Sequential, Pack = 1, CharSet = CharSet.Ansi)]
public struct PlcStruct
{
    // Type must be 'blittable' to the corresponding PLC Struct Type
    // See MSDN for MarshalAs and Default Marshalling.

    [MarshalAs(UnmanagedType.I1)]
    public bool boolVal; // BOOL
    public byte byteVal; // BYTE
    public ushort ushortVal; // UINT
    public short shortVal; // INT
    public uint uintVal; // UDINT
    public int dintVal; // DINT
    public uint udintVal; // UDINT
    public float realVal; // REAL
    public double lrealVal; // LREAL
    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = 81)]

```

```

public string stringVal; // STRING[80]

[MarshalAs(UnmanagedType.U4)]
public uint timeVal; // TIME
[MarshalAs(UnmanagedType.U4)]
public uint todVal; // TOD
[MarshalAs(UnmanagedType.U4)]
public uint dateVal; // DATE
[MarshalAs(UnmanagedType.U4)]
public uint dtVal; // DT
}

```

Marshalling values with 'ANY_TYPES' (synchronous)

```

using (AdsClient client = new AdsClient())
{
    client.Connect(AmsNetId.Local, 851);

    // Bool value
    bool boolValue = (bool)client.ReadValue("MAIN.bool1", typeof(bool));
    client.WriteValue("MAIN.bool1", boolValue);

    // or
    uint handleBool = client.CreateVariableHandle("MAIN.bool1"); // BOOL
    boolValue = (bool)client.ReadAny(handleBool, typeof(bool));
    client.WriteAny(handleBool, boolValue);
    client.DeleteVariableHandle(handleBool);

    // RealValue
    uint handleReal = client.CreateVariableHandle("MAIN.real1"); // REAL
    float realValue = (float)client.ReadAny(handleReal, typeof(float));
    client.WriteAny(handleReal, realValue);
    client.DeleteVariableHandle(handleReal);

    // String
    uint handleString = client.CreateVariableHandle("MAIN.string1"); // STRING[80]
    string stringValue = (string)client.ReadAny(handleString, typeof(string), new int[] { 80 }); //
Needs additional para for strlen
    client.WriteAny(handleString, stringValue, new int[] { 80 });
    client.DeleteVariableHandle(handleString);

    // ushort[]
    uint handleArray = client.CreateVariableHandle("MAIN.uint1Arr"); // ARRAY [0..9] OF UINT
    ushort[] arrayValue = (ushort[])client.ReadAny(handleArray, typeof(ushort[]), new int[] { 10 });
    client.WriteAny(handleArray, arrayValue, new int[] { 10 });
    client.DeleteVariableHandle(handleArray);

    // Complex Struct Type
    // Take care the the corresponding .NET Type is blittable / marshallable to the PLC type
    uint handleStruct = client.CreateVariableHandle("MAIN.struct");
    PlcStruct structValue = (PlcStruct)client.ReadAny(handleStruct, typeof(PlcStruct));
    client.WriteAny(handleStruct, structValue);
    client.DeleteVariableHandle(handleStruct);

    // ARRAY [0..9] OF STRING[80]
    // args[0] --> Number of Characters
    // args[1] --> Number of Array Elements
    // Needs additional para for strlen and number of Elements in Array
    uint handleStringArr = client.CreateVariableHandle("MAIN.stringArr"); // ARRAY [0..9] OF STRING[
80]
    string[] stringArr = (string[])client.ReadAny(handleStringArr, typeof(string[]), new int[] { 80,
10 });
    client.WriteAny(handleStringArr, stringArr, new int[] { 80, 10 });
    client.DeleteVariableHandle(handleStringArr);
}

```

ADS Notifications with Type marshalling (AdsNotificationEx)

The method `AddDeviceNotificationEx` is used to register notifications for a PLC variable. If the value of a variable changes the event `AdsNotificationEx` is fired. The difference to the event `AdsNotification`, is that the value of the variable is stored in an object instead of in an `AdsStream`. Therefore one has to pass the type of the object to the method `AddDeviceNotificationEx`

Notifications with 'ANY_TYPES' (asynchronous)

```

CancellationToken cancel = CancellationTokens.None;

using (AdsClient client = new AdsClient())
{
    client.AdsNotificationEx += Client_AdsNotificationEx;
    client.Connect(AmsNetId.Local, 851);

    // Add UDINT
    ResultHandle resultHandle = await client.AddDeviceNotificationExAsync("MAIN.udint", new NotificationSettings(AdsTransMode.OnChange, 200, 200), null, typeof(uint), null, cancel);
    await Task.Delay(5000, cancel); // Wait ...
    ResultAds resultHandleDelete = await client.DeleteDeviceNotificationAsync(resultHandle.Handle, cancel); // Unregister Event
}

```

Notifications with 'ANY_TYPES'

```

private void Client_AdsNotificationEx(object sender, AdsNotificationExEventArgs e)
{
    uint value = (uint)e.Value; // Marshalled value as .NET Type
}

```

Notifications with 'ANY_TYPES' (synchronous)

```

using (AdsClient client = new AdsClient())
{
    client.AdsNotificationEx += Client_AdsNotificationEx;
    client.Connect(AmsNetId.Local, 851);

    // Add UDINT
    uint notificationHandle = client.AddDeviceNotificationEx("MAIN.udint", new NotificationSettings(AdsTransMode.OnChange, 200, 200), null, typeof(uint));
    Thread.Sleep(5000); // ...
    client.DeleteDeviceNotification(notificationHandle); // Unregister Event
}

```

4.7 Access Data via Symbol Loader

Some ADS Devices (e.g. the TwinCAT PLC) provide symbolic information for download. That means all visible Symbols and DataTypes can be retrieved from the target system. While this needs an extra effort to upload and hold the data, this feature helps to remove the dependency of the code/configuration running on the target device.

E.g. because the symbolic information can now be browsed and determined during runtime, the application can be written without knowing what's running on the target system. Even more having the Symbol information cached, the access of the process image data will be easier because the datasize and access (instance path) is stored in the symbol.

Dependent how it is parametrized, the symbol loader can work with 'ANY_TYPES' (marshallable Primitive types, [Value marshalling with ANYTYPE concept \[► 31\]](#)) or full dynamic symbols ([Automatic dynamic marshalling of values \[► 36\]](#)).

Example**Accessing symbolic data by preloaded Symbolic information (asynchronous)**

```

using (AdsClient client = new AdsClient())
{
    CancellationToken cancel = CancellationTokens.None;

    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);

    // Load all Symbols + DataTypes
    ISymbolLoader loader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);

    ResultSymbols resultSymbols = await loader.GetSymbolsAsync(cancel);

    if (resultSymbols.Succeeded)

```

```

    {
        Symbol symbol = (Symbol)resultSymbols.Symbols["MAIN.nCounter"];

        // Works for ALL Primitive 'ANY TYPES' Symbols
        ResultWriteAccess resultWrite = await symbol.WriteValueAsync(valueToWrite, cancel);
        ResultReadValueAccess resultRead = await symbol.ReadValueAsync(cancel);

        if (resultRead.Succeeded)
            valueToRead = (uint)resultRead.Value;

        // Simple filtering of Symbols
        Regex filterExpression = new Regex(pattern: @"^MAIN.*"); // Everything that starts with "MAIN"

        // FilterFunction that filters for the InstancePath
        Func<ISymbol, bool> filter = s => filterExpression.IsMatch(s.InstancePath);
        SymbolIterator iterator = new SymbolIterator(symbols: resultSymbols.Symbols, recurse: true,
selector: filter);

        foreach (ISymbol filteredSymbol in iterator)
        {
            Console.WriteLine(filteredSymbol.InstancePath);
        }
    }
}

```

Accessing symbolic data by preloaded Symbolic information (synchronous)

```

using (AdsClient client = new AdsClient())
{
    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);

    // Load all Symbols + DataTypes
    ISymbolLoader loader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    Symbol symbol = (Symbol)loader.Symbols["MAIN.nCounter"];

    // Works for ALL Primitive 'ANY TYPES' Symbols
    symbol.WriteValue(valueToWrite);
    valueToRead = (uint)symbol.ReadValue();

    // Simple filtering of Symbols
    Regex filterExpression = new Regex(pattern: @"^MAIN.*"); // Everything that starts with "MAIN"

    // FilterFunction that filters for the InstancePath
    Func<ISymbol, bool> filter = s => filterExpression.IsMatch(s.InstancePath);
    SymbolIterator iterator = new SymbolIterator(symbols: loader.Symbols, recurse: true, selector: filter);

    foreach (ISymbol filteredSymbol in iterator)
    {
        Console.WriteLine(filteredSymbol.InstancePath);
    }
}

```

4.8 Automatic dynamic marshalling of values

The 'Dynamic Symbol Loader' of the .NET ADS Communication API makes use of the .NET dynamic language runtime (DLR). The dynamic language runtime is a runtime environment that adds a set of services for dynamic languages to the common language runtime (CLR). The DLR makes it easier to develop dynamic languages to run on the .NET Framework and to add dynamic features to statically typed languages.

Dynamic languages can identify the type of an object at run time, whereas in statically typed languages (without using the 'dynamic' keyword, specify object types at design time. The advantage here is - from the moment on the symbolic (and dataType) information is available from the ADS Device - Symbol/Variable values can be marshalled 'on-the-fly' during runtime in a type-safe manner.

This works not only with primitive types but also with complex types. This reduces the complexity of the written application code to access the values, because neither the type of the data must be known, nor how the value data must be marshalled from/to the process image. The price to be paid is simply that the full symbolic information and data types must be downloaded from the ADS Device by the symbol loader.

Example

Automatic marshalling values with 'Dynamic Values' (asynchronous)

```
using (AdsClient client = new AdsClient())
{
    CancellationToken cancel = CancellationToken.None;

    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);

    // Load all Symbols + DataTypes
    // Primitive Parts will be automatically resolved to .NET Primitive types.
    IDynamicSymbolLoader loader = (IDynamicSymbolLoader)SymbolLoaderFactory.Create(client, SymbolLoaderSettings.DefaultDynamic);

    //dynamic symbols = loader.SymbolsDynamic;
    ResultDynamicSymbols resultSymbols = await loader.GetDynamicSymbolsAsync(cancel);

    if (resultSymbols.Succeeded)
    {
        dynamic symbols = resultSymbols.Symbols;
        dynamic main = symbols.Main;

        // Use typed object to use InfoTips
        DynamicSymbol nCounter = main.nCounter; // UDINT

        // or to be fullDynamic
        dynamic nCounter2 = main.nCounter;

        // Works for ALL sorts of types (not restricted to ANY_TYPE basing primitive types)
        ResultReadValueAccess resultRead = await nCounter.ReadValueAsync(cancel);

        if (resultRead.Succeeded)
        {
            valueToRead = (uint)resultRead.Value;

            // or
            var varValue = resultRead.Value;
            // or
            dynamic dynValue = resultRead.Value;
        }
        // Same for writing
        ResultWriteAccess resultWrite = await nCounter.WriteValueAsync(valueToWrite, cancel);

        // Or Notifications / Events (typed dynamically)
        nCounter.ValueChanged += NCounter_ValueChanged;

        //Reading complexTypes e.g. Struct

        DynamicSymbol myStructSymbol = main.plcStruct; // Dynamically created
        ResultReadValueAccess resultRead2 = await myStructSymbol.ReadValueAsync(cancel); // Takes an ADS Snapshot of the value

        if (resultRead2.Succeeded)
        {
            dynamic myStructVal = resultRead2.Value;

            dynamic int1Val = myStructVal.int1; // Value to an INT (short)
            dynamic valueNestedStruct = myStructVal.nestedStruct; //
value to another complex type (here a nested Struct)
        }
        myStructSymbol.ValueChanged += MyStructSymbol_ValueChanged;
        //wait 5 seconds to get some events
        Thread.Sleep(5000);
    }
}
```

Automatic marshalling values with 'Dynamic Values' (handler)

```
private void NCounter_ValueChanged(object sender, ValueChangedEventArgs e)
{
    var uintVal = e.Value;
}

private void MyStructSymbol_ValueChanged(object sender, ValueChangedEventArgs e)
```

```
{
    dynamic structValue = e.Value; // Snapshot of the whole Struct and all its contents
}
```

Automatic marshalling values with 'Dynamic Values' (synchronous)

```
using (AdsClient client = new AdsClient())
{
    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);

    // Load all Symbols + DataTypes
    // Primitive Parts will be automatically resolved to .NET Primitive types.
    IDynamicSymbolLoader loader = (IDynamicSymbolLoader)SymbolLoaderFactory.Create(client, SymbolLoaderSettings.DefaultDynamic);

    dynamic symbols = loader.SymbolsDynamic;
    dynamic main = symbols.Main;

    // Use typed object to use InfoTips
    DynamicSymbol nCounter = main.nCounter; // UDINT

    // or to be fullDynamic
    //dynamic nCounter = main.nCounter;

    // Works for ALL sorts of types (not restricted to ANY_TYPE basing primitive types)
    valueToRead = (uint)nCounter.ReadValue();
    // or
    var varValue = nCounter.ReadValue();
    // or
    dynamic dynValue = nCounter.ReadValue();

    // Same for writing
    nCounter.WriteValue(valueToWrite);

    // Or Notifications / Events
    nCounter.ValueChanged += new EventHandler<ValueChangedEventArgs>(NCounter_ValueChanged);

    //Reading complexTypes e.g. Struct

    DynamicSymbol myStructSymbol = main.plcStruct; // Dynamically created
    dynamic myStructVal = myStructSymbol.ReadValue(); // Takes an ADS Snapshot of the value

    dynamic int1Val = myStructVal.int1; // Value to an INT (short)
    dynamic valueNestedStruct = myStructVal.nestedStruct; //
    value to another complex type (here a nested Struct)

    myStructSymbol.ValueChanged += new EventHandler<ValueChangedEventArgs>(MyStructSymbol_ValueChanged);
    //wait for notifications for 5 seconds
    Thread.Sleep(5000);
}
```

Calling 'ReadValue'/'ReadValueAsync' or the incoming 'ValueChanged' notification takes a full snapshot (with snapshot time) of the value. That means, when for example subelements of a struct value will be accessed, all subvalues will represent the value of that snapshot time consistently. The starting point (or the instance that caches the consistent data) is always the 'DynamicSymbol' object that called 'ReadValue'.

An update of the value can be done directly on the value with 'UpdateValue', or with reading a new Value on the 'DynamicSymbol' ('ReadValue').

The 'ValueChanged' event on the 'DynamicSymbol' assigns a Notification for just this symbol. The 'ValueChanged' handler will contain the value completely marshalled as dynamic object.

4.9 Reactive Extensions and ADS

The Reactive Extensions (Rx) is a .NET library for composing asynchronous and event-based programs using observable sequences and LINQ-style query operators. Using Rx, developers represent asynchronous data streams with Observables, query asynchronous data streams using LINQ operators, and parameterize the concurrency in the asynchronous data streams using Schedulers. Simply put, Rx = Observables + LINQ + Schedulers.

In ADS terms, not only the reading and writing data or symbol values can be put into reactive data streams, also ADS Notifications are a perfect fit for reactive code. This eases not only data binding to reactive frameworks (e.g. reactive UI) but also supports enhanced data manipulation via synchronous and asynchronous observers. Multithreaded and parallelized code paths that support multiple CPU cores can be written very easily without the burden of deadlock and synchronization issues.

More about .NET reactive extensions can be read here: [Reactive extensions project site](#).

The TwinCAT ADS Reactive extensions are available via a supplement Nuget Package: [Beckhoff.TwinCAT.Ads.Reactive Nuget Package](#)

Example

Observe for Notifications

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Reactive Notification Handler
    var valueObserver = Observer.Create<ushort>(val =>
    {
        Console.WriteLine(string.Format("Value: {0}", val.ToString()));
    }
    );

    // Turning ADS Notifications into sequences of Value Objects (Taking 20 Values)
    // and subscribe to them.
    IDisposable subscription = client.WhenNotification<ushort>("TwinCAT_SystemInfoVarList._TaskInfo.CycleCount", NotificationSettings.Default).Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Observe for Symbol Notifications

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];

    // Reactive Notification Handler
    var valueObserver = Observer.Create<object>(val =>
    {
        Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToString()));
    }
    );

    cycleCount.NotificationSettings = new NotificationSettings(AdsTransMode.OnChange, 500, 5000); // optional: Change NotificationSettings on Symbol

    // Turning ADS Notifications into sequences of Value Objects (Taking 20 Values)
    // and subscribe to them.
    IDisposable subscription = cycleCount.WhenValueChanged().Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Observer for dynamic Symbol Notifications

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));
```

```

    // Create Symbol information
    var symbolLoader = (IDynamicSymbolLoader)SymbolLoaderFactory.Create(client, SymbolLoaderSettings
.DefaultDynamic);
    dynamic symbols = symbolLoader.SymbolsDynamic;
    dynamic cycleCount = symbols.TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount;

    // Reactive Notification Handler
    var valueObserver = Observer.Create<object>(val =>
    {
        // Value objects can be dynamically (on the fly) created objects here (e.g. structs)
        Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.To
String()));
    }
    );

    cycleCount.NotificationSettings = new NotificationSettings(AdsTransMode.OnChange, 500, 5000); //
optional: Change NotificationSettings on Symbol

    // Turning ADS Notifications into sequences of Value Objects (Taking 20 Values)
    // and subscribe to them.
    // We have to give the 'hint' about IValueSymbol here, that the CLR finds the Extension Method '
WhenValueChanged' during runtime.
    IDisposable subscription = ((IValueSymbol)cycleCount).WhenValueChanged().Take(20).Subscribe(valu
eObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}

```

Polling observer

```

// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInf
o.CycleCount"];

    // Reactive Notification Handler
    var valueObserver = Observer.Create<object>(val =>
    {
        Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.To
String()));
    }
    );

    // Take 20 Values in an Interval of 500ms
    IDisposable subscription = cycleCount.PollValues(TimeSpan.FromMilliseconds(500)).Take(20).Subscr
ibe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}

```

Writing values with observable subject

```

using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information (Symbol 'i : INT' in PLC Global Variables list.
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol gvlIntSymbol = (IValueSymbol)symbolLoader.Symbols["GVL.i"];

    // Produces object (short) Values 0,1,2,3 ... in seconds period
    IObservable<object> timerObservable = Observable.Interval(TimeSpan.FromSeconds(1.0)).Select(i =>
(object)(short)i);

    // Take 10 Values (0..9) and write them to GVL.i
    IDisposable dispose = gvlIntSymbol.WriteValues(timerObservable.Take(10));
}

```



```
Console.ReadKey(); // Wait for Key press  
dispose.Dispose(); // Dispose the Subscription  
}
```

5 HowTo Samples

The topics in this section describe the various changes made to the 'TwinCAT.Ads .NET API' over the life of the project.

HowTo

How to use the .NET TwinCAT API

Sample	Description
Reading/writing (simple) values from/to ADS Servers.	Read/Write values [▶ 42]
Reading/writing string values.	Read/Write string values [▶ 46]
Event driven reading (ADS Notifications)	Event driven reading [▶ 52]
Read/Write data from/to ADS Servers using reactive extensions	Reactive Read/Write [▶ 54]
Upgrading existing ADS Application code (Version 4.X --> 6.X)	Upgrade 4.X --> 6.X [▶ 56]

Other Resources

[TwinCAT ADS .NET API Documentation \[▶ 16\]](#)

5.1 Read/Write primitive values

Reading writing Values from ADS Devices is the most essential part of the communication API. There are several options for communication with your application.

- Accessing by IndexGroup / IndexOffset
- Symbolic access by instance path and optionally use handles for the symbol
- Holding the overall Symbolic information in the SymbolLoader and use easy access via symbol objects.
- Usage the symbolic interface ITcAdsSymbol when complete SymbolLoading by SymbolLoaders is not appropriate.
- Reading / Writing values as .NET managed Types (primitive types or compound primitive types called ANY_TYPES), or complex dynamic types typesafe generated at runtime.

The following section shows the different scenarios as code snippets.

HowTo Read/Write Values

Read/Write AnyType by IndexGroup/IndexOffset (asynchronously)

```
using (AdsClient client = new AdsClient())
{
    CancellationToken cancel = CancellationToken.None;

    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);
    ResultWrite resultWrite = await client.WriteAnyAsync(0x4020, 0x0, valueToWrite, cancel);
    bool succeeded = resultWrite.Succeeded;

    ResultValue<uint> resultRead = await client.ReadAnyAsync<uint>(0x4020, 0x0, cancel);

    if (resultRead.Succeeded)
    {
```

```

    valueToRead = (uint)resultRead.Value;
}
}

```

Read/Write AnyType by IndexGroup/IndexOffset (synchronously)

```

using (AdsClient client = new AdsClient())
{
    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);
    client.WriteAny(0x4020, 0x0, valueToWrite);
    valueToRead = (uint)client.ReadAny(0x4020, 0x0, typeof(uint));
}

```

Read/Write AnyType by variable handle (asynchronously)

```

using (AdsClient client = new AdsClient())
{
    CancellationToken cancel = CancellationToken.None;
    uint varHandle = 0;
    client.Connect(AmsNetId.Local, 851);

    uint valueToRead = 0;
    uint valueToWrite = 42;

    ResultHandle resultHandle = await client.CreateVariableHandleAsync("MAIN.nCounter", cancel);
    varHandle = resultHandle.Handle;

    if (resultHandle.Succeeded)
    {
        try
        {
            ResultWrite resultWrite = await client.WriteAnyAsync(varHandle, valueToWrite, cancel);
            ResultValue<uint> resultRead = await client.ReadAnyAsync<uint>(varHandle, cancel);

            if (resultRead.Succeeded)
                valueToRead = resultRead.Value;
        }
        finally
        {
            // Unregister VarHandle after Use
            ResultAds result = await client.DeleteVariableHandleAsync(varHandle, cancel);
        }
    }
}

```

Read/Write AnyType by variable handle (synchronously)

```

using (AdsClient client = new AdsClient())
{
    uint varHandle = 0;
    client.Connect(AmsNetId.Local, 851);
    try
    {
        uint valueToRead = 0;
        uint valueToWrite = 42;

        varHandle = client.CreateVariableHandle("MAIN.nCounter");
        client.WriteAny(varHandle, valueToWrite);
        valueToRead = (uint)client.ReadAny(varHandle, typeof(uint));
    }
    finally
    {
        // Unregister VarHandle after Use
        client.DeleteVariableHandle(varHandle);
    }
}

```

Read/Write AnyType by instance/symbol path (asynchronously)

```

using (AdsClient client = new AdsClient())
{
    CancellationToken cancel = CancellationToken.None;
    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);
}

```

```

ResultWrite resultWrite = await client.WriteValueAsync("MAIN.nCounter", valueToWrite, cancel);
ResultValue<uint> resultRead = await client.ReadValueAsync<uint>("MAIN.nCounter",cancel);

if (resultRead.Succeeded)
    valueToRead = resultRead.Value;
}

```

Read/Write AnyType by instance/symbol path (synchronously)

```

using (AdsClient client = new AdsClient())
{
    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);
    client.WriteValue("MAIN.nCounter", valueToWrite);
    valueToRead = (uint)client.ReadValue("MAIN.nCounter", typeof(uint));
}

```

Read/Write AnyType by IAdsSymbol (asynchronously)

```

using (AdsClient client = new AdsClient())
{
    CancellationToken cancel = CancellationToken.None;

    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);

    ResultValue<IAdsSymbol> resultSymbol = await client.ReadSymbolAsync("MAIN.nCounter",cancel);

    if (resultSymbol.Succeeded)
    {
        ResultWrite resultWrite = await client.WriteValueAsync(resultSymbol.Value, valueToWrite, cancel)
;
        bool succeeded = resultWrite.Succeeded;

        ResultValue<uint> resultValue = await client.ReadValueAsync<uint>(resultSymbol.Value, cancel);

        if (resultValue.Succeeded)
            valueToRead = resultValue.Value;
    }
}

```

Read/Write AnyType by IAdsSymbol (synchronously)

```

using (AdsClient client = new AdsClient())
{
    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);

    IAdsSymbol symbol = client.ReadSymbol("MAIN.nCounter");
    client.WriteValue(symbol, valueToWrite);
    valueToRead = (uint)client.ReadValue(symbol);
}

```

Read/Write AnyType by SymbolBrowser (asynchronously)

```

using (AdsClient client = new AdsClient())
{
    CancellationToken cancel = CancellationToken.None;

    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);

    // Load all Symbols + DataTypes
    ISymbolLoader loader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);

    ResultSymbols resultSymbols = await loader.GetSymbolsAsync(cancel);

    if (resultSymbols.Succeeded)
    {
        Symbol symbol = (Symbol)resultSymbols.Symbols["MAIN.nCounter"];
    }
}

```

```

// Works for ALL Primitive 'ANY TYPES' Symbols
ResultWriteAccess resultWrite = await symbol.WriteValueAsync(valueToWrite, cancel);
ResultReadValueAccess resultRead = await symbol.ReadValueAsync(cancel);

if (resultRead.Succeeded)
    valueToRead = (uint)resultRead.Value;

// Simple filtering of Symbols
Regex filterExpression = new Regex(pattern: @"^MAIN.*"); // Everything that starts with "MAIN"

// FilterFunction that filters for the InstancePath
Func<ISymbol, bool> filter = s => filterExpression.IsMatch(s.InstancePath);
SymbolIterator iterator = new SymbolIterator(symbols: resultSymbols.Symbols, recurse: true, selector: filter);

foreach (ISymbol filteredSymbol in iterator)
{
    Console.WriteLine(filteredSymbol.InstancePath);
}
}

```

Read/Write AnyType by SymbolBrowser (synchronously)

```

using (AdsClient client = new AdsClient())
{
    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);

    // Load all Symbols + DataTypes
    ISymbolLoader loader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    Symbol symbol = (Symbol)loader.Symbols["MAIN.nCounter"];

    // Works for ALL Primitive 'ANY TYPES' Symbols
    symbol.WriteValue(valueToWrite);
    valueToRead = (uint)symbol.ReadValue();

    // Simple filtering of Symbols
    Regex filterExpression = new Regex(pattern: @"^MAIN.*"); // Everything that starts with "MAIN"

    // FilterFunction that filters for the InstancePath
    Func<ISymbol, bool> filter = s => filterExpression.IsMatch(s.InstancePath);
    SymbolIterator iterator = new SymbolIterator(symbols: loader.Symbols, recurse: true, selector: filter);

    foreach (ISymbol filteredSymbol in iterator)
    {
        Console.WriteLine(filteredSymbol.InstancePath);
    }
}

```

Read/Write dynamic types by SymbolBrowser (asynchronously)

```

using (AdsClient client = new AdsClient())
{
    CancellationToken cancel = CancellationToken.None;

    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);

    // Load all Symbols + DataTypes
    // Primitive Parts will be automatically resolved to .NET Primitive types.
    IDynamicSymbolLoader loader = (IDynamicSymbolLoader)SymbolLoaderFactory.Create(client, SymbolLoaderSettings.DefaultDynamic);

    //dynamic symbols = loader.SymbolsDynamic;
    ResultDynamicSymbols resultSymbols = await loader.GetDynamicSymbolsAsync(cancel);
    if (resultSymbols.Succeeded)
    {
        dynamic symbols = resultSymbols.Symbols; // Symbols collection with 'dynamic' access.
        dynamic main = symbols.Main; // Get the 'dynamic' main FB as Property

        // Use typed object to use InfoTips instead of 'dynamic'
        DynamicSymbol nCounter = main.nCounter;
    }
}

```

```
// or to be fullDynamic
dynamic nCounter2 = main.nCounter;

// Works for ALL sorts of types (not restricted to ANY_TYPE basing primitive types)
ResultWriteAccess resultWrite = await nCounter.WriteValueAsync(valueToWrite, cancel);
ResultReadValueAccess resultRead = await nCounter.ReadValueAsync(cancel);

if (resultRead.Succeeded)
{
    // Because the PLC value is defined as UDINT (32-Bit)
    // We get back an already Marshalled UInt32 here ...
    valueToRead = (uint)resultRead.Value;
}
}
```

Read/Write dynamic types by SymbolBrowser (synchronously)

```
using (AdsClient client = new AdsClient())
{
    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);

    // Load all Symbols + DataTypes
    // Primitive Parts will be automatically resolved to .NET Primitive types.
    IDynamicSymbolLoader loader = (IDynamicSymbolLoader)SymbolLoaderFactory.Create(client, SymbolLoaderSettings.DefaultDynamic);

    dynamic symbols = loader.SymbolsDynamic;
    dynamic main = symbols.Main;

    // Use typed object to use InfoTips
    DynamicSymbol nCounter = main.nCounter;

    // or to be fullDynamic
    //dynamic nCounter = main.nCounter;

    // Works for ALL sorts of types (not restricted to ANY_TYPE basing primitive types)
    nCounter.WriteValue(valueToWrite);
    valueToRead = (uint)nCounter.ReadValue();
}
}
```

5.2 Read/Write string types

ADS Server usually support strings in 2 flavors. The Default (ANSI) and the Unicode encoding (STRING vs. WSTRING) The ANSI encoding reserves 1 byte per character. Unicode reserves 2.

The strings are of fixed size and therefore the length of the the reserved space within the process image is important.

HowTo Read/Write string values

Reading writing ANSI Streams:

Read/Write ANSI Strings (async)

```
CancellationToken cancel = CancellationToken.None;

using (AdsClient client = new AdsClient())
{
    client.Connect(AmsNetId.Local, 851); // Connect to local port 851 (PLC)

    ResultHandle resultHandle = await client.CreateVariableHandleAsync("MAIN.string", cancel); // Symbol "string" in MAIN defined as string

    if (resultHandle.Succeeded)
    {
        try
        {
            // Read ANSI String string[80]
            int byteSize = 81; // Size of 80 ANSI chars + /0 (STRING[80])
            PrimitiveTypeMarshaler converter = new PrimitiveTypeMarshaler(StringMarshaler.DefaultEncoding);
        }
    }
}
}
```

```

        byte[] buffer = new byte[byteSize];

        ResultRead resultRead = await client.ReadAsync(resultHandle.Handle, buffer.AsMemory(), cancel);

        if (resultRead.Succeeded)
        {
            string value = null;
            converter.Unmarshal<string>(buffer.AsSpan(), out value);

            byte[] writeBuffer = new byte[byteSize];
            // Write ANSI String string[80]
            value = "Changed";
            converter.Marshal(value, writeBuffer);
            ResultWrite resultWrite = await client.WriteAsync(resultHandle.Handle, writeBuffer, cancel);
        }
    }
    finally
    {
        ResultAds r1 = await client.DeleteVariableHandleAsync(resultHandle.Handle, cancel);
    }
}

```

Read/Write ANSI Strings (sync)

```

using (AdsClient client = new AdsClient())
{
    client.Connect(AmsNetId.Local, 851); // Connect to local port 851 (PLC)
    uint handle = client.CreateVariableHandle("MAIN.string"); // Symbol "string" in MAIN defined as string

    try
    {
        // Read ANSI String string[80]
        int byteSize = 81; // Size of 80 ANSI chars + /0 (STRING[80])
        PrimitiveTypeMarshaler converter = new PrimitiveTypeMarshaler(StringMarshaler.DefaultEncoding);
        byte[] buffer = new byte[byteSize];

        int readBytes = client.Read(handle, buffer.AsMemory());

        string value = null;
        converter.Unmarshal<string>(buffer.AsSpan(), out value);

        // Write ANSI String string[80]
        byte[] writeBuffer = new byte[byteSize];
        value = "Changed";
        converter.Marshal(value, writeBuffer);
        client.Write(handle, writeBuffer);
    }
    finally
    {
        client.DeleteVariableHandle(handle);
    }
}

```

Reading writing UNICODE Streams:

Read/Write Unicode Strings (async)

```

CancellationToken cancel = CancellationToken.None;

using (AdsClient client = new AdsClient())
{
    client.Connect(AmsNetId.Local, 851); // Connect to local port 851 (PLC)
    ResultHandle resultHandle = await client.CreateVariableHandleAsync("MAIN.wstring", cancel); // Symbol "wstring" defined in MAIN as WSTRING

    if (resultHandle.Succeeded)
    {
        try
        {
            // Read UNICODE String wstring[80]
            PrimitiveTypeMarshaler converter = new PrimitiveTypeMarshaler(Encoding.Unicode);
            int byteSize = converter.MarshalSize(80); // Size of 80 UNICODE chars + /0 (WSTRING[80]) (162)
            byte[] readBuffer = new byte[byteSize];

            ResultRead resultRead = await client.ReadAsync(resultHandle.Handle, readBuffer, cancel);
        }
    }
}

```

```

        if (resultRead.Succeeded)
        {
            string value = null;
            converter.Unmarshal(readBuffer.AsSpan(), out value);

            // Write Unicode String string[80]
            value = "Changed";
            byte[] writeBuffer = new byte[byteSize];
            converter.Marshal(value, writeBuffer.AsSpan());

            ResultWrite resultWrite = await client.WriteAsync(resultHandle.Handle, writeBuffer.AsMemory(
), cancel);
        }
    }
    finally
    {
        {
            ResultAds r1 = await client.DeleteVariableHandleAsync(resultHandle.Handle, cancel);
        }
    }
}

```

Read/Write Unicode Strings (sync)

```

using (AdsClient client = new AdsClient())
{
    client.Connect(AmsNetId.Local, 851); // Connect to local port 851 (PLC)
    uint handle = client.CreateVariableHandle("MAIN.wstring"); // Symbol "wstring" defined in MAIN as WSTRING

    try
    {
        // Read UNICODE String wstring[80]

        PrimitiveTypeMarshaler converter = new PrimitiveTypeMarshaler(Encoding.Unicode);
        int byteSize = converter.MarshalSize(80); // Size of 80 UNICODE chars + /0 (WSTRING[80]) (162)
        byte[] readBuffer = new byte[byteSize];

        int readBytes = client.Read(handle, readBuffer);

        string value = null;
        converter.Unmarshal(readBuffer.AsSpan(), out value);

        // Write Unicode String string[80]
        value = "Changed";
        byte[] writeBuffer = new byte[byteSize];
        converter.Marshal(value, writeBuffer.AsSpan());

        client.Write(handle, writeBuffer.AsMemory());
    }
    finally
    {
        client.DeleteVariableHandle(handle);
    }
}

```

Reading writing strings with ReadAny/WriteAny group of methods:

Read/Write Anystring (async)

```

Cancellation token cancel = CancellationToken.None;

using (AdsClient client = new AdsClient())
{
    client.Connect(AmsNetId.Local, 851); // Connect to local port 851 (PLC)

    ResultHandle resultHandleStr = await client.CreateVariableHandleAsync("MAIN.string", cancel); //
Symbol "string" defined in MAIN as STRING
    ResultHandle resultHandleWStr = await client.CreateVariableHandleAsync("MAIN.wstring", cancel); /
/ Symbol "wstring" defined in MAIN as WSTRING

    if (resultHandleStr.Succeeded && resultHandleWStr.Succeeded)
    {
        try
        {
            ResultAnyValue resultReadStr = await client.ReadAnyStringAsync(resultHandleStr.Handle, 80, S
tringMarshaler.DefaultEncoding, cancel);
            ResultAnyValue resultReadWStr = await client.ReadAnyStringAsync(resultHandleWStr.Handle, 80,
Encoding.Unicode, cancel);

```



```

        string changedValue = "Changed";

        // Attention, take care that the memory of the string in the process image is not exceeded!
        ResultWrite resultWriteStr = await client.WriteAnyStringAsync(resultHandleStr.Handle, change
dValue, 80, StringMarshaler.DefaultEncoding, cancel);
        ResultWrite resultWriteWStr = await client.WriteAnyStringAsync(resultHandleWStr.Handle, chan
gedValue, 80, Encoding.Unicode, cancel);
    }
    finally
    {
        ResultAds r1 = await client.DeleteVariableHandleAsync(resultHandleStr.Handle, cancel);
        ResultAds r2 = await client.DeleteVariableHandleAsync(resultHandleWStr.Handle, cancel);
    }
}
}

```

Read/Write Anystring (sync)

```

using (AdsClient client = new AdsClient())
{
    client.Connect(AmsNetId.Local, 851); // Connect to local port 851 (PLC)

    uint stringHandle = client.CreateVariableHandle("MAIN.string"); // Symbol "string" defined in MA
IN as STRING
    uint wStringHandle = client.CreateVariableHandle("MAIN.wstring"); // Symbol "string" defined in
MAIN as WSTRING

    try
    {
        string str = client.ReadAnyString(stringHandle, 80, StringMarshaler.DefaultEncoding);
        string wStr = client.ReadAnyString(wStringHandle, 80, Encoding.Unicode);

        string changedValue = "Changed";

        // Attention, take care that the memory of the string in the process image is not exceeded!
        client.WriteAnyString(stringHandle, changedValue, 80, StringMarshaler.DefaultEncoding);
        client.WriteAnyString(wStringHandle, changedValue, 80, Encoding.Unicode);
    }
    finally
    {
        client.DeleteVariableHandle(stringHandle);
        client.DeleteVariableHandle(wStringHandle);
    }
}

```

5.3 Read/Write PlcOpen types (DATE, TIME ...)

The following PLCOpen types have specific representations within the TwinCAT.Ads Communication Library:
[DT \[▶ 2034\]](#)

- [DT \[▶ 2034\]](#)
- [DATE \[▶ 2016\]](#)
- [TIME \[▶ 2061\]](#)
- [LTIME \[▶ 2045\]](#)
- [TOD \[▶ 2077\]](#)

The following section shows the different scenarios as code snippets.

HowTo Read/Write PlcOpen values

Reading writing by streams:

Read/Write PlcOpen types (streamed, async)

```

Cancellation token cancel = Cancellation token.None;

using (AdsClient client = new AdsClient())
{
    client.Connect(AmsNetId.Local, 851); // Connect to local plc

```

```

    ResultHandle resultHandleTime = await client.CreateVariableHandleAsync("MAIN.time", cancel); //
TIME
    ResultHandle resultHandleLTime = await client.CreateVariableHandleAsync("MAIN.lTime", cancel); /
/ LTIME
    ResultHandle resultHandleDate = await client.CreateVariableHandleAsync("MAIN.date", cancel); //
DATE

    if (resultHandleTime.Succeeded && resultHandleLTime.Succeeded && resultHandleDate.Succeeded)
    {
        try
        {
            byte[] readBuffer = new byte[LTIME.MarshalSize]; // Largest PlcOpen Type is 8 Bytes
            byte[] writeBuffer = new byte[LTIME.MarshalSize];

            // Reading raw value TIME
            await client.ReadAsync(resultHandleTime.Handle, readBuffer.AsMemory(0, TIME.MarshalSize), ca
ncel);

            // Unmarshalling
            TIME plcTime = null;
            PrimitiveTypeMarshaler.Default.Unmarshal(readBuffer.AsSpan(0, TIME.MarshalSize), out plcTime
);
            TimeSpan time = plcTime.Time;

            // Writing raw value TIME
            PrimitiveTypeMarshaler.Default.Marshal(time, writeBuffer.AsSpan());
            await client.WriteAsync(resultHandleTime.Handle, writeBuffer.AsMemory(0, TIME.MarshalSize),
cancel);

            // Reading raw value LTIME
            await client.ReadAsync(resultHandleLTime.Handle, readBuffer.AsMemory(0, LTIME.MarshalSize),
cancel);

            // Unmarshalling
            LTIME plcLTime = null;
            PrimitiveTypeMarshaler.Default.Unmarshal(readBuffer.AsSpan(0, LTIME.MarshalSize), out plcLTi
me);
            TimeSpan lTime = plcLTime.Time;

            // Writing raw value LTIME
            PrimitiveTypeMarshaler.Default.Marshal(lTime, writeBuffer.AsSpan());
            await client.WriteAsync(resultHandleLTime.Handle, writeBuffer.AsMemory(0, LTIME.MarshalSize)
, cancel);

            // Reading raw value DATE
            DATE plcDate = null;
            await client.ReadAsync(resultHandleDate.Handle, readBuffer.AsMemory(0, DATE.MarshalSize), ca
ncel);

            // Unmarshalling
            PrimitiveTypeMarshaler.Default.Unmarshal(readBuffer.AsSpan(0, DATE.MarshalSize), out plcDate
);
            DateTimeOffset dateTime = plcDate.Date;

            // Writeing raw value DATE
            PrimitiveTypeMarshaler.Default.Marshal(plcDate, writeBuffer.AsSpan());
            await client.WriteAsync(resultHandleDate.Handle, writeBuffer.AsMemory(0, DATE.MarshalSize),
cancel);
        }
        finally
        {
            ResultAds r1 = await client.DeleteVariableHandleAsync(resultHandleLTime.Handle, cancel);
            ResultAds r2 = await client.DeleteVariableHandleAsync(resultHandleTime.Handle, cancel);
            ResultAds r3 = await client.DeleteVariableHandleAsync(resultHandleDate.Handle, cancel);
        }
    }
}

```

Read/Write PlcOpen types (streamed, sync)

```

using (AdsClient client = new AdsClient())
{
    client.Connect(AmsNetId.Local, 851); // Connect to local plc

    uint handleTime = 0;
    uint handleLTime = 0;
    uint handleDate = 0;

    try
    {

```

```

handleTime = client.CreateVariableHandle("MAIN.time"); // TIME
handleLTime = client.CreateVariableHandle("MAIN.lTime"); // LTIME
handleDate = client.CreateVariableHandle("MAIN.date"); // DATE

byte[] readBuffer = new byte[LTIME.MarshalSize]; // Largest PlcOpen Type is 8 Bytes
byte[] writeBuffer = new byte[LTIME.MarshalSize];

// Reading raw value TIME
client.Read(handleTime, readBuffer.AsMemory(0, TIME.MarshalSize));

// Unmarshalling
TIME plcTime = null;
PrimitiveTypeMarshaler.Default.Unmarshal(readBuffer.AsSpan(0, TIME.MarshalSize), out plcTime);
TimeSpan time = plcTime.Time;

// Writing raw value TIME
PrimitiveTypeMarshaler.Default.Marshal(time, writeBuffer.AsSpan());
client.Write(handleTime, writeBuffer.AsMemory(0, TIME.MarshalSize));

// Reading raw value LTIME
client.Read(handleLTime, readBuffer.AsMemory(0, LTIME.MarshalSize));

// Unmarshalling
LTIME plcLTime = null;
PrimitiveTypeMarshaler.Default.Unmarshal(readBuffer.AsSpan(0, LTIME.MarshalSize), out plcLTime);
TimeSpan lTime = plcLTime.Time;

// Writing raw value LTIME
PrimitiveTypeMarshaler.Default.Marshal(lTime, writeBuffer.AsSpan());
client.Write(handleLTime, writeBuffer.AsMemory(0, LTIME.MarshalSize));

// Reading raw value DATE
DATE plcDate = null;
client.Read(handleDate, readBuffer.AsMemory(0, DATE.MarshalSize));

// Unmarshalling
PrimitiveTypeMarshaler.Default.Unmarshal(readBuffer.AsSpan(0, DATE.MarshalSize), out plcDate);
DateTimeOffset dateTime = plcDate.Date;

// Writeing raw value DATE
PrimitiveTypeMarshaler.Default.Marshal(plcDate, writeBuffer.AsSpan());
client.Write(handleDate, writeBuffer.AsMemory(0, DATE.MarshalSize));
}
finally
{
client.DeleteVariableHandle(handleLTime);
client.DeleteVariableHandle(handleTime);
client.DeleteVariableHandle(handleDate);
}
}

```

Reading writing by ANY type concept:

Read/Write PlcOpen types (ANY, async)

```

CancellationToken cancel = CancellationToken.None;

using (AdsClient client = new AdsClient())
{
    client.Connect(AmsNetId.Local, 851); // Connect to local plc

    ResultHandle resultHandleTime = await client.CreateVariableHandleAsync("MAIN.time", cancel); // T
IME
    ResultHandle resultHandleDate = await client.CreateVariableHandleAsync("MAIN.date", cancel); //
DATE
    ResultHandle resultHandleLTime = await client.CreateVariableHandleAsync("MAIN.lTime", cancel); /
/ LTIME

    if (resultHandleTime.Succeeded && resultHandleDate.Succeeded && resultHandleLTime.Succeeded)
    {
        try
        {
            ResultAnyValue resultTime = await client.ReadAnyAsync(resultHandleTime.Handle, typeof(TIME),
cancel); // TIME

            TIME time = (TIME)resultTime.Value;
            TimeSpan timeSpan = time.Time;

            await client.WriteAnyAsync(resultHandleTime.Handle, time, cancel);

```

```

        ResultAnyValue resultData = await client.ReadAnyAsync(resultHandleDate.Handle, typeof(DATE),
cancel); // DATE
        DATE date = (DATE)resultTime.Value;
        DateTimeOffset dateTime = date.Date;

        await client.WriteAnyAsync(resultHandleDate.Handle, date, cancel);

        ResultAnyValue resultLTime = await client.ReadAnyAsync(resultHandleLTime.Handle, typeof(LTIME
E), cancel); // LTIME
        LTIME lTime = (LTIME)resultTime.Value;
        TimeSpan lTimeSpan = lTime.Time;

        await client.WriteAnyAsync(resultHandleLTime.Handle, lTime, cancel);
    }
    finally
    {
        ResultAds r1 = await client.DeleteVariableHandleAsync(resultHandleTime.Handle, cancel);
        ResultAds r2 = await client.DeleteVariableHandleAsync(resultHandleDate.Handle, cancel);
        ResultAds r3 = await client.DeleteVariableHandleAsync(resultHandleLTime.Handle, cancel);
    }
}

```

Read/Write PlcOpen types (ANY, sync)

```

using (AdsClient client = new AdsClient())
{
    client.Connect(AmsNetId.Local, 851); // Connect to local plc

    uint handleTime = 0;
    uint handleDate = 0;
    uint handleLTime = 0;

    try
    {
        handleTime = client.CreateVariableHandle("MAIN.time"); // TIME
        handleDate = client.CreateVariableHandle("MAIN.date"); // DATE
        handleLTime = client.CreateVariableHandle("MAIN.lTime"); // LTIME

        TIME time = (TIME)client.ReadAny(handleTime, typeof(TIME)); // TIME
        TimeSpan timeSpan = time.Time;
        client.WriteAny(handleTime, time);

        DATE date = (DATE)client.ReadAny(handleDate, typeof(DATE)); // DATE
        DateTimeOffset dateTime = date.Date;
        client.WriteAny(handleDate, date);

        LTIME ltime = (LTIME)client.ReadAny(handleLTime, typeof(LTIME)); // LTIME
        TimeSpan lTimeSpan = ltime.Time;
        client.WriteAny(handleLTime, ltime);
    }
    finally
    {
        client.DeleteVariableHandle(handleTime);
        client.DeleteVariableHandle(handleDate);
        client.DeleteVariableHandle(handleLTime);
    }
}

```

5.4 Event driven read with ADS Notifications

Use of ADS Notifications (Async)

Trigger on changed values by ADS Notifications

```

private async Task RegisterNotificationsAsync()
{
    CancellationToken cancel = CancellationToken.None;

    using (AdsClient client = new AdsClient())
    {
        // Add the Notification event handler
        client.AdsNotification += Client_AdsNotification2;

        // Connect to target
    }
}

```

```

client.Connect(AmsNetId.Local, 851);
uint notificationHandle = 0;

// Notification to a DINT Type (UINT32)
// Check for change every 200 ms

//byte[] notificationBuffer = new byte[sizeof(UInt32)];
int size = sizeof(UInt32);

ResultHandle result = await client.AddDeviceNotificationAsync("MAIN.nCounter", size, new NotificationSettings(AdsTransMode.OnChange, 200, 0), null, cancel);

if (result.Succeeded)
{
    notificationHandle = result.Handle;
    await Task.Delay(5000); // Wait asynchronously without blocking the UI Thread.
    // Unregister the Event / Handle
    ResultAds result2 = await client.DeleteDeviceNotificationAsync(notificationHandle, cancel);
}
client.AdsNotification -= Client_AdsNotification2;
}

private void Client_AdsNotification2(object sender, AdsNotificationEventArgs e)
{
    // Or here we know about UDINT type --> can be marshalled as UINT32
    uint nCounter = BinaryPrimitives.ReadUInt32LittleEndian(e.Data.Span);

    // If Synchronization is needed (e.g. in Windows.Forms or WPF applications)
    // we could synchronize via SynchronizationContext into the UI Thread

    /*SynchronizationContext syncContext = SynchronizationContext.Current;
    _context.Post(status => someLabel.Text = nCounter.ToString(), null); // Non-blocking post */
}

```

Use of ADS Notifications (Synchronous)

Trigger on changed values by ADS Notifications

```

private void RegisterNotifications()
{
    using (AdsClient client = new AdsClient())
    {
        // Add the Notification event handler
        client.AdsNotification += Client_AdsNotification;

        // Connect to target
        client.Connect(AmsNetId.Local, 851);
        uint notificationHandle = 0;

        try
        {
            // Notification to a DINT Type (UINT32)
            // Check for change every 200 ms

            int size = sizeof(UInt32);
            //byte[] notificationBuffer = new byte[sizeof(UInt32)];

            notificationHandle = client.AddDeviceNotification("MAIN.nCounter", size, new NotificationSettings(AdsTransMode.OnChange, 200, 0), null);
            Thread.Sleep(5000); // Sleep the main thread to get some (asynchronous Notifications)
        }
        finally
        {
            // Unregister the Event / Handle
            client.DeleteDeviceNotification(notificationHandle);
            client.AdsNotification -= Client_AdsNotification;
        }
    }
}

private void Client_AdsNotification(object sender, AdsNotificationEventArgs e)
{
    // Or here we know about UDINT type --> can be marshalled as UINT32
    uint nCounter = BinaryPrimitives.ReadUInt32LittleEndian(e.Data.Span);

    // If Synchronization is needed (e.g. in Windows.Forms or WPF applications)
    // we could synchronize via SynchronizationContext into the UI Thread
}

```

```

    /*SynchronizationContext syncContext = SynchronizationContext.Current;
    _context.Post(status => someLabel.Text = nCounter.ToString(), null); // Non-blocking post */
}

```

5.5 Reactive Read/Write with Reactive Extensions

Observation of Notifications

Notifications (address specified by InstancePath) will be received cyclically as defined in [Default \[► 1111\]](#) and put into the Observer pipeline for further processing. This example takes 20 Notification samples before returning.

Observe for Notifications

```

// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Reactive Notification Handler
    var valueObserver = Observer.Create<ushort>(val =>
    {
        Console.WriteLine(string.Format("Value: {0}", val.ToString()));
    }
    );

    // Turning ADS Notifications into sequences of Value Objects (Taking 20 Values)
    // and subscribe to them.
    IDisposable subscription = client.WhenNotification<ushort>("TwinCAT_SystemInfoVarList._TaskInfo.CycleCount", NotificationSettings.Default).Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}

```

Observation of Symbolic Notifications

This example determines a symbol via Symbolloader and samples its values by Notifications with customized [NotificationSettings \[► 1106\]](#). Again 20 samples are taken before the Observation finishes.

Observe for Symbol Notifications

```

// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];

    // Reactive Notification Handler
    var valueObserver = Observer.Create<object>(val =>
    {
        Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToString()));
    }
    );

    cycleCount.NotificationSettings = new NotificationSettings(AdsTransMode.OnChange, 500, 5000); // optional: Change NotificationSettings on Symbol

    // Turning ADS Notifications into sequences of Value Objects (Taking 20 Values)
    // and subscribe to them.
    IDisposable subscription = cycleCount.WhenValueChanged().Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}

```

Observation of dynamic Symbol Notifications

Here, the symbol is determined via Symbolloader again, but now the Notifications will be processed as 'dynamic' values.

Observer for dynamic Symbol Notifications

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = (IDynamicSymbolLoader)SymbolLoaderFactory.Create(client, SymbolLoaderSettings.DefaultDynamic);
    dynamic symbols = symbolLoader.SymbolsDynamic;
    dynamic cycleCount = symbols.TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount;

    // Reactive Notification Handler
    var valueObserver = Observer.Create<object>(val =>
    {
        // Value objects can be dynamically (on the fly) created objects here (e.g. structs)
        Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToString()));
    });

    cycleCount.NotificationSettings = new NotificationSettings(AdsTransMode.OnChange, 500, 5000); // optional: Change NotificationSettings on Symbol

    // Turning ADS Notifications into sequences of Value Objects (Taking 20 Values)
    // and subscribe to them.
    // We have to give the 'hint' about IValueSymbol here, that the CLR finds the Extension Method 'WhenValueChanged' during runtime.
    IDisposable subscription = ((IValueSymbol)cycleCount).WhenValueChanged().Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Polling observer

A polling observer doesn't use ADS notifications, but instead the value read is triggered by a time interval (polling) or a customized trigger function (on request).

Polling observer

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo.CycleCount"];

    // Reactive Notification Handler
    var valueObserver = Observer.Create<object>(val =>
    {
        Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToString()));
    });

    // Take 20 Values in an Interval of 500ms
    IDisposable subscription = cycleCount.PollValues(TimeSpan.FromMilliseconds(500)).Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Writing values with observable subject

In this example, a symbolic value is written in a static time interval (1 second). The writing stops after 10 values. The [WriteValues \[▶ 1347\]](#) extension method can be used to seamlessly bind value writing into a reactive application.

Writing values with observable subject

```
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information (Symbol 'i : INT' in PLC Global Variables list.
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol gvlIntSymbol = (IValueSymbol)symbolLoader.Symbols["GVL.i"];

    // Produces object (short) Values 0,1,2,3 ... in seconds period
    IObservable<object> timerObservable = Observable.Interval(TimeSpan.FromSeconds(1.0)).Select(i =>
(object) (short)i);

    // Take 10 Values (0..9) and write them to GVL.i
    IDisposable dispose = gvlIntSymbol.WriteValues(timerObservable.Take(10));

    Console.ReadKey(); // Wait for Key press
    dispose.Dispose(); // Dispose the Subscription
}
```

5.6 Upgrading existing ADS Application code (Version 4.X -- > 6.X)

Code migration path for application code Beckhoff.TwinCAT.Ads 4.X --> 6.X

The Beckhoff.TwinCAT.Ads package Version 4.X implements the Beckhoff ADS API for the Microsoft .NET FullFramework (>= 4.0) / CompactFramework (>= 2.0). Newer .NET framework implementations are basing on '.NET Core' (e.g. .NET 6.0).

To support these newer flavours like .NET Core, .NET Standard together with its new features, a new Version 6.X of the Beckhoff.TwinCAT.Ads API is available.

Because new platform features need some breaking changes within the API for seamless support, the decision was taken to remove some legacy features and burdens in the reimplementaion. Nevertheless most of the interfaces remained stable and only a few approaches changed.

The breaking changes and some guidelines will be discussed here.

Use 'async' wherever possible.

Internally, the Beckhoff.TwinCAT.Ads package are implemented asynchronously (Methodnames ending with 'Async'). The synchronous counterparts of the async methods are implemented as thin wrapper around the async method with synchronization.

As consequence the synchronous method is less flexible (only usable in synchronous code paths) and less performant (blocking calls, more thread switches). Furthermore, with the 'async' keyword support of .NET they are also not easier to use. So take more than a thought about migrate to async code.

Synchronous Read value

```
uint variableHandle = ...;
int value;
AdsErrorCode errorCode = adsClient.TryReadValue<int>(variableHandle, out int value);
```

Asynchronous

```
uint variableHandle = ...;
ResultValue<int> result = await adsClient.ReadValueAsync(variableHandle, CancellationToken.None);
AdsErrorCode errorCode = result.ErrorCode;
int value = result.Value;
```


However in synchronous application code paths the await keyword is not allowed. Using the synchronous methods is the only option besides migrating the whole application code up to the main root. Further information can be found here: [Asynchronous programming with async and await](#)

Name change TcAdsClient --> AdsClient

This should simply document that there is another implementation. Just rename the class instance.

Version 4.X

```
TcAdsClient client = new TcAdsClient()
```

Version 6.X

```
AdsClient client = new AdsClient()
```

Marshalling of byte[] and missing AdsStream class.

Newer .NET implementations have more efficient possibilities to exchange memory data between software layers and classes (see [Memory- and span-related types](#)).

All API interface methods that use the combination of parameters (bytes, offset, length) or AdsStream

Version 4.X

```
public int TcAdsClient.Read(uint indexGroup, uint indexOffset, AdsStream dataStream)
```

Version 4.X

```
public int TcAdsClient.Write(uint indexGroup, uint indexOffset, AdsStream dataStream)
```

or

Version 4.X

```
public int TcAdsClient.Read(uint indexGroup, uint indexOffset, byte[] bytes, int offset, length)
```

Version 4.X

```
public int TcAdsClient.Write(uint indexGroup, uint indexOffset, byte[] bytes, int offset, length)
```

are changed like this:

Version 6.X

```
public int AdsClient.Read(uint indexGroup, uint indexOffset, Memory<byte> buffer)
```

Version 6.X

```
public int AdsClient.Write(uint indexGroup, uint indexOffset, ReadOnlyMemory<byte> buffer)
```

With this change a lot of interfaces will look much cleaner and act more efficient because of less array copy operations.

Handles are defined as uint type.

Handles are now streamlined consistently as uint type instead of int.

Version 4.X

```
public int TcAdsClient.CreateVariableHandle(string symbolPath)
```

Version 6.X

```
public uint AdsClient.CreateVariableHandle(string symbolPath)
```

Notification parameters bundled in one settings class.**Version 4.X**

```
public int AddDeviceNotification(string symbolPath, AdsStream dataStream, int offset, int length, AdsTransMode transMode, int cycleTime, int maxDelay, Object userData)
```

Version 4.X

```
public int TcAdsClient.AddDeviceNotification(string symbolPath, AdsStream dataStream, int offset, int length, AdsTransMode transMode, int cycleTime, int maxDelay, Object userData)
```

The new parameter-set simply needs the (byte) size of the transferred data instead of AdsStream, offset and length. Furthermore transMode, cycleTime and maxDelay are bundled in the NotificationSettings object.

Version 6.X

```
public uint AdsClient.AddDeviceNotification(string symbolPath, int dataSize, NotificationSettings settings, object userData)
```

6 TwinCAT.Ads Namespaces

TwinCAT ADS .NET API for .NET CORE

Namespaces

Namespace	Description
TwinCAT [▶ 59]	Common namespace for types that are not specific to ADS.
TwinCAT.Ads [▶ 179]	ADS root namespace.
TwinCAT.Ads.Extensions [▶ 3292]	
TwinCAT.Ads.Reactive [▶ 1249]	Reactive Extensions for the ADS Client. All types within are contained in the ADS companion package "Beckhoff.TwinCAT.Ads.Reactive" which must be referenced separately. https://www.nuget.org/packages/Beckhoff.TwinCAT.Ads.Reactive/
TwinCAT.Ads.Server [▶ 1374]	This framework can be used to build custom ADS server application. Important note: Please contact Beckhoff to receive a reserved ADS port number. Namespace for the TwinCAT ADS Server Component.
TwinCAT.Ads.Server.TypeSystem [▶ 3230]	
TwinCAT.Ads.SumCommand [▶ 1525]	ADS offers powerful and fast communication to exchange any kind of information. It's possible to read single variables or complete arrays and structures with each one single ADS-API call. ADS Sum-Commands offer to read/write with one single ADS call multiple variables which are not structured within a linear memory, effectively reducing roundtrips.
TwinCAT.Ads.TcpRouter [▶ 1619]	
TwinCAT.Ads.TypeSystem [▶ 1698]	Root namespace for the ADS type system.
TwinCAT.Ams [▶ 2003]	
TwinCAT.PlcOpen [▶ 2016]	
TwinCAT.Router [▶ 3255]	
TwinCAT.SystemService [▶ 3272]	
TwinCAT.Tls [▶ 3288]	
TwinCAT.TypeSystem [▶ 2083]	Namespace for the common (non ADS dependent) type system.
TwinCAT.TypeSystem.Generic [▶ 3086]	Namespace for the dynamic part of the common type system.
TwinCAT.ValueAccess [▶ 3179]	Namespace for the common (non ADS dependent) value access.












Also see about this

- [TwinCAT.Ads.ValueAccess Namespace \[▶ 2002\]](#)






6.1 TwinCAT Namespace





Common namespace for types that are not specific to ADS.

Classes







	Class	Description
	AdsCommunicationStatistics [▶ 160]	ADS Communication statistics
	AdsException [▶ 61]	Base class for all exceptions thrown by the TwinCAT.Ads component
	ClientNotConnectedException [▶ 67]	Class ClientNotConnectedException
	ConnectionStateChangedEventArgs [▶ 74]	Event arguments for the Connection status changed event.
	HandleBagNotInitializedException [▶ 171]	Class ClientNotConnectedException
	Session [▶ 107]	Abstract Session base class.
	SessionConnectionStateChangedEventArgs [▶ 125]	EventArguments for the ConnectionStateChanged events.
	SessionException [▶ 130]	Session Exception
	SessionNotConnectedException [▶ 137]	Class SessionNotConnectedException.
	SessionProvider.TSession, TAddress, TSettings . [▶ 142]	Abstract base class for a Custom Session provider
	SymbolLoaderSettings [▶ 148]	Settings object for the SymbolLoader initialization.

Interfaces

	Interface	Description
	IAdsClientSettings [▶ 175]	Settings for the Ads client
	IConnection [▶ 79]	Interface IConnection
	IConnectionStateProvider [▶ 89]	Interface IConnectionStateProvider
	ISession [▶ 92]	Interface ISession
	ISessionProvider [▶ 99]	Interface ISessionProvider

	Interface	Description
	ISessionProvider.TSession, TAddress, TSettings. [▸ 102]	Generic ISessionProvider interface
	ISymbolLoaderSettings [▸ 104]	Interface ISymbolLoaderSettings
	ISymbolServerProvider [▸ 106]	Interface ISymbolServerProvider
	IValueFactorySettings [▸ 176]	Interface IValueFactorySettings

Enumerations

	Enumeration	Description
	ConnectionState [▸ 72]	Connection state enumeration
	ConnectionStateChangedReason [▸ 79]	Reason for the Connection status changed event.
	RouteAccessType [▸ 169]	Specifies the method to determine the Remote routes list.
	RoutePersistenceType [▸ 170]	Route persistence type (Static or Temporary)
	SessionProviderCapabilities [▸ 148]	Enum SessionProviderCapabilities
	SymbolsLoadMode [▸ 159]	Enum SymbolsLoadMode

Also see about this

 [ValueUpdateMode Enumeration \[▸ 159\]](#)

6.1.1 AdsException Class

Base class for all exceptions thrown by the TwinCAT.Ads component

Inheritance Hierarchy

[System.Object](#)
[System.Exception](#)
[TwinCAT.AdsException](#)
[More... \[▸ 63\]](#)

Namespace: [TwinCAT \[▸ 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229





Syntax

C#









```
[SerializableAttribute]
public class AdsException : Exception
```

The AdsException type exposes the following members.




Constructors






	Name	Description
	AdsException. [▶ 64]	Initializes a new Instance of the AdsException class.
	AdsException(String) [▶ 64]	Initializes a new Instance of the AdsException class.
	AdsException(SerializationInfo, StreamingContext) [▶ 65]	Initializes a new instance of the AdsException class.
	AdsException(String, Exception) [▶ 65]	Initializes a new Instance of the AdsException class.

Properties


	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)

	Name	Description
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

Reference




[TwinCAT Namespace](#) [► 59]


Inheritance Hierarchy

- [System.Object](#)
- [System.Exception](#)
- [TwinCAT.AdsException](#)
- [TwinCAT.Ads.AdsErrorException](#) [► 682]
- [TwinCAT.Ads.AdsInvalidNotificationException](#) [► 690]
- [TwinCAT.ClientNotConnectedException](#) [► 67]
- [TwinCAT.HandleBagNotInitializedException](#) [► 171]
- [TwinCAT.SessionException](#) [► 130]
- [TwinCAT.TypeSystem.DataTypeException](#) [► 2121]
- [TwinCAT.TypeSystem.MarshalException](#) [► 2795]
- [TwinCAT.TypeSystem.SymbolException](#) [► 2933]

6.1.1.1 AdsException Constructor

Overload List

	Name	Description
	AdsException . [► 64]	Initializes a new Instance of the AdsException [► 61] class.
	AdsException(String) [► 64]	Initializes a new Instance of the AdsException [► 61] class.
	AdsException(SerializationInfo, StreamingContext) [► 65]	Initializes a new instance of the AdsException [► 61] class.

	Name	Description
	<code>AdsException(String, Exception)</code> [▶ 65]	Initializes a new Instance of the <code>AdsException</code> class.

Reference

[AdsException Class](#) [[▶ 61](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.1.1 AdsException Constructor

Initializes a new Instance of the [AdsException](#) [[▶ 61](#)] class.

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsException()
```

Reference

[AdsException Class](#) [[▶ 61](#)]

[AdsException Overload](#) [[▶ 63](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.1.1.2 AdsException Constructor (String)

Initializes a new Instance of the [AdsException](#) [[▶ 61](#)] class.

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsException(
    string message
)
```

Parameters

message	Type: <code>System.String</code> A message that describes the error.
---------	---

Reference

[AdsException Class](#) [[▶ 61](#)]

[AdsException Overload](#) [[▶ 63](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.1.1.3 AdsException Constructor (SerializationInfo, StreamingContext)

Initializes a new instance of the [AdsException](#) [► 61] class.

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected AdsException(
    SerializationInfo serializationInfo,
    StreamingContext streamingContext
)
```

Parameters

serializationInfo	Type: System.Runtime.Serialization.SerializationInfo The serialization information.
streamingContext	Type: System.Runtime.Serialization.StreamingContext The streaming context.

Reference

[AdsException Class](#) [► 61]

[AdsException Overload](#) [► 63]

[TwinCAT Namespace](#) [► 59]

6.1.1.1.4 AdsException Constructor (String, Exception)

Initializes a new Instance of the AdsException class.

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsException(
    string message,
    Exception? innerException
)
```

Parameters

message	Type: System.String The error message that explains the reason for the exception.
innerException	Type: System.Exception The exception that is the cause of the current exception. If the innerException parameter is not a null reference, the current exception is raised in a catch block that handles the inner exception.

Reference

[AdsException Class](#) [► 61]









[AdsException Overload](#) [► 63]

[TwinCAT Namespace](#) [► 59]

6.1.1.2 AdsException Properties

The [AdsException](#) [► 61] type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Reference









[AdsException Class](#) [► 61]

[TwinCAT Namespace](#) [► 59]

6.1.1.3 AdsException Methods

The [AdsException](#) [► 61] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Reference


[AdsException Class \[▶ 61\]](#)

[TwinCAT Namespace \[▶ 59\]](#)

6.1.1.4 AdsException Events

The [AdsException \[▶ 61\]](#) type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)

Reference

[AdsException Class \[▶ 61\]](#)

[TwinCAT Namespace \[▶ 59\]](#)

6.1.2 ClientNotConnectedException Class

Class [ClientNotConnectedException](#)

Inheritance Hierarchy

- [System.Object](#)
- [System.Exception](#)
- [TwinCAT.AdsException \[▶ 61\]](#)
- [TwinCAT.ClientNotConnectedException](#)

Namespace: [TwinCAT \[▶ 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229



Syntax

C#










```
[SerializableAttribute]
public class ClientNotConnectedException : AdsException
```

The [ClientNotConnectedException](#) type exposes the following members.









Constructors

	Name	Description
	ClientNotConnectedException(IConnection) [▶ 70]	Initializes a new instance of the ClientNotConnectedException class.
	ClientNotConnectedException(SerializationInfo, StreamingContext) [▶ 69]	Initializes a new instance of the ClientNotConnectedException class.


Properties

	Name	Description
	Connection [▶ 71]	Gets the connection.
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)



Reference

[TwinCAT Namespace](#) [[▶ 59](#)]

Also see about this

 [ClientNotConnectedException Constructor \[▶ 69\]](#)

6.1.2.1 ClientNotConnectedException Constructor**Overload List**

	Name	Description
	ClientNotConnectedException(IConnection) [▶ 70]	Initializes a new instance of the ClientNotConnectedException [▶ 67] class.
	ClientNotConnectedException(SerializationInfo, StreamingContext) [▶ 69]	Initializes a new instance of the ClientNotConnectedException [▶ 67] class.

Reference

[ClientNotConnectedException Class \[▶ 67\]](#)

[TwinCAT Namespace \[▶ 59\]](#)

Also see about this

 [ClientNotConnectedException Constructor \[▶ 69\]](#)

6.1.2.1.1 ClientNotConnectedException Constructor

Initializes a new instance of the [ClientNotConnectedException \[▶ 67\]](#) class.

Namespace: [TwinCAT \[▶ 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.116+a71ced3

Syntax**C#**

```
public ClientNotConnectedException()
```

Reference

[ClientNotConnectedException Class \[▶ 67\]](#)

[ClientNotConnectedException Overload \[▶ 69\]](#)

[TwinCAT Namespace \[▶ 59\]](#)

6.1.2.1.2 ClientNotConnectedException Constructor (SerializationInfo, StreamingContext)

Initializes a new instance of the [ClientNotConnectedException \[▶ 67\]](#) class.

Namespace: [TwinCAT \[▶ 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected ClientNotConnectedException(
    SerializationInfo serializationInfo,
    StreamingContext streamingContext
)
```

Parameters

serializationInfo	Type: System.Runtime.Serialization.SerializationInfo The serialization information.
streamingContext	Type: System.Runtime.Serialization.StreamingContext The streaming context.

Reference

[ClientNotConnectedException Class](#) [► 67]

[ClientNotConnectedException Overload](#) [► 69]

[TwinCAT Namespace](#) [► 59]

6.1.2.1.3 ClientNotConnectedException Constructor (IConnection)

Initializes a new instance of the [ClientNotConnectedException](#) [► 67] class.

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ClientNotConnectedException(
    IConnection? connection
)
```

Parameters

connection	Type: TwinCAT.IConnection [► 79]
------------	--

Reference

[ClientNotConnectedException Class](#) [► 67]


[ClientNotConnectedException Overload](#) [► 69]









[TwinCAT Namespace](#) [► 59]

6.1.2.2 ClientNotConnectedException Properties

The [ClientNotConnectedException](#) [► 67] type exposes the following members.

Properties

	Name	Description
	Connection [► 71]	Gets the connection.

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Reference

[ClientNotConnectedException Class](#) [► 67]

[TwinCAT Namespace](#) [► 59]

6.1.2.2.1 ClientNotConnectedException.Connection Property

Gets the connection.

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IConnection? Connection { get; }
```

Property Value

Type: [IConnection](#) [► 79]

The connection.

Reference


[ClientNotConnectedException Class](#) [► 67]








[TwinCAT Namespace](#) [► 59]

6.1.2.3 ClientNotConnectedException Methods

The [ClientNotConnectedException](#) [► 67] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)

	Name	Description
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Reference


[ClientNotConnectedException Class](#) [► 67]

[TwinCAT Namespace](#) [► 59]

6.1.2.4 ClientNotConnectedException Events

The [ClientNotConnectedException](#) [► 67] type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

Reference

[ClientNotConnectedException Class](#) [► 67]

[TwinCAT Namespace](#) [► 59]

6.1.3 ConnectionState Enumeration

Table 1: Extracted nested table 0

AdsErrorCode [► 664]	Description
WSA_ConnRefused [► 664]	The target computer actively refused the socket connection.
PortDisabled [► 664]	The target port is disabled.
PortNotConnected [► 664]	The target port is not connected.
ClientSyncTimeOut [► 664]	The target system didn't answer within the configured timeout timespan.
TargetMachineNotFound [► 664]	The target machine was not found.
TargetPortNotFound [► 664]	The ADS target port couldn't be found on the target address.

AdsErrorCode [▶ 664]	Description
ClientPortNotOpen [▶ 664]	The ADS target port on the target address is not open.

Connection state enumeration

Namespace: [TwinCAT \[▶ 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum ConnectionState
```

Members

	Member name	Value	Description
	None	0	Unknown / Uninitialized
	Disconnected	1	Disconnected
	Connected	2	Connected
	Lost	3	<p>Connection lost</p> <p>The connection is (temporarily) lost and can be reestablished / resurrected after a (resurrection) timeout.</p> <p>During ADS communication, specific AdsErrorCode [▶ 664] responses lead to so called 'Tripping Errors' that triggers the change from Connected to Lost. This is a behavior of the internal ConnectionStateInterceptor that observes the condition of the actual connection and can be configured via the session settings. So called tripping errors are the following:see "Extracted nested table 0"</p> <p>All theses described ADS error lead to lost connection state. Depending on the configured communication interceptor / or Session settings the ConnectionState returns to Connected under the following conditions:</p> <ul style="list-style-type: none"> • An ADS communication is triggered (Read/Write/ReadWrite) is triggered by the application. • The resurrection time is elapsed (21 Seconds, if the Default Session settings / Interceptors are used) <p>A setting of <code>TwinCAT.Ads.SessionSettings.FastWriteThrough</code> is switching off this behaviour.</p>

Reference

[TwinCAT Namespace](#) [► 59]

6.1.4 ConnectionStateChangedEventArgs Class

Event arguments for the Connection status changed event.

Inheritance Hierarchy

[System.Object](#)

[System.EventArgs](#)

[TwinCAT.ConnectionStateChangedEventArgs](#)

[TwinCAT.SessionConnectionStateChangedEventArgs](#) [► 125]

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax

C#





```
public class ConnectionStateChangedEventArgs : EventArgs
```

The ConnectionStateChangedEventArgs type exposes the following members.







Constructors

	Name	Description
	ConnectionStateChangedEventArgs(ConnectionStateChangeReason, ConnectionState, ConnectionState) [► 75]	Constructs the ConnectionStateChangedEventArgs arguments.
	ConnectionStateChangedEventArgs(ConnectionStateChangeReason, ConnectionState, ConnectionState, Exception) [► 76]	Constructs the ConnectionStateChangedEventArgs arguments.

Properties

	Name	Description
	Exception [► 77]	Exception, (only for Error [► 79])
	NewState [► 77]	New connection state
	OldState [► 78]	Old connection state
	Reason [► 78]	Reason for the event

Methods



	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[TwinCAT Namespace](#) [[▶](#) 59]

6.1.4.1 ConnectionStateChangedEventArgs Constructor

Overload List

	Name	Description
	ConnectionStateCh angedEventArgs(Con nectionStateChange dReason, ConnectionState, ConnectionState) [▶ 75]	Constructs the ConnectionStateChangedEventArgs [▶ 74] arguments.
	ConnectionStateCh angedEventArgs(Con nectionStateChange dReason, ConnectionState, ConnectionState, Exception) [▶ 76]	Constructs the ConnectionStateChangedEventArgs [▶ 74] arguments.

Reference

[ConnectionStateChangedEventArgs Class](#) [[▶](#) 74]

[TwinCAT Namespace](#) [[▶](#) 59]

6.1.4.1.1 ConnectionStateChangedEventArgs Constructor (ConnectionStateChangedReason, ConnectionState, ConnectionState)

Constructs the [ConnectionStateChangedEventArgs](#) [[▶](#) 74] arguments.

Namespace: [TwinCAT](#) [[▶](#) 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ConnectionStateChangedEventArgs(
    ConnectionStateChangedReason reason,
    ConnectionState newState,
    ConnectionState oldState
)
```

Parameters

reason	Type: TwinCAT.ConnectionStateChangedReason [► 79] The reason.
newState	Type: TwinCAT.ConnectionState [► 72] The new state.
oldState	Type: TwinCAT.ConnectionState [► 72] The old state.

Reference

[ConnectionStateChangedEventArgs Class](#) [► 74]

[ConnectionStateChangedEventArgs Overload](#) [► 75]

[TwinCAT Namespace](#) [► 59]

6.1.4.1.2 **ConnectionStateChangedEventArgs Constructor (ConnectionStateChangedReason, ConnectionState, ConnectionState, Exception)**

Constructs the [ConnectionStateChangedEventArgs](#) [► 74] arguments.

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ConnectionStateChangedEventArgs(
    ConnectionStateChangedReason reason,
    ConnectionState newState,
    ConnectionState oldState,
    Exception? e
)
```

Parameters

reason	Type: TwinCAT.ConnectionStateChangedReason [► 79] The reason.
newState	Type: TwinCAT.ConnectionState [► 72] The new state.
oldState	Type: TwinCAT.ConnectionState [► 72] The old state.
e	Type: System.Exception The e.

Reference

[ConnectionStateChangedEventArgs Class](#) [► 74]





[ConnectionStringChangedEventArgs Overload \[► 75\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.4.2 ConnectionStateChangedEventArgs Properties

The [ConnectionStringChangedEventArgs \[► 74\]](#) type exposes the following members.

Properties

	Name	Description
	Exception [► 77]	Exception, (only for Error [► 79])
	NewState [► 77]	New connection state
	OldState [► 78]	Old connection state
	Reason [► 78]	Reason for the event

Reference

[ConnectionStringChangedEventArgs Class \[► 74\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.4.2.1 ConnectionStateChangedEventArgs.Exception Property

Exception, (only for [Error \[► 79\]](#))

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Exception? Exception { get; }
```

Property Value

Type: [Exception](#)

Reference

[ConnectionStringChangedEventArgs Class \[► 74\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.4.2.2 ConnectionStateChangedEventArgs.NewState Property

New connection state

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ConnectionState NewState { get; }
```

Property Value

Type: [ConnectionState](#) [► 72]

Reference

[ConnectionStateChangedEventArgs Class](#) [► 74]

[TwinCAT Namespace](#) [► 59]

6.1.4.2.3 ConnectionStateChangedEventArgs.OldState Property

Old connection state

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ConnectionState OldState { get; }
```

Property Value

Type: [ConnectionState](#) [► 72]

Reference

[ConnectionStateChangedEventArgs Class](#) [► 74]

[TwinCAT Namespace](#) [► 59]

6.1.4.2.4 ConnectionStateChangedEventArgs.Reason Property

Reason for the event

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ConnectionStateChangedReason Reason { get; }
```

Property Value

Type: [ConnectionStateChangedReason](#) [► 79]

Reference







[ConnectionStateChangedEventArgs Class](#) [► 74]

[TwinCAT Namespace](#) [► 59]

6.1.4.3 ConnectionStateChangedEventArgs Methods

The [ConnectionStateChangedEventArgs](#) [► 74] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ConnectionStringChangedEventArgs Class \[► 74\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.5 ConnectionStateChangedReason Enumeration

Reason for the Connection status changed event.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum ConnectionStateChangedReason
```

Members

	Member name	Value	Description
	None	0	None / Uninitialized
	Established	1	The connection to the target has been established
	Closed	2	The Connection was closed
	Lost	3	The connection to the target has been lost
	Error	4	Communication error to the target (the connection is not shutting down)
	Resurrected	5	Communication was resurrected (available again)

Reference

[TwinCAT Namespace \[► 59\]](#)

6.1.6 IConnection Interface

Interface IConnection

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229







Syntax

C#




```
public interface IConnection : IConnectionStateProvider
```

The IConnection type exposes the following members.


Properties

	Name	Description
	ConnectionState [▶ 90]	Gets the current Connection state of the IConnectionStateProvider [▶ 89] (Inherited from IConnectionStateProvider [▶ 89].)
	DefaultValueEncoding [▶ 81]	Gets the default value encoding.
	Id [▶ 81]	Gets the Connection Identifier .
	IsConnected [▶ 82]	Gets a value indicating whether the local ADS port was opened successfully. It does not indicate if the target port is available. Use the method ReadState to determine if the target port is available.
	Session [▶ 82]	Gets the session that initiated this IConnection
	Timeout [▶ 83]	Gets the timeout (in milliseconds)

Methods

	Name	Description
	Close [▶ 83]	Closes this IConnection
	Connect [▶ 84]	(Re)Connects the IConnection when disconnected.
	Disconnect [▶ 84]	Disconnects this IConnection.

Events

	Name	Description
	ConnectionStateChanged [▶ 91]	Occurs when connection status of the IConnectionStateProvider [▶ 89] has been changed. (Inherited from IConnectionStateProvider [▶ 89].)


Reference







[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.6.1 IConnection Properties

The [IConnection](#) [[▶ 79](#)] type exposes the following members.

Properties

	Name	Description
	ConnectionState [▶ 90]	Gets the current Connection state of the IConnectionStateProvider [▶ 89] (Inherited from IConnectionStateProvider [▶ 89].)

	Name	Description
		
	DefaultValueEncoding [▶ 81]	Gets the default value encoding.
	Id [▶ 81]	Gets the Connection Identifier .
	IsConnected [▶ 82]	Gets a value indicating whether the local ADS port was opened successfully. It does not indicate if the target port is available. Use the method ReadState to determine if the target port is available.
	Session [▶ 82]	Gets the session that initiated this IConnection [▶ 79]
	Timeout [▶ 83]	Gets the timeout (in milliseconds)

Reference

[IConnection Interface](#) [[▶ 79](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.6.1.1 IConnection.DefaultValueEncoding Property

Gets the default value encoding.

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Encoding DefaultValueEncoding { get; }
```

Property Value

Type: [Encoding](#)

The default value encoding.

Reference

[IConnection Interface](#) [[▶ 79](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.6.1.2 IConnection.Id Property

Gets the Connection Identifier .

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Id { get; }
```

Property Value

Type: [Int32](#)

The identifier.

Reference

[IConnection Interface](#) [► 79]

[TwinCAT Namespace](#) [► 59]

6.1.6.1.3 IConnection.IsConnected Property

Gets a value indicating whether the local ADS port was opened successfully. It does not indicate if the target port is available. Use the method `ReadState` to determine if the target port is available.

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsConnected { get; }
```

Property Value

Type: [Boolean](#)

Reference

[IConnection Interface](#) [► 79]

[TwinCAT Namespace](#) [► 59]

6.1.6.1.4 IConnection.Session Property

Gets the session that initiated this [IConnection](#) [► 79]

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISession? Session { get; }
```

Property Value

Type: [ISession](#) [► 92]

The session or NULL

Remarks

The Session can be null on standalone connections.

Reference

[IConnection Interface](#) [► 79]

[TwinCAT Namespace](#) [► 59]

6.1.6.1.5 IConnection.Timeout Property

Gets the timeout (in milliseconds)

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Timeout { get; set; }
```

Property Value

Type: [Int32](#)

The timeout.

Reference




[IConnection Interface \[► 79\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.6.2 IConnection Methods

The [IConnection \[► 79\]](#) type exposes the following members.

Methods

	Name	Description
	Close [► 83]	Closes this IConnection [► 79]
	Connect [► 84]	(Re)Connects the IConnection [► 79] when disconnected.
	Disconnect [► 84]	Disconnects this IConnection [► 79] .

Reference

[IConnection Interface \[► 79\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.6.2.1 IConnection.Close Method

Closes this [IConnection \[► 79\]](#)

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void Close()
```

Reference

[IConnection Interface \[► 79\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.6.2 IConnection.Connect Method

(Re)Connects the [IConnection](#) [[▶ 79](#)] when disconnected.

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool Connect()
```

Return Value

Type: [Boolean](#)

true if connected, false otherwise.

Reference

[IConnection Interface](#) [[▶ 79](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.6.2.3 IConnection.Disconnect Method

Disconnects this [IConnection](#) [[▶ 79](#)].

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool Disconnect()
```

Return Value

Type: [Boolean](#)

true if disconnected, false otherwise.

Reference



[IConnection Interface](#) [[▶ 79](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.6.3 IConnection Events

The [IConnection](#) [[▶ 79](#)] type exposes the following members.

Events

	Name	Description
 	ConnectionStateChanged [▶ 91]	Occurs when connection status of the IConnectionStateProvider [▶ 89] has been changed. (Inherited from IConnectionStateProvider [▶ 89].)

Reference

[IConnection Interface](#) [▶ 79]

[TwinCAT Namespace](#) [▶ 59]

6.1.7 IConnectionStateObserver Interface

Interface for a Connection state watcher (observer)

Namespace: [TwinCAT](#) [▶ 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14








Syntax

C#


```
public interface IConnectionStateObserver : IConnectionStateProvider
```

The IConnectionStateObserver type exposes the following members.

Properties

	Name	Description
	ConnectionState [▶ 90]	Gets the current Connection state of the IConnectionStateProvider [▶ 89] (Inherited from IConnectionStateProvider [▶ 89].)
	ErrorsSinceLastSucceeded [▶ 86]	Gets the number of errors since the last successful access
	LastAccess [▶ 86]	Gets the DateTimeOffset of the last tried access
	LastSucceededAccess [▶ 87]	Gets the DateTimeOffset of the last successful data communication
	Quality [▶ 87]	Gets the quality of the current cached value (the age of the data)
	TotalCycles [▶ 88]	Gets the number of successful reads / writes
	TotalErrors [▶ 88]	Gets the error count of accesses

Events

	Name	Description
	ConnectionStateChanged [▶ 91]	Occurs when connection status of the IConnectionStateProvider [▶ 89] has been changed. (Inherited from IConnectionStateProvider [▶ 89].)








Reference

[TwinCAT Namespace](#) [▶ 59]

6.1.7.1 IConnectionStateObserver Properties

The [IConnectionStateObserver](#) [▶ 85] type exposes the following members.

Properties

	Name	Description
	ConnectionState [▶ 90]	Gets the current Connection state of the IConnectionStateProvider [▶ 89] (Inherited from IConnectionStateProvider [▶ 89].)
	ErrorsSinceLastSucceeded [▶ 86]	Gets the number of errors since the last successful access
	LastAccess [▶ 86]	Gets the DateTimeOffset of the last tried access
	LastSucceededAccess [▶ 87]	Gets the DateTimeOffset of the last successful data communication
	Quality [▶ 87]	Gets the quality of the current cached value (the age of the data)
	TotalCycles [▶ 88]	Gets the number of successful reads / writes
	TotalErrors [▶ 88]	Gets the error count of accesses

Reference

[IConnectionStateObserver Interface](#) [[▶ 85](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.7.1.1 IConnectionStateObserver.ErrorsSinceLastSucceeded Property

Gets the number of errors since the last successful access

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
int ErrorsSinceLastSucceeded { get; }
```

Property Value

Type: [Int32](#)

Reference

[IConnectionStateObserver Interface](#) [[▶ 85](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.7.1.2 IConnectionStateObserver.LastAccess Property

Gets the [DateTimeOffset](#) of the last tried access

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
DateTimeOffset LastAccess { get; }
```

Property Value

Type: [DateTimeOffset](#)

Reference

[IConnectionStateObserver Interface](#) [► 85]

[TwinCAT Namespace](#) [► 59]

6.1.7.1.3 IConnectionStateObserver.LastSucceededAccess Property

Gets the [DateTimeOffset](#) of the last successful data communication

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
DateTimeOffset LastSucceededAccess { get; }
```

Property Value

Type: [DateTimeOffset](#)

Reference

[IConnectionStateObserver Interface](#) [► 85]

[TwinCAT Namespace](#) [► 59]

6.1.7.1.4 IConnectionStateObserver.Quality Property

Gets the quality of the current cached value (the age of the data)

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
TimeSpan Quality { get; }
```

Property Value

Type: [TimeSpan](#)

Reference

[IConnectionStateObserver Interface](#) [► 85]

[TwinCAT Namespace](#) [► 59]

6.1.7.1.5 IConnectionStateObserver.TotalCycles Property

Gets the number of successful reads / writes

Namespace: [TwinCAT](#) [[► 59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
int TotalCycles { get; }
```

Property Value

Type: [Int32](#)

Reference

[IConnectionStateObserver Interface](#) [[► 85](#)]

[TwinCAT Namespace](#) [[► 59](#)]

6.1.7.1.6 IConnectionStateObserver.TotalErrors Property

Gets the error count of accesses

Namespace: [TwinCAT](#) [[► 59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
int TotalErrors { get; }
```

Property Value

Type: [Int32](#)

Reference



[IConnectionStateObserver Interface](#) [[► 85](#)]

[TwinCAT Namespace](#) [[► 59](#)]

6.1.7.2 IConnectionStateObserver Events

The [IConnectionStateObserver](#) [[► 85](#)] type exposes the following members.

Events

	Name	Description
 	ConnectionStateChanged [► 91]	Occurs when connection status of the IConnectionStateProvider [► 89] has been changed. (Inherited from IConnectionStateProvider [► 89].)

Reference

[IConnectionStateObserver Interface \[▶ 85\]](#)

[TwinCAT Namespace \[▶ 59\]](#)

6.1.8 IConnectionStateProvider Interface

Interface IConnectionStateProvider

Namespace: [TwinCAT \[▶ 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#


```
public interface IConnectionStateProvider
```

The IConnectionStateProvider type exposes the following members.

Properties

	Name	Description
	ConnectionState [▶ 90]	Gets the current Connection state of the IConnectionStateProvider

Events

	Name	Description
	ConnectionStateChanged [▶ 91]	Occurs when connection status of the IConnectionStateProvider has been changed.


Reference

[TwinCAT Namespace \[▶ 59\]](#)

6.1.8.1 IConnectionStateProvider Properties

The [IConnectionStateProvider \[▶ 89\]](#) type exposes the following members.

Properties

	Name	Description
	ConnectionState [▶ 90]	Gets the current Connection state of the IConnectionStateProvider [▶ 89]

Reference

[IConnectionStateProvider Interface \[▶ 89\]](#)

[TwinCAT Namespace \[▶ 59\]](#)

6.1.8.1.1 IConnectionStateProvider.ConnectionState Property

Gets the current Connection state of the [IConnectionStateProvider](#) [[▶ 89](#)]

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ConnectionState ConnectionState { get; }
```

Property Value

Type: [ConnectionState](#) [[▶ 72](#)]

The state of the connection.

Remarks

The Connection state changes only if the [IConnection](#) [[▶ 79](#)] is established / shut down or active communication is triggered by the User of the [IConnection](#) [[▶ 79](#)] object.

Examples

The following sample shows how to keep the ConnectionState updated by triggering ADS Communication.

Trigger ConnectionState changes in WPF Applications

```
private DispatcherTimer _timer = null;
private AdsSession _session = null;
//private AdsConnection _connection = null;

private void Window_Loaded(object sender, RoutedEventArgs e)
{
    _session = new AdsSession(AmsNetId.Local, 10000);
    IConnection connection = _session.Connect();
    tbConnectionState.Text = connection.ConnectionState.ToString();
    _session.ConnectionStateChanged += _session_ConnectionStateChanged;

    _timer = new DispatcherTimer();
    _timer.Interval = TimeSpan.FromMilliseconds(200);
    _timer.Tick += TimerOnTick;

    _timer.Start();
}

private void Window_Unloaded(object sender, RoutedEventArgs e)
{
    _timer.Stop();
    _session.Dispose();
}

private void _session_ConnectionStateChanged(object sender, TwinCAT.ConnectionStateChangedEventArgs e)
{
    // ConnectionStateChanged will be triggered by communication Invokes
    tbConnectionState.Text = e.NewState.ToString();
}

private void TimerOnTick(object sender, EventArgs eventArgs)
{
    // The Timer Event will occur here in the UI thread because its an DispatcherTimer event!
    // An active ADS request will trigger Connection State periodically!
    StateInfo stateInfo;
    if (_session.Connection.TryReadState(out stateInfo) == AdsErrorCode.NoError)
    {
        tbAdsState.Text = stateInfo.AdsState.ToString();
    }
    else
    {

```

```

    tbAdsState.Text = "Invalid";
  }
}

```

Reference

[IConnectionStateProvider Interface \[► 89\]](#)



[TwinCAT Namespace \[► 59\]](#)

[IConnectionStateProvider.ConnectionStateChanged \[► 91\]](#)

6.1.8.2 IConnectionStateProvider Events

The [IConnectionStateProvider \[► 89\]](#) type exposes the following members.

Events

	Name	Description
 	ConnectionStateChanged [► 91]	Occurs when connection status of the IConnectionStateProvider [► 89] has been changed.

Reference

[IConnectionStateProvider Interface \[► 89\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.8.2.1 IConnectionStateProvider.ConnectionStateChanged Event

Occurs when connection status of the [IConnectionStateProvider \[► 89\]](#) has been changed.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
event EventHandler<ConnectionStateChangedEventArgs> ConnectionStateChanged
```

Value

Type: [System.EventHandler.ConnectionStateChangedEventArgs \[► 74\]](#).

Remarks

The Connection state changes only if the [IConnection \[► 79\]](#) is established / shut down or active communication is triggered by the User of the [IConnection \[► 79\]](#) object.

Examples

The following sample shows how to keep the [ConnectionState \[► 90\]](#) updated by triggering ADS Communication.

Trigger ConnectionState changes in WPF Applications

```

private DispatcherTimer _timer = null;
private AdsSession _session = null;
//private AdsConnection _connection = null;

```

```

private void Window_Loaded(object sender, RoutedEventArgs e)
{
    _session = new AdsSession(AmsNetId.Local, 10000);
    IConnection connection = _session.Connect();
    tbConnectionState.Text = connection.ConnectionState.ToString();
    _session.ConnectionStateChanged += _session_ConnectionStateChanged;

    _timer = new DispatcherTimer();
    _timer.Interval = TimeSpan.FromMilliseconds(200);
    _timer.Tick += TimerOnTick;

    _timer.Start();
}

private void Window_Unloaded(object sender, RoutedEventArgs e)
{
    _timer.Stop();
    _session.Dispose();
}

private void _session_ConnectionStateChanged(object sender, TwinCAT.ConnectionStateChangedEventArgs e)
{
    // ConnectionStateChanged will be triggered by communication Invokes
    tbConnectionState.Text = e.NewState.ToString();
}

private void TimerOnTick(object sender, EventArgs eventArgs)
{
    // The Timer Event will occur here in the UI thread because its an DispatcherTimer event!
    // An active ADS request will trigger Connection State periodically!
    StateInfo stateInfo;
    if (_session.Connection.TryReadState(out stateInfo) == AdsErrorCode.NoError)
    {
        tbAdsState.Text = stateInfo.AdsState.ToString();
    }
    else
    {
        tbAdsState.Text = "Invalid";
    }
}

```

Reference

[IConnectionStateProvider Interface](#) [► 89]

[TwinCAT Namespace](#) [► 59]

[IConnectionStateProvider.ConnectionState](#) [► 90]

6.1.9 ISession Interface

Interface ISession

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax








C#

```
public interface ISession : IConnectionStateProvider
```






The ISession type exposes the following members.

Properties



	Name	Description
	AddressSpecifier [► 94]	Gets the communication endpoint address string representation.

	Name	Description
	Connection [▶ 94]	Gets the Connection object.
 	ConnectionState [▶ 90]	Gets the current Connection state of the IConnectionStateProvider [▶ 89] (Inherited from IConnectionStateProvider [▶ 89].)
	EstablishedAt [▶ 95]	Gets the UTC time when the session was established.
	Id [▶ 95]	Gets the Session Id
	IsConnected [▶ 96]	Gets a value indicating whether the session is connected.
	Provider [▶ 96]	Gets the Session Provider

Methods

	Name	Description
	Close [▶ 97]	Closes this ISession
	Connect [▶ 97]	Connects the session and returns the established IConnection [▶ 79] object.
	ConnectAsync [▶ 99]	Connects the session and returns the established IConnection [▶ 79] object.
	Disconnect [▶ 98]	Disconnects the ISession
	EnsureConnection [▶ 98]	Ensures, that the ISession is connected and returns the IConnection [▶ 79] object.

Events

	Name	Description
 	ConnectionStateChanged [▶ 91]	Occurs when connection status of the IConnectionStateProvider [▶ 89] has been changed. (Inherited from IConnectionStateProvider [▶ 89].)





Reference





[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.9.1 ISession Properties

The [ISession](#) [[▶ 92](#)] type exposes the following members.

Properties

	Name	Description
	AddressSpecifier [▶ 94]	Gets the communication endpoint address string representation.
	Connection [▶ 94]	Gets the Connection object.
 	ConnectionState [▶ 90]	Gets the current Connection state of the IConnectionStateProvider [▶ 89] (Inherited from IConnectionStateProvider [▶ 89].)

	Name	Description
	EstablishedAt [▶ 95]	Gets the UTC time when the session was established.
	Id [▶ 95]	Gets the Session Id
	IsConnected [▶ 96]	Gets a value indicating whether the session is connected.
	Provider [▶ 96]	Gets the Session Provider

Reference

[ISession Interface](#) [▶ 92]

[TwinCAT Namespace](#) [▶ 59]

6.1.9.1.1 ISession.AddressSpecifier Property

Gets the communication endpoint address string representation.

Namespace: [TwinCAT](#) [▶ 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string AddressSpecifier { get; }
```

Property Value

Type: [String](#)
The address.

Reference

[ISession Interface](#) [▶ 92]

[TwinCAT Namespace](#) [▶ 59]

6.1.9.1.2 ISession.Connection Property

Gets the Connection object.

Namespace: [TwinCAT](#) [▶ 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IConnection? Connection { get; }
```

Property Value

Type: [IConnection](#) [▶ 79]
The connection.

Remarks

The [IConnection](#) [▸ 79] object is established by the [ISession](#) [▸ 92] via [Connect.](#) [▸ 97] and is valid until the [Disconnect.](#) [▸ 98] method or the Dispose method is called. Any possible resurrections after communication losses will be done transparently under the hood of the [IConnection](#) [▸ 79] so that the [IConnection](#) [▸ 79] instance and [ISession](#) [▸ 92] instance.

Reference

[ISession Interface](#) [▸ 92]

[TwinCAT Namespace](#) [▸ 59]

6.1.9.1.3 ISession.EstablishedAt Property

Gets the UTC time when the session was established.

Namespace: [TwinCAT](#) [▸ 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
DateTimeOffset EstablishedAt { get; }
```

Property Value

Type: [DateTimeOffset](#)

The session established at.

Reference

[ISession Interface](#) [▸ 92]

[TwinCAT Namespace](#) [▸ 59]

6.1.9.1.4 ISession.Id Property

Gets the Session Id

Namespace: [TwinCAT](#) [▸ 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Id { get; }
```

Property Value

Type: [Int32](#)

The identifier.

Reference

[ISession Interface](#) [▸ 92]

[TwinCAT Namespace](#) [▸ 59]

6.1.9.1.5 ISession.IsConnected Property

Gets a value indicating whether the session is connected.

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsConnected { get; }
```

Property Value

Type: [Boolean](#)

true if session is connected; otherwise, false.

Reference

[ISession Interface](#) [[▶ 92](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.9.1.6 ISession.Provider Property

Gets the Session Provider

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISessionProvider Provider { get; }
```

Property Value

Type: [ISessionProvider](#) [[▶ 99](#)]

The provider.

Reference




[ISession Interface](#) [[▶ 92](#)]



[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.9.2 ISession Methods

The [ISession](#) [[▶ 92](#)] type exposes the following members.

Methods

	Name	Description
	Close [▶ 97]	Closes this ISession [▶ 92]
	Connect [▶ 97]	Connects the session and returns the established IConnection [▶ 79] object.
	ConnectAsync [▶ 99]	Connects the session and returns the established IConnection [▶ 79] object.

	Name	Description
	Disconnect [▶ 98]	Disconnects the ISession [▶ 92]
	EnsureConnection [▶ 98]	Ensures, that the ISession [▶ 92] is connected and returns the IConnection [▶ 79] object.

Reference

[ISession Interface](#) [[▶ 92](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.9.2.1 ISession.Close Method

Closes this [ISession](#) [[▶ 92](#)]

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void Close()
```

Remarks

Closes also the [IConnection](#) [[▶ 79](#)].

Reference

[ISession Interface](#) [[▶ 92](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.9.2.2 ISession.Connect Method

Connects the session and returns the established [IConnection](#) [[▶ 79](#)] object.

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IConnection Connect()
```

Return Value

Type: [IConnection](#) [[▶ 79](#)]

The [IConnection](#) [[▶ 79](#)] object.

Remarks

The [IConnection](#) [[▶ 79](#)] will be valid until the [ISession](#) [[▶ 92](#)] is disconnected via the [Disconnect](#). [[▶ 98](#)] method or the Dispose method is called. Any possible resurrections after communication losses will be done transparently within the [IConnection](#) [[▶ 79](#)] so that the [IConnection](#) [[▶ 79](#)] instance and [ISession](#) [[▶ 92](#)] instance remains.

Reference

[ISession Interface \[▸ 92\]](#)

[TwinCAT Namespace \[▸ 59\]](#)

6.1.9.2.3 ISession.Disconnect Method

Disconnects the [ISession \[▸ 92\]](#)

Namespace: [TwinCAT \[▸ 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool Disconnect()
```

Return Value

Type: [Boolean](#)

true if Session was disconnected, false if the session was already closed.

Remarks

Disposes also the [IConnection \[▸ 79\]](#).

Reference

[ISession Interface \[▸ 92\]](#)

[TwinCAT Namespace \[▸ 59\]](#)

6.1.9.2.4 ISession.EnsureConnection Method

Ensures, that the [ISession \[▸ 92\]](#) is connected and returns the [IConnection \[▸ 79\]](#) object.

Namespace: [TwinCAT \[▸ 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IConnection EnsureConnection()
```

Return Value

Type: [IConnection \[▸ 79\]](#)

IConnection.

Remarks

If the session is actually not connected an exception will be thrown.

Reference

[ISession Interface \[▸ 92\]](#)

[TwinCAT Namespace \[▸ 59\]](#)

6.1.9.2.5 ISession.ConnectAsync Method

Connects the session and returns the established [IConnection](#) [[79](#)] object.

Namespace: [TwinCAT](#) [[59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<IConnection> ConnectAsync (
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken
--------	--

Return Value

Type: [Task.IConnection](#) [[79](#)].

The [IConnection](#) [[79](#)] object.

Remarks

The [IConnection](#) [[79](#)] will be valid until the [ISession](#) [[92](#)] is disconnected via the [Disconnect](#) [[98](#)] method or the Dispose method is called. Any possible resurrections after communication losses will be done transparently within the [IConnection](#) [[79](#)] so that the [IConnection](#) [[79](#)] instance and [ISession](#) [[92](#)] instance remains.

Reference



[ISession Interface](#) [[92](#)]

[TwinCAT Namespace](#) [[59](#)]

6.1.9.3 ISession Events

The [ISession](#) [[92](#)] type exposes the following members.

Events

	Name	Description
 	ConnectionStateChanged [91]	Occurs when connection status of the IConnectionStateProvider [89] has been changed. (Inherited from IConnectionStateProvider [89].)

Reference

[ISession Interface](#) [[92](#)]

[TwinCAT Namespace](#) [[59](#)]

6.1.10 IServiceProvider Interface

Interface [ISessionProvider](#)

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax

C#


```
public interface ISessionProvider
```

The ISessionProvider type exposes the following members.

Properties

	Name	Description
	Capabilities [► 100]	Gets the capabilities.
	Name [► 101]	Gets the name of the SessionProvider

Methods

	Name	Description
	Create [► 101]	Creates the Session with address and communication settings.

Remarks

Interface for Supporting Session / Communication providers



Reference

[TwinCAT Namespace \[► 59\]](#)

6.1.10.1 ISessionProvider Properties

The [ISessionProvider \[► 99\]](#) type exposes the following members.

Properties

	Name	Description
	Capabilities [► 100]	Gets the capabilities.
	Name [► 101]	Gets the name of the SessionProvider

Reference

[ISessionProvider Interface \[► 99\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.10.1.1 ISessionProvider.Capabilities Property

Gets the capabilities.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
SessionProviderCapabilities Capabilities { get; }
```

Property Value

Type: [SessionProviderCapabilities](#) [[▶](#) [148](#)]

The capabilities.

Reference

[ISessionProvider Interface](#) [[▶](#) [99](#)]

[TwinCAT Namespace](#) [[▶](#) [59](#)]

6.1.10.1.2 ISessionProvider.Name Property

Gets the name of the SessionProvider

Namespace: [TwinCAT](#) [[▶](#) [59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string Name { get; }
```

Property Value

Type: [String](#)

The name.

Reference


[ISessionProvider Interface](#) [[▶](#) [99](#)]

[TwinCAT Namespace](#) [[▶](#) [59](#)]

6.1.10.2 ISessionProvider Methods

The [ISessionProvider](#) [[▶](#) [99](#)] type exposes the following members.

Methods

	Name	Description
	Create [▶ 101]	Creates the Session with address and communication settings.

Reference

[ISessionProvider Interface](#) [[▶](#) [99](#)]

[TwinCAT Namespace](#) [[▶](#) [59](#)]

6.1.10.2.1 ISessionProvider.Create Method

Creates the Session with address and communication settings.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISession Create(
    Object address,
    ISessionSettings? settings
)
```

Parameters

address	Type: System.Object The address.
settings	Type: ISessionSettings The settings.

Return Value

Type: [ISession \[► 92\]](#)

ISession.

Reference

[ISessionProvider Interface \[► 99\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.11 ISessionProvider.TSession, TAddress, TSettings. Interface

Generic ISessionProvider interface

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#



```
public interface ISessionProvider<TSession, TAddress, TSettings> : ISessionProvider
where TSession : ISession
```

Type Parameters



TSession	Session type
TAddress	Address specifier
TSettings	Communication settings type

The ISessionProvider.TSession, TAddress, TSettings. type exposes the following members.

Properties

	Name	Description
	Capabilities [► 100]	Gets the capabilities. (Inherited from ISessionProvider [► 99].)
	Name [► 101]	Gets the name of the SessionProvider (Inherited from ISessionProvider [► 99].)

Methods

	Name	Description
	Create(TAddress, TSettings) [▶ 104]	Creates the Session with specified address and communication settings.
	Create(Object, ISessionSettings) [▶ 101]	Creates the Session with address and communication settings. (Inherited from ISessionProvider [▶ 99].)

Reference



[TwinCAT Namespace](#) [[▶ 59](#)]

[TwinCAT.ISessionProvider](#) [[▶ 99](#)]

6.1.11.1 ISessionProvider.TSession, TAddress, TSettings. Properties

The [ISessionProvider.TSession, TAddress, TSettings](#). [[▶ 102](#)] generic type exposes the following members.

Properties

	Name	Description
	Capabilities [▶ 100]	Gets the capabilities. (Inherited from ISessionProvider [▶ 99].)
	Name [▶ 101]	Gets the name of the SessionProvider (Inherited from ISessionProvider [▶ 99].)



Reference

[ISessionProvider.TSession, TAddress, TSettings. Interface](#) [[▶ 102](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.11.2 ISessionProvider.TSession, TAddress, TSettings. Methods

Methods

	Name	Description
	Create(TAddress, TSettings) [▶ 104]	Creates the Session with specified address and communication settings.
	Create(Object, ISessionSettings) [▶ 101]	Creates the Session with address and communication settings. (Inherited from ISessionProvider [▶ 99].)



Reference

[ISessionProvider.TSession, TAddress, TSettings. Interface](#) [[▶ 102](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.11.2.1 IServiceProvider.TSession, TAddress, TSettings..Create Method

Overload List

	Name	Description
	Create(TAddress, TSettings) [▶ 104]	Creates the Session with specified address and communication settings.
	Create(Object, ISessionSettings) [▶ 101]	Creates the Session with address and communication settings. (Inherited from ISessionProvider [▶ 99].)

Reference

[ISessionProvider.TSession, TAddress, TSettings. Interface](#) [[▶ 102](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.11.2.1.1 IServiceProvider.TSession, TAddress, TSettings..Create Method (TAddress, TSettings)

Creates the Session with specified address and communication settings.

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
TSession Create(
    TAddress address,
    TSettings settings
)
```

Parameters

address	Type: TAddress [▶ 102] The address.
settings	Type: TSettings [▶ 102] The communicationSettings.

Return Value

Type: [TSession](#) [[▶ 102](#)]
S.

Reference

[ISessionProvider.TSession, TAddress, TSettings. Interface](#) [[▶ 102](#)]

[Create Overload](#) [[▶ 104](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.12 ISymbolLoaderSettings Interface

Interface ISymbolLoaderSettings

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public interface ISymbolLoaderSettings
```

The ISymbolLoaderSettings type exposes the following members.

Properties

	Name	Description
	SymbolsLoadMode [▶ 105]	Gets or sets the symbols load mode.

Reference

[TwinCAT Namespace](#) [▶ 59]

6.1.12.1 ISymbolLoaderSettings Properties

The [ISymbolLoaderSettings](#) [▶ 104] type exposes the following members.

Properties

	Name	Description
	SymbolsLoadMode [▶ 105]	Gets or sets the symbols load mode.

Reference

[ISymbolLoaderSettings Interface](#) [▶ 104]

[TwinCAT Namespace](#) [▶ 59]

6.1.12.1.1 ISymbolLoaderSettings.SymbolsLoadMode Property

Gets or sets the symbols load mode.

Namespace: [TwinCAT](#) [▶ 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
SymbolsLoadMode SymbolsLoadMode { get; set; }
```

Property Value

Type: [SymbolsLoadMode](#) [▶ 159]

The symbols load mode.

Reference

[ISymbolLoaderSettings Interface](#) [▶ 104]

[TwinCAT Namespace](#) [▶ 59]

6.1.13 ISymbolServerProvider Interface

Interface ISymbolServerProvider

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#

```
public interface ISymbolServerProvider
```

The ISymbolServerProvider type exposes the following members.

Properties

	Name	Description
	SymbolServer [► 106]	Gets the symbol server.


Reference

[TwinCAT Namespace \[► 59\]](#)

6.1.13.1 ISymbolServerProvider Properties

The [ISymbolServerProvider \[► 106\]](#) type exposes the following members.

Properties

	Name	Description
	SymbolServer [► 106]	Gets the symbol server.

Reference

[ISymbolServerProvider Interface \[► 106\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.13.1.1 ISymbolServerProvider.SymbolServer Property

Gets the symbol server.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISymbolServer SymbolServer { get; }
```

Property Value

Type: [ISymbolServer \[► 2719\]](#)

The symbol server. To initially create this information the connection must be established.

Reference

[ISymbolServerProvider Interface \[▶ 106\]](#)

[TwinCAT Namespace \[▶ 59\]](#)

6.1.14 Session Class

Abstract Session base class.

Inheritance Hierarchy

System.Object
 TwinCAT.Session
 TwinCAT.Ads.AdsSessionBase [▶ 716]

Namespace: [TwinCAT \[▶ 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229











Syntax

C#



```
public abstract class Session : ISession,
    IConnectionStateProvider, ISymbolServerProvider, IDisposable
```



















The Session type exposes the following members.

Properties


	Name	Description
	AddressSpecifier [▶ 109]	Gets the communication endpoint address string representation.
	Connection [▶ 110]	Gets the (established) connection.
 	ConnectionState [▶ 110]	Gets the current Connection state of the Session
	Disposed [▶ 111]	Gets a value indicating whether this Session is disposed.
	EstablishedAt [▶ 112]	Gets the UTC time when the session was established.
	Id [▶ 112]	Gets the Session Identifier
	IsConnected [▶ 113]	Gets a value indicating whether this instance is connected.
	Name [▶ 113]	Gets the name of the session
	SymbolServer [▶ 113]	Gets the symbol server.

Methods

	Name	Description
	Close [▶ 115]	Closes this ISession [▶ 92]
	Connect [▶ 116]	Connects the session.

	Name	Description
	ConnectAsync [▶ 122]	Connects the session.
	Disconnect [▶ 116]	Disconnects the session from the target.
	Dispose. [▶ 117]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.
	Dispose(Boolean) [▶ 118]	Releases unmanaged and - optionally - managed resources.
	EnsureConnection [▶ 120]	Ensures, that the ISession [▶ 92] is connected and returns the IConnection [▶ 79] object.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetSessionName [▶ 118]	Gets the name/string identifier of the session.
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnConnect [▶ 119]	Handler function connecting the Session.
	OnConnectAsync [▶ 123]	Handler function connecting the Session.
	OnConnectionState Changed [▶ 121]	Handles the [E:ConnectionStateChanged] event.
	OnCreateSymbolSer ver [▶ 119]	Handler function creating the ISymbolServer [▶ 2719]
	OnDisconnect [▶ 120]	Handler function disconnecting the session.
	OnGetAddress [▶ 120]	Handler function getting the address of the session.
	ToString [▶ 121]	Returns a String that represents this instance. (Overrides Object.ToString .)

Events

	Name	Description
	ConnectionStateCha nged [▶ 124]	Occurs when connection status of the IConnectionStateProvider [▶ 89] has been changed.

Reference

[TwinCAT Namespace](#) [▶ 59]











[TwinCAT.ISession](#) [▶ 92]

[System.IDisposable](#)

6.1.14.1 Session Properties

The [Session](#) [[▶ 107](#)] type exposes the following members.

Properties

	Name	Description
	AddressSpecifier [▶ 109]	Gets the communication endpoint address string representation.
	Connection [▶ 110]	Gets the (established) connection.
 	ConnectionState [▶ 110]	Gets the current Connection state of the Session [▶ 107]
	Disposed [▶ 111]	Gets a value indicating whether this Session [▶ 107] is disposed.
	EstablishedAt [▶ 112]	Gets the UTC time when the session was established.
	Id [▶ 112]	Gets the Session Identifier
	IsConnected [▶ 113]	Gets a value indicating whether this instance is connected.
	Name [▶ 113]	Gets the name of the session
	SymbolServer [▶ 113]	Gets the symbol server.

Reference

[Session Class](#) [[▶ 107](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.14.1.1 Session.AddressSpecifier Property

Gets the communication endpoint address string representation.

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string AddressSpecifier { get; }
```

Property Value

Type: [String](#)
The address.

Implements

[ISession.AddressSpecifier](#) [[▶ 94](#)]

Reference

[Session Class](#) [► 107]

[TwinCAT Namespace](#) [► 59]

6.1.14.1.2 Session.Connection Property

Gets the (established) connection.

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IConnection? Connection { get; protected set; }
```

Property Value

Type: [IConnection](#) [► 79]

The [IConnection](#) [► 79] if connection established, or **null** if not connected.

Implements

[ISession.Connection](#) [► 94]

Reference

[Session Class](#) [► 107]

[TwinCAT Namespace](#) [► 59]

6.1.14.1.3 Session.ConnectionState Property

Gets the current Connection state of the [Session](#) [► 107]

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ConnectionState ConnectionState { get; }
```

Property Value

Type: [ConnectionState](#) [► 72]

The state of the connection.

Implements

[IConnectionStateProvider.ConnectionState](#) [► 90]

Remarks

The Connection state changes only if the [IConnection](#) [► 79] is established / shut down or active communication is triggered by the User of the [IConnection](#) [► 79] object.

Examples

The following sample shows how to keep the ConnectionState updated by triggering ADS Communication.

Trigger ConnectionState changes in WPF Applications

```
private DispatcherTimer _timer = null;
private AdsSession _session = null;
//private AdsConnection _connection = null;

private void Window_Loaded(object sender, RoutedEventArgs e)
{
    _session = new AdsSession(AmsNetId.Local, 10000);
    IConnection connection = _session.Connect();
    tbConnectionState.Text = connection.ConnectionState.ToString();
    _session.ConnectionStateChanged += _session_ConnectionStateChanged;

    _timer = new DispatcherTimer();
    _timer.Interval = TimeSpan.FromMilliseconds(200);
    _timer.Tick += TimerOnTick;

    _timer.Start();
}

private void Window_Unloaded(object sender, RoutedEventArgs e)
{
    _timer.Stop();
    _session.Dispose();
}

private void _session_ConnectionStateChanged(object sender, TwinCAT.ConnectionStateChangedEventArgs e)
{
    // ConnectionStateChanged will be triggered by communication Invokes
    tbConnectionState.Text = e.NewState.ToString();
}

private void TimerOnTick(object sender, EventArgs eventArgs)
{
    // The Timer Event will occur here in the UI thread because its an DispatcherTimer event!
    // An active ADS request will trigger Connection State periodically!
    StateInfo stateInfo;
    if (_session.Connection.TryReadState(out stateInfo) == AdsErrorCode.NoError)
    {
        tbAdsState.Text = stateInfo.AdsState.ToString();
    }
    else
    {
        tbAdsState.Text = "Invalid";
    }
}
}
```

Reference

[Session Class \[► 107\]](#)

[TwinCAT Namespace \[► 59\]](#)

[Session.ConnectionStateChanged \[► 124\]](#)

6.1.14.1.4 Session.Disposed Property

Gets a value indicating whether this [Session \[► 107\]](#) is disposed.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Disposed { get; }
```

Property Value

Type: [Boolean](#)
true if disposed; otherwise, false.

Reference

[Session Class](#) [► 107]

[TwinCAT Namespace](#) [► 59]

6.1.14.1.5 Session.EstablishedAt Property

Gets the UTC time when the session was established.

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTimeOffset EstablishedAt { get; }
```

Property Value

Type: [DateTimeOffset](#)
The session established at.

Implements

[ISession.EstablishedAt](#) [► 95]

Reference

[Session Class](#) [► 107]

[TwinCAT Namespace](#) [► 59]

6.1.14.1.6 Session.Id Property

Gets the Session Identifier

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Id { get; }
```

Property Value

Type: [Int32](#)
The identifier.

Implements

[ISession.Id](#) [► 95]

Reference

[Session Class](#) [▶ 107]

[TwinCAT Namespace](#) [▶ 59]

6.1.14.1.7 Session.IsConnected Property

Gets a value indicating whether this instance is connected.

Namespace: [TwinCAT](#) [▶ 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsConnected { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is connected; otherwise, false.

Implements

[ISession.IsConnected](#) [▶ 96]

Reference

[Session Class](#) [▶ 107]

[TwinCAT Namespace](#) [▶ 59]

6.1.14.1.8 Session.Name Property

Gets the name of the session

Namespace: [TwinCAT](#) [▶ 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Name { get; }
```

Property Value

Type: [String](#)

The name.

Reference

[Session Class](#) [▶ 107]

[TwinCAT Namespace](#) [▶ 59]

6.1.14.1.9 Session.SymbolServer Property

Gets the symbol server.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISymbolServer SymbolServer { get; }
```

Property Value

Type: [ISymbolServer \[► 2719\]](#)

The symbol server.

Implements

[ISymbolServerProvider.SymbolServer \[► 106\]](#)

Remarks

The [Session \[► 107\]](#) object holds and caches the symbolic information. To initially create this information, the Connection must be established.

Reference











[Session Class \[► 107\]](#)











[TwinCAT Namespace \[► 59\]](#)

6.1.14.2 Session Methods

The [Session \[► 107\]](#) type exposes the following members.

Methods

	Name	Description
	Close [► 115]	Closes this ISession [► 92]
	Connect [► 116]	Connects the session.
	ConnectAsync [► 122]	Connects the session.
	Disconnect [► 116]	Disconnects the session from the target.
	Dispose. [► 117]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.
	Dispose(Boolean) [► 118]	Releases unmanaged and - optionally - managed resources.
	EnsureConnection [► 120]	Ensures, that the ISession [► 92] is connected and returns the IConnection [► 79] object.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)

	Name	Description
	GetSessionName [▶ 118]	Gets the name/string identifier of the session.
	GetType	Gets the <u>Type</u> of the current instance. (Inherited from <u>Object</u> .)
	MemberwiseClone	Creates a shallow copy of the current <u>Object</u> . (Inherited from <u>Object</u> .)
	OnConnect [▶ 119]	Handler function connecting the Session.
	OnConnectAsync [▶ 123]	Handler function connecting the Session.
	OnConnectionState Changed [▶ 121]	Handles the [E:ConnectionStateChanged] event.
	OnCreateSymbolSer ver [▶ 119]	Handler function creating the <u>ISymbolServer</u> [▶ 2719]
	OnDisconnect [▶ 120]	Handler function disconnecting the session.
	OnGetAddress [▶ 120]	Handler function getting the address of the session.
	ToString [▶ 121]	Returns a <u>String</u> that represents this instance. (Overrides <u>Object.ToString</u> ..)

Reference

[Session Class](#) [▶ 107]

[TwinCAT Namespace](#) [▶ 59]

6.1.14.2.1 Session.Close Method

Closes this ISession [▶ 92]

Namespace: [TwinCAT](#) [▶ 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Close()
```

Implements

[ISession.Close](#). [▶ 97]

Remarks

Closes also the IConnection [▶ 79].

Reference

[Session Class](#) [▶ 107]

[TwinCAT Namespace](#) [▶ 59]

6.1.14.2 Session.Connect Method

Connects the session.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IConnection Connect()
```

Return Value

Type: [IConnection \[► 79\]](#)

The [IConnection \[► 79\]](#) object.

Implements

[ISession.Connect. \[► 97\]](#)

Exceptions

Exception	Condition
ObjectDisposedException	
AdsException [► 61]	Connection to '{this.AddressSpecifier}' couldn't be established!

Remarks


The [IConnection \[► 79\]](#) will be valid until the [ISession \[► 92\]](#) is disconnected via the [Disconnect. \[► 98\]](#) method or the [Dispose](#) method is called. Any possible resurrections after communication losses will be done transparently within the [IConnection \[► 79\]](#) so that the [IConnection \[► 79\]](#) instance and [ISession \[► 92\]](#) instance remains.

Reference

[Session Class \[► 107\]](#)

[TwinCAT Namespace \[► 59\]](#)

Also see about this

 [Session.Disconnect Method \[► 116\]](#)

6.1.14.3 Session.Disconnect Method

Disconnects the session from the target.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Disconnect()
```

Return Value

Type: [Boolean](#)

true if the [Session](#) [[▶ 107](#)] is disconnected, false otherwise (the connection was disconnected before)

Implements

[ISession.Disconnect.](#) [[▶ 98](#)]

Exceptions

Exception	Condition
ObjectDisposedException	

Remarks

Closes (and disposes) the underlying [IConnection](#) [[▶ 79](#)] The [Session](#) [[▶ 107](#)] itself will not be Disposed and can be reconnected.



Reference

[Session Class](#) [[▶ 107](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.14.2.4 Session.Dispose Method

Overload List

	Name	Description
	Dispose. [▶ 117]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.
	Dispose(Boolean) [▶ 118]	Releases unmanaged and - optionally - managed resources.

Reference

[Session Class](#) [[▶ 107](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.14.2.4.1 Session.Dispose Method

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Dispose()
```

Implements

[IDisposable.Dispose.](#)

Reference

[Session Class](#) [► 107]

[Dispose Overload](#) [► 117]

[TwinCAT Namespace](#) [► 59]

6.1.14.2.4.2 Session.Dispose Method (Boolean)

Releases unmanaged and - optionally - managed resources.

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual void Dispose(  
    bool disposing  
)
```

Parameters

disposing	Type: System.Boolean true to release both managed and unmanaged resources; false to release only unmanaged resources.
-----------	--

Reference

[Session Class](#) [► 107]

[Dispose Overload](#) [► 117]

[TwinCAT Namespace](#) [► 59]

6.1.14.2.5 Session.GetSessionName Method

Gets the name/string identifier of the session.

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected abstract string GetSessionName()
```

Return Value

Type: [String](#)
System.String.

Reference

[Session Class](#) [► 107]

[TwinCAT Namespace](#) [► 59]

6.1.14.2.6 Session.OnConnect Method

Handler function connecting the Session.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual IConnection? OnConnect(
    bool reconnect
)
```

Parameters

reconnect	Type: System.Boolean if set to true [reconnect].
-----------	---

Return Value

Type: [IConnection \[► 79\]](#)

IConnection.

Reference

[Session Class \[► 107\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.14.2.7 Session.OnCreateSymbolServer Method

Handler function creating the [ISymbolServer \[► 2719\]](#)

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected abstract ISymbolServer OnCreateSymbolServer()
```

Return Value

Type: [ISymbolServer \[► 2719\]](#)

ISymbolServer.

Exceptions

Exception	Condition
SessionNotConnectedException [► 137]	The connection is not established!

Reference

[Session Class \[► 107\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.14.2.8 Session.OnDisconnect Method

Handler function disconnecting the session.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual bool OnDisconnect()
```

Return Value

Type: [Boolean](#)

true if the [Session \[► 107\]](#) is disconnected, false if the [Session \[► 107\]](#) was disconnected before.

Reference

[Session Class \[► 107\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.14.2.9 Session.OnGetAddress Method

Handler function getting the address of the session.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected abstract string OnGetAddress()
```

Return Value

Type: [String](#)

System.String.

Reference

[Session Class \[► 107\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.14.2.10 Session.EnsureConnection Method

Ensures, that the [ISession \[► 92\]](#) is connected and returns the [IConnection \[► 79\]](#) object.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IConnection EnsureConnection()
```


Return Value

Type: [IConnection](#) [[▶ 79](#)]
 IConnection.

Implements

[ISession.EnsureConnection.](#) [[▶ 98](#)]

Exceptions

Exception	Condition
NotImplementedException	

Remarks

If the session is actually not connected an exception will be thrown.

Reference

[Session Class](#) [[▶ 107](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.14.2.11 Session.OnConnectionStateChanged Method

Handles the [\[E:ConnectionStateChanged\]](#) event.

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected void OnConnectionStateChanged(
    Object? sender,
    ConnectionStateChangedEventArgs e
)
```

Parameters

sender	Type: System.Object The sender.
e	Type: TwinCAT.ConnectionStateChangedEventArgs [▶ 74] The ConnectionStateChangedEventArgs [▶ 74] instance containing the event data.

Reference

[Session Class](#) [[▶ 107](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.14.2.12 Session.ToString Method

Returns a [String](#) that represents this instance.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```

Return Value

Type: [String](#)

A [String](#) that represents this instance.

Reference

[Session Class \[► 107\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.14.2.13 Session.ConnectAsync Method

Connects the session.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<IConnection> ConnectAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token that can be used by other objects or threads to receive notice of cancellation.
--------	--

Return Value

Type: [Task.IConnection \[► 79\]](#).

A [Task<IConnection>](#) representing the asynchronous operation.

Implements

[ISession.ConnectAsync\(CancellationToken\) \[► 99\]](#)

Exceptions

Exception	Condition
ObjectDisposedException	
AdsException [► 61]	Connection to '{this.AddressSpecifier}' couldn't be established!

Remarks

The [IConnection](#) [▶ 79] will be valid until the [ISession](#) [▶ 92] is disconnected via the [Disconnect](#) [▶ 98] method or the [Dispose](#) method is called. Any possible resurrections after communication losses will be done transparently within the [IConnection](#) [▶ 79] so that the [IConnection](#) [▶ 79] instance and [ISession](#) [▶ 92] instance remains.

Reference

[Session Class](#) [▶ 107]

[TwinCAT Namespace](#) [▶ 59]

6.1.14.2.14 Session.OnConnectAsync Method

Handler function connecting the Session.

Namespace: [TwinCAT](#) [▶ 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<IConnection?> OnConnectAsync (
    bool reconnect,
    CancellationToken cancel
)
```

Parameters

reconnect	Type: System.Boolean if set to true [reconnect].
cancel	Type: System.Threading.CancellationToken The cancellation token that can be used by other objects or threads to receive notice of cancellation.

Return Value

Type: [Task.IConnection](#) [▶ 79].
[Task<System.Nullable<IConnection>>](#).

Reference


[Session Class](#) [▶ 107]

[TwinCAT Namespace](#) [▶ 59]

6.1.14.3 Session Events

The [Session](#) [▶ 107] type exposes the following members.

Events

	Name	Description
	ConnectionStateChanged [▶ 124]	Occurs when connection status of the IConnectionStateProvider [▶ 89] has been changed.

Reference

[Session Class](#) [► 107]

[TwinCAT Namespace](#) [► 59]

6.1.14.3.1 Session.ConnectionStateChanged Event

Occurs when connection status of the [IConnectionStateProvider](#) [► 89] has been changed.

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<ConnectionStateChangedEventArgs> ConnectionStateChanged
```

Value

Type: [System.EventHandler.ConnectionStateChangedEventArgs](#) [► 74].

Implements

[IConnectionStateProvider.ConnectionStateChanged](#) [► 91]

Remarks

The Connection state changes only if the [IConnection](#) [► 79] is established / shut down or active communication is triggered by the User of the [IConnection](#) [► 79] object.

Examples

The following sample shows how to keep the [ConnectionState](#) [► 110] updated by triggering ADS Communication.

Trigger ConnectionState changes in WPF Applications

```
private DispatcherTimer _timer = null;
private AdsSession _session = null;
//private AdsConnection _connection = null;

private void Window_Loaded(object sender, RoutedEventArgs e)
{
    _session = new AdsSession(AmsNetId.Local, 10000);
    IConnection connection = _session.Connect();
    tbConnectionState.Text = connection.ConnectionState.ToString();
    _session.ConnectionStateChanged += _session_ConnectionStateChanged;

    _timer = new DispatcherTimer();
    _timer.Interval = TimeSpan.FromMilliseconds(200);
    _timer.Tick += TimerOnTick;

    _timer.Start();
}

private void Window_Unloaded(object sender, RoutedEventArgs e)
{
    _timer.Stop();
    _session.Dispose();
}

private void _session_ConnectionStateChanged(object sender, TwinCAT.ConnectionStateChangedEventArgs e)
{
    // ConnectionStateChanged will be triggered by communication Invokes
    tbConnectionState.Text = e.NewState.ToString();
}
```

```

}

private void TimerOnTick(object sender, EventArgs eventArgs)
{
    // The Timer Event will occur here in the UI thread because its an DispatcherTimer event!
    // An active ADS request will trigger Connection State periodically!
    StateInfo stateInfo;
    if (_session.Connection.TryReadState(out stateInfo) == AdsErrorCode.NoError)
    {
        tbAdsState.Text = stateInfo.AdsState.ToString();
    }
    else
    {
        tbAdsState.Text = "Invalid";
    }
}
}

```

Reference

[Session Class](#) [▶ 107]

[TwinCAT Namespace](#) [▶ 59]

[Session.ConnectionState](#) [▶ 110]

6.1.15 SessionConnectionStateChangedEventArgs Class

EventArguments for the ConnectionStatusChanged events.

Inheritance Hierarchy

- System.Object
- System.EventArgs
- TwinCAT.ConnectionStateChangedEventArgs [▶ 74]
- TwinCAT.SessionConnectionStateChangedEventArgs

Namespace: TwinCAT [▶ 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax

C#

```
public class SessionConnectionStateChangedEventArgs : ConnectionStateChangedEventArgs
```







The SessionConnectionStateChangedEventArgs type exposes the following members.

Constructors







	Name	Description
	SessionConnectionStateChangedEventArgs(ConnectionState ChangedReason, ConnectionState, ConnectionState, ISession, IConnection) [▶ 127]	Initializes a new instance of the SessionConnectionStateChangedEventArgs class.
	SessionConnectionStateChangedEventArgs(ConnectionState ChangedReason,	Initializes a new instance of the SessionConnectionStateChangedEventArgs class.

	Name	Description
	ConnectionState , ConnectionState , ISession , IConnection , Exception) [▶ 128]	

Properties

	Name	Description
	Connection [▶ 129]	The connection
	Exception [▶ 77]	Exception, (only for Error [▶ 79]) (Inherited from ConnectionStateChangedEventArgs [▶ 74].)
	NewState [▶ 77]	New connection state (Inherited from ConnectionStateChangedEventArgs [▶ 74].)
	OldState [▶ 78]	Old connection state (Inherited from ConnectionStateChangedEventArgs [▶ 74].)
	Reason [▶ 78]	Reason for the event (Inherited from ConnectionStateChangedEventArgs [▶ 74].)
	Session [▶ 129]	The session

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)


Reference


[TwinCAT Namespace](#) [[▶ 59](#)]

[TwinCAT.ConnectionStateChangedEventArgs](#) [[▶ 74](#)]

6.1.15.1 SessionConnectionStateChangedEventArgs Constructor

Overload List

	Name	Description
	SessionConnectionState StateChangedEventA rgs(ConnectionState ChangedReason,	Initializes a new instance of the SessionConnectionStateChangedEventArgs [▶ 125] class.

	Name	Description
	ConnectionState , ConnectionState , ISession , IConnection) [▶ 127]	
	SessionConnectionStateChangedEventArgs (ConnectionState , ConnectionState , ConnectionState , ISession , IConnection , Exception) [▶ 128]	Initializes a new instance of the SessionConnectionStateChangedEventArgs [▶ 125] class.

Reference

[SessionConnectionStateChangedEventArgs Class](#) [[▶ 125](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.15.1 SessionConnectionStateChangedEventArgs Constructor (ConnectionState, ConnectionState, ConnectionState, ISession, IConnection)

Initializes a new instance of the [SessionConnectionStateChangedEventArgs](#) [[▶ 125](#)] class.

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public SessionConnectionStateChangedEventArgs (
    ConnectionStateChangedReason reason,
    ConnectionState newState,
    ConnectionState oldState,
    ISession session,
    IConnection connection
)
```

Parameters

reason	Type: TwinCAT.ConnectionStateChangedReason [▶ 79] The reason.
newState	Type: TwinCAT.ConnectionState [▶ 72] The new state.
oldState	Type: TwinCAT.ConnectionState [▶ 72] The old state.
session	Type: TwinCAT.ISession [▶ 92] The session.
connection	Type: TwinCAT.IConnection [▶ 79] The connection.

Reference[SessionConnectionStateChangedEventArgs Class \[► 125\]](#)[SessionConnectionStateChangedEventArgs Overload \[► 126\]](#)[TwinCAT Namespace \[► 59\]](#)**6.1.15.1.2 SessionConnectionStateChangedEventArgs Constructor (ConnectionStateChangedReason, ConnectionState, ConnectionState, ISession, IConnection, Exception)**Initializes a new instance of the [SessionConnectionStateChangedEventArgs \[► 125\]](#) class.**Namespace:** [TwinCAT \[► 59\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**







```
public SessionConnectionStateChangedEventArgs (
    ConnectionStateChangedReason reason,
    ConnectionState newState,
    ConnectionState oldState,
    ISession session,
    IConnection connection,
    Exception? e
)
```

Parameters

reason	Type: TwinCAT.ConnectionStateChangedReason [► 79] The reason.
newState	Type: TwinCAT.ConnectionState [► 72] The new state.
oldState	Type: TwinCAT.ConnectionState [► 72] The old state.
session	Type: TwinCAT.ISession [► 92] The session.
connection	Type: TwinCAT.IConnection [► 79] The connection.
e	Type: System.Exception The e.

Reference[SessionConnectionStateChangedEventArgs Class \[► 125\]](#)[SessionConnectionStateChangedEventArgs Overload \[► 126\]](#)[TwinCAT Namespace \[► 59\]](#)**6.1.15.2 SessionConnectionStateChangedEventArgs Properties**The [SessionConnectionStateChangedEventArgs \[► 125\]](#) type exposes the following members.

Properties

	Name	Description
	Connection [▶ 129]	The connection
	Exception [▶ 77]	Exception, (only for Error [▶ 79] (Inherited from ConnectionStringChangedEventArgs [▶ 74].))
	NewState [▶ 77]	New connection state (Inherited from ConnectionStringChangedEventArgs [▶ 74].)
	OldState [▶ 78]	Old connection state (Inherited from ConnectionStringChangedEventArgs [▶ 74].)
	Reason [▶ 78]	Reason for the event (Inherited from ConnectionStringChangedEventArgs [▶ 74].)
	Session [▶ 129]	The session

Reference

[SessionConnectionStringChangedEventArgs Class](#) [[▶ 125](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.15.2.1 SessionConnectionStringChangedEventArgs.Connection Property

The connection

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IConnection Connection { get; }
```

Property Value

Type: [IConnection](#) [[▶ 79](#)]

Reference

[SessionConnectionStringChangedEventArgs Class](#) [[▶ 125](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.15.2.2 SessionConnectionStringChangedEventArgs.Session Property

The session

Namespace: [TwinCAT](#) [[▶ 59](#)]







Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISession Session { get; }
```

Property ValueType: [ISession](#) [[▶ 92](#)]**Reference**[SessionConnectionStateChangedEventArgs Class](#) [[▶ 125](#)][TwinCAT Namespace](#) [[▶ 59](#)]**6.1.15.3 SessionConnectionStateChangedEventArgs Methods**The [SessionConnectionStateChangedEventArgs](#) [[▶ 125](#)] type exposes the following members.**Methods**

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference[SessionConnectionStateChangedEventArgs Class](#) [[▶ 125](#)][TwinCAT Namespace](#) [[▶ 59](#)]**6.1.16 SessionException Class**





Session Exception

Inheritance Hierarchy[System.Object](#)[System.Exception](#)[TwinCAT.AdsException](#) [[▶ 61](#)][TwinCAT.SessionException](#)[TwinCAT.SessionNotConnectedException](#) [[▶ 137](#)]**Namespace:** [TwinCAT](#) [[▶ 59](#)]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**










```
[SerializableAttribute]
public class SessionException : AdsException
```

The [SessionException](#) type exposes the following members.





Constructors





	Name	Description
	SessionException(String) [▶ 134]	Initializes a new instance of the SessionException class.
	SessionException(SerializationInfo, StreamingContext) [▶ 132]	Initializes a new instance of the SessionException class.
	SessionException(String, ISession) [▶ 133]	Initializes a new instance of the SessionException class.
	SessionException(String, ISession, Exception) [▶ 133]	Initializes a new instance of the SessionException class.

Properties


	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Session [▶ 135]	Gets the session.
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)

	Name	Description
	GetObjectData [▶ 136]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Overrides Exception.GetObjectData(SerializationInfo, StreamingContext) .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)





Reference

[TwinCAT Namespace](#) [▶ 59]

[TwinCAT.AdsException](#) [▶ 61]

6.1.16.1 SessionException Constructor

Overload List

	Name	Description
	SessionException(String) [▶ 134]	Initializes a new instance of the SessionException [▶ 130] class.
	SessionException(SerializationInfo, StreamingContext) [▶ 132]	Initializes a new instance of the SessionException [▶ 130] class.
	SessionException(String, ISession) [▶ 133]	Initializes a new instance of the SessionException [▶ 130] class.
	SessionException(String, ISession, Exception) [▶ 133]	Initializes a new instance of the SessionException [▶ 130] class.

Reference

[SessionException Class](#) [▶ 130]

[TwinCAT Namespace](#) [▶ 59]

6.1.16.1.1 SessionException Constructor (SerializationInfo, StreamingContext)

Initializes a new instance of the [SessionException](#) [▶ 130] class.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected SessionException(
    SerializationInfo serializationInfo,
    StreamingContext streamingContext
)
```

Parameters

serializationInfo	Type: System.Runtime.Serialization.SerializationInfo The serialization information.
streamingContext	Type: System.Runtime.Serialization.StreamingContext The streaming context.

Reference

[SessionException Class \[► 130\]](#)

[SessionException Overload \[► 132\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.16.1.2 SessionException Constructor (String, ISession)

Initializes a new instance of the [SessionException \[► 130\]](#) class.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SessionException(
    string message,
    ISession? session
)
```

Parameters

message	Type: System.String The message.
session	Type: TwinCAT.ISession [► 92] The session.

Reference

[SessionException Class \[► 130\]](#)

[SessionException Overload \[► 132\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.16.1.3 SessionException Constructor (String, ISession, Exception)

Initializes a new instance of the [SessionException \[► 130\]](#) class.

Namespace: [TwinCAT \[▸ 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SessionException(
    string message,
    ISession? session,
    Exception? innerException
)
```

Parameters

message	Type: System.String The message.
session	Type: TwinCAT.ISession [▸ 92] The session.
innerException	Type: System.Exception The inner exception.

Reference

[SessionException Class \[▸ 130\]](#)

[SessionException Overload \[▸ 132\]](#)

[TwinCAT Namespace \[▸ 59\]](#)

6.1.16.1.4 SessionException Constructor (String)

Initializes a new instance of the [SessionException \[▸ 130\]](#) class.

Namespace: [TwinCAT \[▸ 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SessionException(
    string message
)
```

Parameters

message	Type: System.String The message.
---------	---

Reference

[SessionException Class \[▸ 130\]](#)










[SessionException Overload \[▸ 132\]](#)

[TwinCAT Namespace \[▸ 59\]](#)

6.1.16.2 SessionException Properties

The [SessionException \[▸ 130\]](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Session [▶ 135]	Gets the session.
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Reference

[SessionException Class](#) [[▶ 130](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.16.2.1 SessionException.Session Property

Gets the session.

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISession? Session { get; }
```

Property Value

Type: [ISession](#) [[▶ 92](#)]

The session.

Reference









[SessionException Class](#) [[▶ 130](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.16.3 SessionException Methods

The [SessionException](#) [[▶ 130](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData [▶ 136]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Overrides Exception.GetObjectData(SerializationInfo, StreamingContext) .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Reference

[SessionException Class](#) [▶ 130]

[TwinCAT Namespace](#) [▶ 59]

6.1.16.3.1 SessionException.GetObjectData Method

When overridden in a derived class, sets the [SerializationInfo](#) with information about the exception.

Namespace: [TwinCAT](#) [▶ 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override void GetObjectData(
    SerializationInfo info,
    StreamingContext context
)
```

Parameters

info	Type: System.Runtime.Serialization.SerializationInfo The SerializationInfo that holds the serialized object data about the exception being thrown.
context	Type: System.Runtime.Serialization.StreamingContext The StreamingContext that contains contextual information about the source or destination.

Implements

[ISerializable.GetObjectData\(SerializationInfo, StreamingContext\)](#)

Exceptions

Exception	Condition
ArgumentNullException	info

Reference


[SessionException Class](#) [▶ 130]

[TwinCAT Namespace](#) [▶ 59]

6.1.16.4 SessionException Events

The [SessionException](#) [▶ 130] type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

Reference

[SessionException Class](#) [▶ 130]

[TwinCAT Namespace](#) [▶ 59]

6.1.17 SessionNotConnectedException Class

Class [SessionNotConnectedException](#).

Inheritance Hierarchy

[System.Object](#)

[System.Exception](#)

[TwinCAT.AdsException](#) [▶ 61]

[TwinCAT.SessionException](#) [▶ 130]

[TwinCAT.SessionNotConnectedException](#)

Namespace: [TwinCAT](#) [▶ 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229


Syntax



C#

```
[SerializableAttribute]
public class SessionNotConnectedException : SessionException
```










The [SessionNotConnectedException](#) type exposes the following members.

Constructors









	Name	Description
	SessionNotConnectedException(ISession) [▶ 139]	Initializes a new instance of the SessionNotConnectedException class.

	Name	Description
	SessionNotConnectedException(SerializationInfo, StreamingContext) [▶ 140]	Initializes a new instance of the SessionNotConnectedException class.
	SessionNotConnectedException(String, ISession) [▶ 140]	Initializes a new instance of the SessionNotConnectedException class.


Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Session [▶ 135]	Gets the session. (Inherited from SessionException [▶ 130].)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData [▶ 136]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from SessionException [▶ 130].)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)




Reference

[TwinCAT Namespace](#) [▶ 59]

[TwinCAT.SessionException](#) [▶ 130]

6.1.17.1 SessionNotConnectedException Constructor

Overload List

	Name	Description
	SessionNotConnectedException(ISession) [▶ 139]	Initializes a new instance of the SessionNotConnectedException [▶ 137] class.
	SessionNotConnectedException(SerializationInfo, StreamingContext) [▶ 140]	Initializes a new instance of the SessionNotConnectedException [▶ 137] class.
	SessionNotConnectedException(String, ISession) [▶ 140]	Initializes a new instance of the SessionNotConnectedException [▶ 137] class.

Reference

[SessionNotConnectedException Class](#) [▶ 137]

[TwinCAT Namespace](#) [▶ 59]

6.1.17.1.1 SessionNotConnectedException Constructor (ISession)

Initializes a new instance of the [SessionNotConnectedException](#) [▶ 137] class.

Namespace: [TwinCAT](#) [▶ 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SessionNotConnectedException(
    ISession session
)
```

Parameters

session	Type: TwinCAT.ISession [▸ 92] The session.
---------	---

Reference

[SessionNotConnectedException Class \[▸ 137\]](#)

[SessionNotConnectedException Overload \[▸ 139\]](#)

[TwinCAT Namespace \[▸ 59\]](#)

6.1.17.1.2 **SessionNotConnectedException Constructor (SerializationInfo, StreamingContext)**

Initializes a new instance of the [SessionNotConnectedException \[▸ 137\]](#) class.

Namespace: [TwinCAT \[▸ 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
protected SessionNotConnectedException(
    SerializationInfo serializationInfo,
    StreamingContext streamingContext
)
```

Parameters

serializationInfo	Type: System.Runtime.Serialization.SerializationInfo The serialization information.
streamingContext	Type: System.Runtime.Serialization.StreamingContext The streaming context.

Reference

[SessionNotConnectedException Class \[▸ 137\]](#)

[SessionNotConnectedException Overload \[▸ 139\]](#)

[TwinCAT Namespace \[▸ 59\]](#)

6.1.17.1.3 **SessionNotConnectedException Constructor (String, ISession)**

Initializes a new instance of the [SessionNotConnectedException \[▸ 137\]](#) class.

Namespace: [TwinCAT \[▸ 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public SessionNotConnectedException(
    string message,
    ISession session
)
```

Parameters

message	Type: System.String The message.
session	Type: TwinCAT.ISession [▶ 92] The session.

Reference

[SessionNotConnectedException Class](#) [▶ 137]










[SessionNotConnectedException Overload](#) [▶ 139]

[TwinCAT Namespace](#) [▶ 59]

6.1.17.2 SessionNotConnectedException Properties

The [SessionNotConnectedException](#) [▶ 137] type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Session [▶ 135]	Gets the session. (Inherited from SessionException [▶ 130].)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Reference


[SessionNotConnectedException Class](#) [▶ 137]








[TwinCAT Namespace](#) [▶ 59]

6.1.17.3 SessionNotConnectedException Methods

The [SessionNotConnectedException](#) [▶ 137] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)

	Name	Description
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData ▶ 136	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from SessionException ▶ 130 .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Reference


[SessionNotConnectedException Class](#) ▶ [137](#)

[TwinCAT Namespace](#) ▶ [59](#)

6.1.17.4 SessionNotConnectedException Events

The [SessionNotConnectedException](#) ▶ [137](#) type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

Reference

[SessionNotConnectedException Class](#) ▶ [137](#)

[TwinCAT Namespace](#) ▶ [59](#)

6.1.18 SessionProvider.TSession, TAddress, TSettings. Class

Abstract base class for a Custom Session provider

Inheritance Hierarchy

[System.Object](#)

TwinCAT.SessionProvider.TSession, TAddress, TSettings.

Namespace: [TwinCAT](#) ▶ [59](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#



```
public abstract class SessionProvider<TSession, TAddress, TSettings> : ISessionProvider<TSession, TAddress, TSettings>,
    ISessionProvider
where TSession : ISession
where TSettings : class
```

Type Parameters



TSession	SessionType
TAddress	Address type
TSettings	Communication settings type

The SessionProvider.TSession, TAddress, TSettings. type exposes the following members.







Constructors

	Name	Description
	SessionProvider.TSession, TAddress, TSettings.. [► 144]	Initializes a new instance of the SessionProvider.TSession, TAddress, TSettings. class.
	SessionProvider.TSession, TAddress, TSettings. (SessionProviderCapabilities) [► 145]	Initializes a new instance of the SessionProvider.TSession, TAddress, TSettings. class.



Properties

	Name	Description
	Capabilities [► 145]	Gets the capabilities.
	Name [► 146]	Gets the name of the SessionProvider

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Fields



	Name	Description
 	s_self [▶ 147]	Singleton Instance.

Reference

[TwinCAT Namespace](#) [[▶ 59](#)]

[TwinCAT.ISessionProvider.TSession, TAddress, TSettings.](#) [[▶ 102](#)]

6.1.18.1 SessionProvider.TSession, TAddress, TSettings. Constructor**Overload List**

	Name	Description
	SessionProvider.TSession, TAddress, TSettings.. [▶ 144]	Initializes a new instance of the SessionProvider.TSession, TAddress, TSettings. [▶ 142] class.
	SessionProvider.TSession, TAddress, TSettings. (SessionProviderCapabilities) [▶ 145]	Initializes a new instance of the SessionProvider.TSession, TAddress, TSettings. [▶ 142] class.

Reference

[SessionProvider.TSession, TAddress, TSettings. Class](#) [[▶ 142](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.18.1.1 SessionProvider.TSession, TAddress, TSettings. Constructor

Initializes a new instance of the [SessionProvider.TSession, TAddress, TSettings.](#) [[▶ 142](#)] class.

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
protected SessionProvider()
```

Exceptions

Exception	Condition
Exception	Session provider already instantiated!

Reference

[SessionProvider.TSession, TAddress, TSettings. Class](#) [[▶ 142](#)]

[SessionProvider.TSession, TAddress, TSettings. Overload](#) [[▶ 144](#)]

[TwinCAT Namespace \[▶ 59\]](#)

6.1.18.1.2 **SessionProvider.TSession, TAddress, TSettings. Constructor (SessionProviderCapabilities)**

Initializes a new instance of the [SessionProvider.TSession, TAddress, TSettings. \[▶ 142\]](#) class.

Namespace: [TwinCAT \[▶ 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected SessionProvider(
    SessionProviderCapabilities cap
)
```

Parameters

cap	Type: TwinCAT.SessionProviderCapabilities [▶ 148]
-----	---

Exceptions

Exception	Condition
Exception	Session provider already instantiated!

Reference

[SessionProvider.TSession, TAddress, TSettings. Class \[▶ 142\]](#)



[SessionProvider.TSession, TAddress, TSettings. Overload \[▶ 144\]](#)

[TwinCAT Namespace \[▶ 59\]](#)

6.1.18.2 **SessionProvider.TSession, TAddress, TSettings. Properties**

The [SessionProvider.TSession, TAddress, TSettings. \[▶ 142\]](#) generic type exposes the following members.

Properties

	Name	Description
	Capabilities [▶ 145]	Gets the capabilities.
	Name [▶ 146]	Gets the name of the SessionProvider

Reference

[SessionProvider.TSession, TAddress, TSettings. Class \[▶ 142\]](#)

[TwinCAT Namespace \[▶ 59\]](#)

6.1.18.2.1 **SessionProvider.TSession, TAddress, TSettings..Capabilities Property**

Gets the capabilities.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SessionProviderCapabilities Capabilities { get; }
```

Property Value

Type: [SessionProviderCapabilities \[► 148\]](#)

The capabilities.

Implements

[ISessionProvider.Capabilities \[► 100\]](#)

Reference

[SessionProvider.TSession, TAddress, TSettings. Class \[► 142\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.18.2 SessionProvider.TSession, TAddress, TSettings..Name Property

Gets the name of the SessionProvider

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public abstract string Name { get; }
```

Property Value

Type: [String](#)

The name.

Implements

[ISessionProvider.Name \[► 101\]](#)

Reference







[SessionProvider.TSession, TAddress, TSettings. Class \[► 142\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.18.3 SessionProvider.TSession, TAddress, TSettings. Methods

The [SessionProvider.TSession, TAddress, TSettings. \[► 142\]](#) generic type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference



[SessionProvider.TSession, TAddress, TSettings. Class \[► 142\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.18.4 SessionProvider.TSession, TAddress, TSettings. Fields

The [SessionProvider.TSession, TAddress, TSettings. \[► 142\]](#) generic type exposes the following members.

Fields

	Name	Description
 	s_self [► 147]	Singleton Instance.

Reference

[SessionProvider.TSession, TAddress, TSettings. Class \[► 142\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.18.4.1 SessionProvider.TSession, TAddress, TSettings..s_self Field

Singleton Instance.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected static ISessionProvider<TSession, TAddress, TSettings>? s_self
```

Field Value

Type: [ISessionProvider \[► 102\].TSession \[► 142\]](#), [TAddress \[► 142\]](#), [TSettings \[► 142\]](#).

Reference

[SessionProvider.TSession, TAddress, TSettings. Class \[► 142\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.19 SessionProviderCapabilities Enumeration

Enum SessionProviderCapabilities

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
[FlagsAttribute]
public enum SessionProviderCapabilities
```

Members

	Member name	Value	Description
	DataTypeSupport	1	Supports DataTypes
	SymbolBrowsing	2	Supports Symbol Browsing
	ValueRead	4	Supports Value Read
	ValueWrite	8	Support Value Write
	ValueNotifications	16	Supports Value changed Notifications
	None	0	Uninitialized / None
	Mask_All	31	All Capabilities active

Reference

[TwinCAT Namespace \[► 59\]](#)

6.1.20 SymbolLoaderSettings Class

Settings object for the SymbolLoader initialization.

Inheritance Hierarchy

[System.Object](#)

 TwinCAT.SymbolLoaderSettings

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229


Syntax



C#

```
public class SymbolLoaderSettings : ISymbolLoaderSettings,
    IValueFactorySettings, IAdsClientSettings
```










The SymbolLoaderSettings type exposes the following members.

Constructors




	Name	Description
	SymbolLoaderSettings(SymbolsLoadMode) [► 151]	Initializes a new instance of the SymbolLoaderSettings class with IndexGroupOffsetPreferred [► 3228] .




	Name	Description
	SymbolLoaderSettings(SymbolsLoadMode, ValueAccessMode) [▶ 151]	Initializes a new instance of the SymbolLoaderSettings class.
	SymbolLoaderSettings(SymbolsLoadMode, ValueCreationMode, ValueAccessMode) [▶ 152]	Initializes a new instance of the SymbolLoaderSettings class.

Properties

	Name	Description
	AutomaticReconnection [▶ 153]	Gets or sets a value indicating whether Disconnect connections can be reconnected.
	Default [▶ 154]	Gets the default settings object for standard symbols.
	DefaultDynamic [▶ 154]	Gets the default settings object for Dynamic symbols.
	ExpandArrayValues [▶ 158]	Gets or sets a value indicating whether the Value Factory automatically expands ArrayValues to its Elements (Default: true)
	NonCachedArrayElements [▶ 155]	Gets or sets the setting to create ArrayElements "On-The-Fly" (Default True)
	SymbolsLoadMode [▶ 155]	Gets or sets the symbols load mode.
	ValueAccessMode [▶ 156]	Gets or sets the value access mode.
	ValueCreation [▶ 156]	Gets or sets the value creation mode.
	ValueUpdateMode [▶ 157]	Gets or sets the value update mode.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)

	Name	Description
	GetType	Gets the <u>Type</u> of the current instance. (Inherited from <u>Object</u> .)
	MemberwiseClone	Creates a shallow copy of the current <u>Object</u> . (Inherited from <u>Object</u> .)
	ToString	Returns a string that represents the current object. (Inherited from <u>Object</u> .)

Remarks

This settings object is used for the initialization of the internal Symbol loader object.

Reference

[TwinCAT Namespace](#) [► 59]

[TwinCAT.ISymbolLoaderSettings](#) [► 104]

[TwinCAT.Ads.TypeSystem.SymbolLoaderFactory](#) [► 1923]

[TwinCAT.TypeSystem.ISymbolLoader](#) [► 2714]

[TwinCAT.SymbolsLoadMode](#) [► 159]

[TwinCAT.ValueAccess.ValueAccessMode](#) [► 3228]




Also see about this

 [ValueAccessMode Enumeration](#) [► 2002]

 [ValueAccessMode Enumeration](#) [► 2002]

6.1.20.1 SymbolLoaderSettings Constructor

Overload List

	Name	Description
	SymbolLoaderSettings(SymbolsLoadMode) [► 151]	Initializes a new instance of the SymbolLoaderSettings [► 148] class with IndexGroupOffsetPreferred [► 3228].
	SymbolLoaderSettings(SymbolsLoadMode, ValueAccessMode) [► 151]	Initializes a new instance of the SymbolLoaderSettings [► 148] class.
	SymbolLoaderSettings(SymbolsLoadMode, ValueCreationModes, ValueAccessMode) [► 152]	Initializes a new instance of the SymbolLoaderSettings [► 148] class.

Reference

[SymbolLoaderSettings Class](#) [► 148]

[TwinCAT Namespace](#) [► 59]

Also see about this

 [ValueAccessMode Enumeration](#) [► 2002]

6.1.20.1.1 SymbolLoaderSettings Constructor (SymbolsLoadMode)

Initializes a new instance of the [SymbolLoaderSettings](#) [► 148] class with [IndexGroupOffsetPreferred](#) [► 3228].

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolLoaderSettings(
    SymbolsLoadMode loadMode
)
```

Parameters

loadMode	Type: TwinCAT.SymbolsLoadMode [► 159] The load mode.
----------	---

Reference

[SymbolLoaderSettings Class](#) [► 148]

[SymbolLoaderSettings Overload](#) [► 150]

[TwinCAT Namespace](#) [► 59]

Also see about this

 [ValueAccessMode Enumeration](#) [► 2002]

6.1.20.1.2 SymbolLoaderSettings Constructor (SymbolsLoadMode, ValueAccessMode)

Initializes a new instance of the [SymbolLoaderSettings](#) [► 148] class.

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolLoaderSettings(
    SymbolsLoadMode loadMode,
    ValueAccessMode valueAccess
)
```

Parameters

loadMode	Type: TwinCAT.SymbolsLoadMode [► 159] The load mode.
valueAccess	Type: TwinCAT.ValueAccess.ValueAccessMode [► 3228] The value access.


Reference[SymbolLoaderSettings Class \[▶ 148\]](#)[SymbolLoaderSettings Overload \[▶ 150\]](#)[TwinCAT Namespace \[▶ 59\]](#)**Also see about this** [ValueAccessMode Enumeration \[▶ 2002\]](#)**6.1.20.1.3 SymbolLoaderSettings Constructor (SymbolsLoadMode, ValueCreationModes, ValueAccessMode)**Initializes a new instance of the [SymbolLoaderSettings \[▶ 148\]](#) class.**Namespace:** [TwinCAT \[▶ 59\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**









```
public SymbolLoaderSettings(
    SymbolsLoadMode loadMode,
    ValueCreationModes valueCreation,
    ValueAccessMode valueAccess
)
```

Parameters

loadMode	Type: TwinCAT.SymbolsLoadMode [▶ 159] The load mode.
valueCreation	Type: TwinCAT.ValueAccess.ValueCreationModes [▶ 3223] The dynamic value creation.
valueAccess	Type: TwinCAT.ValueAccess.ValueAccessMode [▶ 3228] The value access.

Reference[SymbolLoaderSettings Class \[▶ 148\]](#)[SymbolLoaderSettings Overload \[▶ 150\]](#)[TwinCAT Namespace \[▶ 59\]](#)**Also see about this** [ValueAccessMode Enumeration \[▶ 2002\]](#)**6.1.20.2 SymbolLoaderSettings Properties**The [SymbolLoaderSettings \[▶ 148\]](#) type exposes the following members.**Properties**

	Name	Description
	AutomaticReconnection [▶ 153]	Gets or sets a value indicating whether Disconnect connections can be reconnected.

	Name	Description
 S	Default [▶ 154]	Gets the default settings object for standard symbols.
 S	DefaultDynamic [▶ 154]	Gets the default settings object for Dynamic symbols.
	ExpandArrayValues [▶ 158]	Gets or sets a value indicating whether the Value Factory automatically expands ArrayValues to its Elements (Default: true)
	NonCachedArrayElements [▶ 155]	Gets or sets the setting to create ArrayElements "On-The-Fly" (Default True)
	SymbolsLoadMode [▶ 155]	Gets or sets the symbols load mode.
	ValueAccessMode [▶ 156]	Gets or sets the value access mode.
	ValueCreation [▶ 156]	Gets or sets the value creation mode.
	ValueUpdateMode [▶ 157]	Gets or sets the value update mode.

Reference

[SymbolLoaderSettings Class](#) [[▶ 148](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.20.2.1 SymbolLoaderSettings.AutomaticReconnection Property

Gets or sets a value indicating whether Disconnect connections can be reconnected.

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool AutomaticReconnection { get; set; }
```

Property Value

Type: [Boolean](#)

true if Disconnect connections can be reconnected; otherwise, false.

Implements

[IAdsClientSettings.AutomaticReconnection](#) [[▶ 176](#)]

Reference

[SymbolLoaderSettings Class](#) [[▶ 148](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.20.2 SymbolLoaderSettings.Default Property

Gets the default settings object for standard symbols.

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static SymbolLoaderSettings Default { get; }
```

Property Value

Type: [SymbolLoaderSettings](#) [[▶ 148](#)]

The default settings object.

Remarks

The following defaults are set here:

Setting	Description
Symbols load mode (SymbolsLoadMode [▶ 155])	Create virtual tree (VirtualTree [▶ 159]).
Value access mode (ValueAccessMode [▶ 156])	Prefer Symbolic access of values (SymbolicByHandle [▶ 3228]).
Value creation mode (ValueCreation [▶ 156])	Create .NET integral primitives if possible (Default [▶ 3223]).

Reference

[SymbolLoaderSettings Class](#) [[▶ 148](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

[SymbolLoaderSettings.DefaultDynamic](#) [[▶ 154](#)]

Also see about this

 [ValueAccessMode Enumeration](#) [[▶ 2002](#)]

6.1.20.3 SymbolLoaderSettings.DefaultDynamic Property

Gets the default settings object for Dynamic symbols.

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static SymbolLoaderSettings DefaultDynamic { get; }
```

Property Value

Type: [SymbolLoaderSettings](#) [[▶ 148](#)]

The dynamic default settings object.

Remarks

The following defaults are set here:

Setting	Description
Symbols load mode (SymbolsLoadMode [▶ 155])	Create dynamic tree (DynamicTree [▶ 159]).
Value access mode (ValueAccessMode [▶ 156])	Prefer Symbolic access of values (SymbolicByHandle [▶ 3228]).
Value creation mode (ValueCreation [▶ 156])	Create .NET integral primitives if possible (Default [▶ 3223]).

Reference

[SymbolLoaderSettings Class](#) [▶ 148]

[TwinCAT Namespace](#) [▶ 59]

[SymbolLoaderSettings.Default](#) [▶ 154]

Also see about this

 [ValueAccessMode Enumeration](#) [▶ 2002]

6.1.20.2.4 SymbolLoaderSettings.NonCachedArrayElements Property

Gets or sets the setting to create ArrayElements "On-The-Fly" (Default True)

Namespace: [TwinCAT](#) [▶ 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public bool NonCachedArrayElements { get; set; }
```

Property Value

Type: [Boolean](#)

The value access mode.

Implements

[IValueFactorySettings.NonCachedArrayElements](#) [▶ 177]

Reference

[SymbolLoaderSettings Class](#) [▶ 148]

[TwinCAT Namespace](#) [▶ 59]

6.1.20.2.5 SymbolLoaderSettings.SymbolsLoadMode Property

Gets or sets the symbols load mode.

Namespace: [TwinCAT](#) [▶ 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolsLoadMode SymbolsLoadMode { get; set; }
```

Property Value

Type: [SymbolsLoadMode](#) [► 159]

The symbols load mode.

Implements

[ISymbolLoaderSettings.SymbolsLoadMode](#) [► 105]

Reference

[SymbolLoaderSettings Class](#) [► 148]

[TwinCAT Namespace](#) [► 59]

6.1.20.2.6 SymbolLoaderSettings.ValueAccessMode Property

Gets or sets the value access mode.

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ValueAccessMode ValueAccessMode { get; set; }
```

Property Value

Type: [ValueAccessMode](#) [► 3228]

The value access mode.

Implements

[IValueFactorySettings.ValueAccessMode](#) [► 178]

Reference

[SymbolLoaderSettings Class](#) [► 148]

[TwinCAT Namespace](#) [► 59]

Also see about this

 [ValueAccessMode Enumeration](#) [► 2002]

6.1.20.2.7 SymbolLoaderSettings.ValueCreation Property

Gets or sets the value creation mode.

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ValueCreationModes ValueCreation { get; set; }
```

Property Value

Type: [ValueCreationModes](#) [[▶ 3223](#)]

The dynamic value mode.

Implements

[IValueFactorySettings.ValueCreation](#) [[▶ 178](#)]

Reference

[SymbolLoaderSettings Class](#) [[▶ 148](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.20.2.8 SymbolLoaderSettings.ValueUpdateMode Property

Gets or sets the value update mode.

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ValueUpdateMode ValueUpdateMode { get; set; }
```

Property Value

Type: [ValueUpdateMode](#) [[▶ 3229](#)]

The value update mode.

Implements

[IValueFactorySettings.ValueUpdateMode](#) [[▶ 179](#)]

Remarks



Mode	Description
Immediately [▶ 3229]	Writes the values of this DynamicValue [▶ 2347 "]/> instantly when setting its value or the value of its child members/elements.
Triggered [▶ 3229]	Caches internally the value of this DynamicValue [▶ 2347 "]/> until the Write [▶ 2372] method is called. This reduces ADS roundtrips, if one or more member/element values should be changed. Furthermore the write on the destination system happens consistently in one ADS Write operation, which could be important for dependent properties/members/elements.

Reference

[SymbolLoaderSettings Class](#) [[▶ 148](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

Also see about this

-  [ValueUpdateMode Enumeration](#) [▶ 159]
-  [ValueUpdateMode Enumeration](#) [▶ 159]

6.1.20.2.9 SymbolLoaderSettings.ExpandArrayValues Property

Gets or sets a value indicating whether the Value Factory automatically expands ArrayValues to its Elements (Default: true)

Namespace: [TwinCAT](#) [▶ 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public bool ExpandArrayValues { get; set; }
```

Property Value

Type: [Boolean](#)

true if [create array elements]; otherwise, false.

Implements

[IValueFactorySettings.ExpandArrayValues](#) [▶ 177]

Remarks

Actually, this takes place when DynamicSymbols are created. Array Elements are created on-access to the Array, so that they can be accessed by a JSON Serializer without further expanding. If this property is set to false, then an array element is only created on access!

Reference







[SymbolLoaderSettings Class](#) [▶ 148]

[TwinCAT Namespace](#) [▶ 59]

6.1.20.3 SymbolLoaderSettings Methods

The [SymbolLoaderSettings](#) [▶ 148] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[SymbolLoaderSettings Class \[▶ 148\]](#)

[TwinCAT Namespace \[▶ 59\]](#)

6.1.21 SymbolsLoadMode Enumeration

Enum SymbolsLoadMode

Namespace: [TwinCAT \[▶ 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum SymbolsLoadMode
```

Members

	Member name	Value	Description
	Flat	0	Loads the Symbols organized as Flat List
	VirtualTree	1	Loads the Symbols organized as Virtual tree (Symbol Parent - Child relationships)
	DynamicTree	2	Loads the Symbols as a Virtual tree with Dynamic Symbols (Only available within versions > 4.X of this ADS Api)

Reference

[TwinCAT Namespace \[▶ 59\]](#)

6.1.22 ValueUpdateMode Enumeration

Value Update Mode.

Namespace: [TwinCAT \[▶ 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public enum ValueUpdateMode
```

Members

	Member name	Value	Description
	None	0	No automatic Value Update / Uninitialized
	Immediately	1	Update Value immediately on property set access.
	Triggered	2	Triggers the ValueUpdate explicitly

Remarks

Mode	Description
Immediately	Writes the values of this DynamicValue instantly when setting its value or the value of its child members/elements.
Triggered	Caches internally the value of this DynamicValue until the DynamicValue.Write method is called. This reduces ADS roundtrips, if one or more member/element values should be changed. Furthermore the write on the destination system happens consistently in one ADS Write operation, which could be important for dependent properties/members/elements.

Reference

[TwinCAT Namespace \[► 59\]](#)

6.1.23 AdsCommunicationStatistics Class

ADS Communication statistics

Inheritance Hierarchy

[System.Object](#)

TwinCAT.AdsCommunicationStatistics

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public class AdsCommunicationStatistics
```

The AdsCommunicationStatistics type exposes the following members.

Properties

Name	Description
AccessWaitTime [► 162]	Gets the wait time for the next access (Resurrection time) if in Lost [► 72] .
CommunicationErrorsSinceLastSucceeded [► 163]	Gets the number of communication errors since the last successful access
ConnectionActiveSince [► 163]	Gets the DateTime of the last connection activation.
ConnectionEstablishedAt [► 164]	Gets the UTC time when the current connection was established.
ConnectionLostAt [► 164]	Gets the connection lost time.
LastAccessAt [► 164]	Gets the Time of the last read/write access
LastCommunicationErrorAt [► 165]	Gets the last tripping error time.
LastErrorAt [► 165]	Get the Time of the last occurred Error (Error response codes and tripping errors)

	Name	Description
	LastErrorCode [▶ 166]	Gets error code of the last access.
	LastSucceededAt [▶ 166]	Gets the Date/Time of the last succeeded ADS communication/Roundtrip.
	SessionEstablishedAt [▶ 167]	Gets the UTC time when the session was established.
	TotalCommunicationErrors [▶ 167]	Gets the communication error count.
	TotalConnectionLosses [▶ 167]	Gets the connection lost count.
	TotalCycles [▶ 168]	Gets the total cycles/requests done so far by this session.
	TotalErrors [▶ 168]	Gets the total number of negative ADS responses.
	TotalResurrections [▶ 169]	Gets the number of resurrections on the AdsConnection [▶ 412]

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Remarks

These statistics can be used for communication diagnosis. They contain Error/Succeed counts as well as Resurrection infos.

Reference

[TwinCAT Namespace](#) [▶ 59]

6.1.23.1 AdsCommunicationStatistics Properties

The [AdsCommunicationStatistics](#) [▶ 160] type exposes the following members.

Properties

	Name	Description
	AccessWaitTime [▶ 162]	Gets the wait time for the next access (Resurrection time) if in Lost [▶ 72].

	Name	Description
	CommunicationErrorsSinceLastSucceeded [▶ 163]	Gets the number of communication errors since the last successful access
	ConnectionActiveSince [▶ 163]	Gets the DateTime of the last connection activation.
	ConnectionEstablishedAt [▶ 164]	Gets the UTC time when the current connection was established.
	ConnectionLostAt [▶ 164]	Gets the connection lost time.
	LastAccessAt [▶ 164]	Gets the Time of the last read/write access
	LastCommunicationErrorAt [▶ 165]	Gets the last tripping error time.
	LastErrorAt [▶ 165]	Get the Time of the last occurred Error (Error response codes and tripping errors)
	LastErrorCode [▶ 166]	Gets error code of the last access.
	LastSucceededAt [▶ 166]	Gets the Date/Time of the last succeeded ADS communication/Roundtrip.
	SessionEstablishedAt [▶ 167]	Gets the UTC time when the session was established.
	TotalCommunicationErrors [▶ 167]	Gets the communication error count.
	TotalConnectionLosses [▶ 167]	Gets the connection lost count.
	TotalCycles [▶ 168]	Gets the total cycles/requests done so far by this session.
	TotalErrors [▶ 168]	Gets the total number of negative ADS responses.
	TotalResurrections [▶ 169]	Gets the number of resurrections on the AdsConnection [▶ 412]

Reference

[AdsCommunicationStatistics Class](#) [[▶ 160](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.23.1.1 **AdsCommunicationStatistics.AccessWaitTime** Property

Gets the wait time for the next access (Resurrection time) if in [Lost](#) [[▶ 72](#)].

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public TimeSpan AccessWaitTime { get; }
```

Property Value

Type: [TimeSpan](#)

The wait time if in [Lost](#) [[▶ 72](#)] otherwise **TimeSpan.Zero**.

Reference

[AdsCommunicationStatistics Class](#) [[▶ 160](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.23.1.2 **AdsCommunicationStatistics.CommunicationErrorsSinceLastSucceeded Property**

Gets the number of communication errors since the last successful access

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int CommunicationErrorsSinceLastSucceeded { get; }
```

Property Value

Type: [Int32](#)

Remarks

Only communication (tripping, TrippingErrors) errors count here. A succeeded roundtrip (non tripping) sets this value to zero.

Reference

[AdsCommunicationStatistics Class](#) [[▶ 160](#)]

[TwinCAT Namespace](#) [[▶ 59](#)]

6.1.23.1.3 **AdsCommunicationStatistics.ConnectionActiveSince Property**

Gets the DateTime of the last connection activation.

Namespace: [TwinCAT](#) [[▶ 59](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTimeOffset? ConnectionActiveSince { get; }
```

Property Value

Type: [Nullable.DateTimeOffset](#).
Connection active time.

Reference

[AdsCommunicationStatistics Class](#) [[▶](#) 160]

[TwinCAT Namespace](#) [[▶](#) 59]

6.1.23.1.4 **AdsCommunicationStatistics.ConnectionEstablishedAt** Property

Gets the UTC time when the current connection was established.

Namespace: [TwinCAT](#) [[▶](#) 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTimeOffset? ConnectionEstablishedAt { get; }
```

Property Value

Type: [Nullable.DateTimeOffset](#).
The connection established at.

Reference

[AdsCommunicationStatistics Class](#) [[▶](#) 160]

[TwinCAT Namespace](#) [[▶](#) 59]

6.1.23.1.5 **AdsCommunicationStatistics.ConnectionLostAt** Property

Gets the connection lost time.

Namespace: [TwinCAT](#) [[▶](#) 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTimeOffset? ConnectionLostAt { get; }
```

Property Value

Type: [Nullable.DateTimeOffset](#).
The connection lost time.

Reference

[AdsCommunicationStatistics Class](#) [[▶](#) 160]

[TwinCAT Namespace](#) [[▶](#) 59]

6.1.23.1.6 **AdsCommunicationStatistics.LastAccessAt** Property

Gets the Time of the last read/write access

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTimeOffset? LastAccessAt { get; }
```

Property Value

Type: [Nullable.DateTimeOffset](#).

DateTime of the last access.

Reference

[AdsCommunicationStatistics Class \[► 160\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.23.1.7 **AdsCommunicationStatistics.LastCommunicationErrorAt Property**

Gets the last tripping error time.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTimeOffset LastCommunicationErrorAt { get; }
```

Property Value

Type: [DateTimeOffset](#)

The last tripping error time.

Remarks

Tripping errors are errors that are preventing the ADS Session to communicate until a resurrection occurs. These are all errors that are classified that the target system could not be reached.

Reference

[AdsCommunicationStatistics Class \[► 160\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.23.1.8 **AdsCommunicationStatistics.LastErrorAt Property**

Get the Time of the last occurred Error (Error response codes and tripping errors)

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTimeOffset LastErrorAt { get; }
```

Property Value

Type: [DateTimeOffset](#)

The error count since last access.

Reference

[AdsCommunicationStatistics Class](#) [[▶](#) 160]

[TwinCAT Namespace](#) [[▶](#) 59]

6.1.23.1.9 **AdsCommunicationStatistics.LastErrorCode** Property

Gets error code of the last access.

Namespace: [TwinCAT](#) [[▶](#) 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode LastErrorCode { get; }
```

Property Value

Type: [AdsErrorCode](#) [[▶](#) 664]

The last succeeded access.

Reference

[AdsCommunicationStatistics Class](#) [[▶](#) 160]

[TwinCAT Namespace](#) [[▶](#) 59]

6.1.23.1.10 **AdsCommunicationStatistics.LastSucceededAt** Property

Gets the Date/Time of the last succeeded ADS communication/Roundtrip.

Namespace: [TwinCAT](#) [[▶](#) 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTimeOffset? LastSucceededAt { get; }
```

Property Value

Type: [Nullable.DateTimeOffset](#).

The Date/Time value.

Remarks

A successful communication is also a negative ADS response (not [NoError](#) [[▶](#) 664]) that is not classified as communication/tripping error (TrippingErrors).

Reference

[AdsCommunicationStatistics Class](#) [[▶](#) 160]

[TwinCAT Namespace \[► 59\]](#)

6.1.23.1.11 **AdsCommunicationStatistics.SessionEstablishedAt** Property

Gets the UTC time when the session was established.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTimeOffset SessionEstablishedAt { get; }
```

Property Value

Type: [DateTimeOffset](#)

The session established at.

Reference

[AdsCommunicationStatistics Class \[► 160\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.23.1.12 **AdsCommunicationStatistics.TotalCommunicationErrors** Property

Gets the communication error count.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int TotalCommunicationErrors { get; }
```

Property Value

Type: [Int32](#)

The communication error count.

Remarks

The communication errors are the errors that are classified as communication tripping errors (Network communication problems e.g. device not reachable, TrippingErrors)

Reference

[AdsCommunicationStatistics Class \[► 160\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.23.1.13 **AdsCommunicationStatistics.TotalConnectionLosses** Property

Gets the connection lost count.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int TotalConnectionLosses { get; }
```

Property Value

Type: [Int32](#)

The connection lost count.

Reference

[AdsCommunicationStatistics Class](#) [[▶](#) [160](#)]

[TwinCAT Namespace](#) [[▶](#) [59](#)]

6.1.23.1.14 **AdsCommunicationStatistics.TotalCycles** Property

Gets the total cycles/requests done so far by this session.

Namespace: [TwinCAT](#) [[▶](#) [59](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int TotalCycles { get; }
```

Property Value

Type: [Int32](#)

The total cycles / requests

Reference

[AdsCommunicationStatistics Class](#) [[▶](#) [160](#)]

[TwinCAT Namespace](#) [[▶](#) [59](#)]

6.1.23.1.15 **AdsCommunicationStatistics.TotalErrors** Property

Gets the total number of negative ADS responses.

Namespace: [TwinCAT](#) [[▶](#) [59](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int TotalErrors { get; }
```

Property Value

Type: [Int32](#)

The total number of negative ADS responses.

Remarks

This number includes all communication/tripping errors and succeeded negative ADS responses.

Reference

[AdsCommunicationStatistics Class \[► 160\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.23.1.16 AdsCommunicationStatistics.TotalResurrections Property

Gets the number of resurrections on the [AdsConnection \[► 412\]](#)

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public int TotalResurrections { get; }
```

Property Value

Type: [Int32](#)

The resurrections.

Reference

[AdsCommunicationStatistics Class \[► 160\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.23.2 AdsCommunicationStatistics Methods

The [AdsCommunicationStatistics \[► 160\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[AdsCommunicationStatistics Class \[► 160\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.24 RouteAccessType Enumeration

Specifies the method to determine the Remote routes list.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public enum RouteAccessType
```

Members

	Member name	Value	Description
	None	0	None / Uninitialized
	Actual	1	Gets the actual Remote Routes used by the TwinCAT Router. This can contain all 'virtual' Routes that are superimposed by a valid MQTT Publisher configuration. This list doesn't contain configuration data of the single routes.
	Registry	2	The static routes are the Route configurations stored in the Registry (TC2 and Windows Platform only), Obsolete
	Configured	3	Gets the configured Routes from the TwinCAT System Server. These Routes contain all the standard Routes, the configured AdsSecure Routes and the MQTT Provider (only the root). Configuration data of the Route objects are available.
	Merged	4	Produces a merged list of all available RouteTargets with configuration data (merged from Actual and Configured).
	Default	4	The default (actually Merged)

Reference

[TwinCAT Namespace \[► 59\]](#)

6.1.25 RoutePersistenceType Enumeration

Route persistence type (Static or Temporary)

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public enum RoutePersistenceType
```

Members

	Member name	Value	Description
	None	0	None
	Server	0	Server
	Temporary	1	Temporary route
	Static	2	Static route

Reference

[TwinCAT Namespace](#) [▶ 59]

6.1.26 HandleBagNotInitializedException Class

Class ClientNotConnectedException

Inheritance Hierarchy

System.Object
 System.Exception
 TwinCAT.AdsException [▶ 61]
 TwinCAT.HandleBagNotInitializedException

Namespace: [TwinCAT](#) [▶ 59]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
[SerializableAttribute]
public class HandleBagNotInitializedException : AdsException
```

The HandleBagNotInitializedException type exposes the following members.

Constructors

	Name	Description
	HandleBagNotInitializedException . [▶ 173]	Initializes a new instance of the HandleBagNotInitializedException class.
	HandleBagNotInitializedException(SerializationInfo, StreamingContext) [▶ 173]	Initializes a new instance of the HandleBagNotInitializedException class.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)

	Name	Description
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

Reference

[TwinCAT Namespace](#) [► 59]

6.1.26.1 HandleBagNotInitializedException Constructor

Overload List

	Name	Description
	HandleBagNotInitial izedException [► 173]	Initializes a new instance of the HandleBagNotInitializedException [► 171] class.
	HandleBagNotInitial izedException(Serial izationInfo, StreamingContext) [► 173]	Initializes a new instance of the HandleBagNotInitializedException [► 171] class.

Reference

[HandleBagNotInitializedException Class](#) [► 171]

[TwinCAT Namespace \[► 59\]](#)

6.1.26.1.1 HandleBagNotInitializedException Constructor

Initializes a new instance of the [HandleBagNotInitializedException \[► 171\]](#) class.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public HandleBagNotInitializedException()
```

Reference

[HandleBagNotInitializedException Class \[► 171\]](#)

[HandleBagNotInitializedException Overload \[► 172\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.26.1.2 HandleBagNotInitializedException Constructor (SerializationInfo, StreamingContext)

Initializes a new instance of the [HandleBagNotInitializedException \[► 171\]](#) class.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected HandleBagNotInitializedException(
    SerializationInfo serializationInfo,
    StreamingContext streamingContext
)
```

Parameters

serializationInfo	Type: System.Runtime.Serialization.SerializationInfo The serialization information.
streamingContext	Type: System.Runtime.Serialization.StreamingContext The streaming context.

Reference

[HandleBagNotInitializedException Class \[► 171\]](#)

[HandleBagNotInitializedException Overload \[► 172\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.26.2 HandleBagNotInitializedException Properties

The [HandleBagNotInitializedException \[► 171\]](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Reference

[HandleBagNotInitializedException Class \[► 171\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.26.3 HandleBagNotInitializedException Methods

The [HandleBagNotInitializedException \[► 171\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Reference

[HandleBagNotInitializedException Class \[► 171\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.26.4 HandleBagNotInitializedException Events

The [HandleBagNotInitializedException](#) [► 171] type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

Reference

[HandleBagNotInitializedException Class](#) [► 171]

[TwinCAT Namespace](#) [► 59]

6.1.27 IAdsClientSettings Interface

Settings for the Ads client

Namespace: [TwinCAT](#) [► 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public interface IAdsClientSettings
```

The [IAdsClientSettings](#) type exposes the following members.

Properties

	Name	Description
	AutomaticReconnection [► 176]	Gets or sets a value indicating whether Disconnect connections can be reconnected.

Reference

[TwinCAT Namespace](#) [► 59]

6.1.27.1 IAdsClientSettings Properties

The [IAdsClientSettings](#) [► 175] type exposes the following members.

Properties

	Name	Description
	AutomaticReconnection [► 176]	Gets or sets a value indicating whether Disconnect connections can be reconnected.

Reference

[IAdsClientSettings Interface](#) [► 175]

[TwinCAT Namespace \[► 59\]](#)

6.1.27.1.1 IAdsClientSettings.AutomaticReconnection Property

Gets or sets a value indicating whether Disconnect connections can be reconnected.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool AutomaticReconnection { get; set; }
```

Property Value

Type: [Boolean](#)

true if Disconnect connections can be reconnected; otherwise, false.

Reference

[IAdsClientSettings Interface \[► 175\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.28 IValueFactorySettings Interface

Interface IValueFactorySettings

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public interface IValueFactorySettings
```

The IValueFactorySettings type exposes the following members.

Properties

	Name	Description
	ExpandArrayValues [► 177]	Gets or sets a value indicating whether the Value Factory automatically expands ArrayValues to its Elements (Default: true)
	NonCachedArrayElements [► 177]	Gets or sets the setting to create ArrayElements "On-The-Fly" (Default True)
	ValueAccessMode [► 178]	Gets or sets the value access mode.
	ValueCreation [► 178]	Gets or sets the value creation mode.
	ValueUpdateMode [► 179]	Gets or sets the value update mode.

Reference

[TwinCAT Namespace \[► 59\]](#)

6.1.28.1 IValueFactorySettings Properties

The [IValueFactorySettings](#) [▸ 176] type exposes the following members.

Properties

	Name	Description
	ExpandArrayValues [▸ 177]	Gets or sets a value indicating whether the Value Factory automatically expands ArrayValues to its Elements (Default: true)
	NonCachedArrayElements [▸ 177]	Gets or sets the setting to create ArrayElements "On-The-Fly" (Default True)
	ValueAccessMode [▸ 178]	Gets or sets the value access mode.
	ValueCreation [▸ 178]	Gets or sets the value creation mode.
	ValueUpdateMode [▸ 179]	Gets or sets the value update mode.

Reference

[IValueFactorySettings Interface](#) [▸ 176]

[TwinCAT Namespace](#) [▸ 59]

6.1.28.1.1 IValueFactorySettings.ExpandArrayValues Property

Gets or sets a value indicating whether the Value Factory automatically expands ArrayValues to its Elements (Default: true)

Namespace: [TwinCAT](#) [▸ 59]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool ExpandArrayValues { get; set; }
```

Property Value

Type: [Boolean](#)

true if [create array elements]; otherwise, false.

Remarks

Actually, this takes place when DynamicSymbols are created. Array Elements are created on-access to the Array, so that they can be accessed by a JSON Serializer without further expanding. If this property is set to false, then an array element is only created on access!

Reference

[IValueFactorySettings Interface](#) [▸ 176]

[TwinCAT Namespace](#) [▸ 59]

6.1.28.1.2 IValueFactorySettings.NonCachedArrayElements Property

Gets or sets the setting to create ArrayElements "On-The-Fly" (Default True)

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool NonCachedArrayElements { get; set; }
```

Property Value

Type: [Boolean](#)

The value access mode.

Reference

[IValueFactorySettings Interface \[► 176\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.28.1.3 IValueFactorySettings.ValueAccessMode Property

Gets or sets the value access mode.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ValueAccessMode ValueAccessMode { get; set; }
```

Property Value

Type: [ValueAccessMode \[► 3228\]](#)

The value access mode.

Reference

[IValueFactorySettings Interface \[► 176\]](#)

[TwinCAT Namespace \[► 59\]](#)

6.1.28.1.4 IValueFactorySettings.ValueCreation Property

Gets or sets the value creation mode.

Namespace: [TwinCAT \[► 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ValueCreationModes ValueCreation { get; set; }
```

Property Value

Type: [ValueCreationModes \[► 3223\]](#)

The dynamic value mode.

Reference

[IValueFactorySettings Interface \[▶ 176\]](#)

[TwinCAT Namespace \[▶ 59\]](#)

6.1.28.1.5 IValueFactorySettings.ValueUpdateMode Property

Gets or sets the value update mode.

Namespace: [TwinCAT \[▶ 59\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ValueUpdateMode ValueUpdateMode { get; set; }
```

Property Value

Type: [ValueUpdateMode \[▶ 3229\]](#)

The value update mode.

Remarks

Mode	Description
Immediately [▶ 3229]	Writes the values of this DynamicValue instantly when setting its value or the value of its child members/elements.
Triggered [▶ 3229]	Caches internally the value of this DynamicValue until the DynamicValue.Write method is called. This reduces ADS roundtrips, if one or more member/element values should be changed. Furthermore the write on the destination system happens consistently in one ADS Write operation, which could be important for dependent properties/members/elements.

Reference





[IValueFactorySettings Interface \[▶ 176\]](#)

[TwinCAT Namespace \[▶ 59\]](#)






















6.2 TwinCAT.Ads Namespace

ADS root namespace.


Classes

	Class	Description
	AdsClient [▶ 183]	ADS Client / ADS Communication object.
	AdsClientExtensions [▶ 1215]	Class AdsClientExtensions.
	AdsClientSettings [▶ 399]	Settings object for the AdsClient [▶ 183] class.
	AdsConnection [▶ 412]	ADS Connection class

	Class	Description
	AdsDataTypeArrayInfo [▶ 660]	Array definition for a single dimension.
	AdsErrorCodeExtensions [▶ 680]	Class AdsErrorCodeExtensions.
	AdsErrorException [▶ 682]	The exception that is thrown when an ADS error occurs.
	AdsInvalidNotificationException [▶ 690]	This AdsInvalidNotificationException is created if the length of the notification data is 0. This indicates that the notification handle is not valid any more. This exception is passed to the AdsNotificationErrorEvent.
	AdsNotificationErrorEventArgs [▶ 694]	Arguments for the AdsNotificationError [▶ 993] events.
	AdsNotificationEventArgs [▶ 695]	Event argument class for AdsNotification [▶ 991] events.
	AdsNotificationEventArgs [▶ 699]	Arguments for AdsNotificationEx [▶ 993] events.
	AdsNotificationsUnregisteredEventArgs [▶ 1228]	Class AdsNotificationsUnregisteredEventArgs. This class cannot be inherited. Implements the EventArgs
 	AdsSession [▶ 701]	AdsSession class
	AdsSessionBase [▶ 716]	Abstract base class for ADS Sessions.
	AdsStateChangedEventArgs [▶ 729]	Arguments for the AdsStateChanged [▶ 1061] event.
	AdsStateChangedEventArgs2 [▶ 732]	Event Arguments for AdsStateChanged events.
	AdsSumCommandException [▶ 735]	The exception that is thrown when an ADS SumCommandBase error occurs.
	AdsSymbolVersionChangedEventArgs [▶ 739]	Arguments for the AdsSymbolVersionChanged [▶ 1065] event.
	AdsVersion [▶ 743]	The structure contains the version number, revision number and build number.
	AmsAddress [▶ 752]	Ams/Ads Address
	AmsNetId [▶ 767]	AMS/ADS Net ID
	AmsRouterNotificationEventArgs [▶ 798]	Arguments for the IRouterNotificationProvider [▶ 1099] events.

	Class	Description
	DeviceInfo [▶ 801]	The structure contains the name and the version information of the device.
	Notification [▶ 1100]	Class Notification. Implements the INotification [▶ 1096]
	NotificationSettings [▶ 1106]	Notification communication settings
	ResultAds [▶ 1116]	Base class for an (asynchronous) ADS Task Result
	ResultAnyValue [▶ 1129]	Result object for asynchronous reading an 'AnyValue'/Primitive Value via tasks.
	ResultDeviceInfo [▶ 1132]	Ads Task Result for DeviceInfo [▶ 1134] requests (async operations).
	ResultHandle [▶ 1136]	Result object for asynchronous registering an ADS Handle via tasks.
	ResultRead [▶ 1143]	Asynchronous ADS Read result.
	ResultReadAdsState [▶ 1146]	Result object for asynchronous reading AdsStates via tasks.
	ResultReadBytes [▶ 1150]	ADS Task Result returning Read data for async Read operations.
	ResultReadDeviceState [▶ 1157]	Result object for asynchronous ADS ReadDeviceState tasks.
	ResultReadWrite [▶ 1163]	Result object for asynchronous ADS ReadWrite tasks.
	ResultReadWriteBytes [▶ 1170]	Result object for asynchronous ADS ReadWrite tasks.
	ResultRpcMethod [▶ 1176]	Class representing a result of an asynchronous RpcMethod call. Implements the ResultAds [▶ 1116]
	ResultValue.TValue. [▶ 1181]	ADS Result object returning a generic value result (TValue) (asynchronous read). Implements the ResultAds [▶ 1116]
	ResultValue2.I, V. [▶ 1234]	Result object of an asynchronous read. Implements the ResultAccess [▶ 3197]
	ResultWrite [▶ 1187]	Result for asynchronous ADS write tasks.
	ResultWriteControl [▶ 1236]	Class ResultWriteControl. Implements the ResultAds [▶ 1116]
	SessionSettings [▶ 1193]	Session settings class
	TaskExtensions [▶ 1209]	TaskExtensions for Task Cancellation and Timeout
	ValueNotificationEventArgs.T. [▶ 1213]	Arguments for AdsNotificationEx [▶ 993] events.












Structures

	Structure	Description
	StateInfo [▶ 1200]	The structure contains the ADS state and device state.

Interfaces

	Interface	Description
	IAdsAnyAccess [▶ 804]	Interface for accessing ADS 'Any' objects.
	IAdsConnectAddresses [▶ 835]	Interface for method to connect the ADS client via AMS Address.
	IAdsConnection [▶ 876]	ADS Connection interface
	IAdsDisposableConnection [▶ 915]	Interface IAdsDisposableConnection Implements the IAdsConnectAddresses [▶ 835] Implements the IRouterNotificationProvider [▶ 1099] Implements the IAdsSymbolChangedProvider [▶ 1061] Implements the IDisposable
	IAdsHandle [▶ 954]	Interface for ADS access via variable handle
	IAdsNotifications [▶ 966]	Interface for Notification management.
	IAdsReadWrite [▶ 995]	Interface for ADS Read/Write access via IndexGroup / IndexOffset
	IAdsReadWrite2 [▶ 1000]	Interface for ADS Read/Write access via IndexGroup / IndexOffset
	IAdsReadWriteTimeoutAccess [▶ 1005]	Interface IAdsReadWriteTimeoutAccess
	IAdsRpclInvoke [▶ 1012]	Interface IAdsRpclInvoke
	IAdsSession [▶ 1033]	Interface IAdsSession
	IAdsSessionSettings [▶ 1037]	Interface for ADS Session Settings
	IAdsStateControl [▶ 1039]	Interface for reading and controlling the ADS state.
	IAdsStateControlTimeout [▶ 1048]	Interface IAdsStateControlTimeout
	IAdsStateObserver [▶ 1054]	Interface for an AdsState observer
	IAdsStateProvider [▶ 1056]	Interface IAdsStateProvider
	IAdsSymbolChangedProvider [▶ 1061]	Interface IAdsSymbolChangedProvider
	IAdsSymbolicAccess [▶ 1065]	Interface for symbolic ads access.
	INotification [▶ 1096]	Common INotification interface
	INotificationSettings [▶ 1098]	Interface for Notification Settings Implements the IComparable.T .
	IRouterNotificationProvider [▶ 1099]	Interface for AMS Router Notifications.

Enumerations

	Enumeration	Description
	AdsCommandId [▶ 404]	AdsCommandId Enumeration
	AdsDataTyped [▶ 663]	ADS data types.
	AdsErrorCode [▶ 664]	ADS request return codes.
	AdsState [▶ 729]	Describes the AdsState.
	AdsStateCommand [▶ 1233]	Describes the AdsState Commands
	AdsTransMode [▶ 740]	ADS Transmission Mode for ADS Notifications.
	AmsPort [▶ 795]	Ams Ports Definitions.
	AmsPortRange [▶ 1233]	Defines ranges of AmsPorts . [▶ 795]
	AmsRouterState [▶ 800]	State of the AMS Router.
	IndexGroupSymbol Access [▶ 1231]	ADS IndexGroups defined for ADS Symbol Handling
	TransportProtocols [▶ 1212]	Enum ADS TransportProtocol

Also see about this

-  [IAdsSymbolTableProvider Interface](#) [▶ 1093]

6.2.1 AdsClient Class

ADS Client / ADS Communication object.

Inheritance Hierarchy

System.Object
TwinCAT.Ads.AdsClient

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#












```
public sealed class AdsClient : IAdsDisposableConnection,
    IAdsConnectAddress, IAdsConnection, IConnection, IConnectionStateProvider, IAdsNotifications,
    IAdsSymbolicAccess, IAdsAnyAccess, IAdsHandle, IAdsReadWrite2, IAdsReadWrite,
    IAdsStateProvider, IAdsStateControl, IAdsSymbolChangedProvider, IAdsRpcInvoke, IRouterNotificati
onProvider,
    IDisposable
```

The AdsClient type exposes the following members.



Constructors

	Name	Description
	AdsClient [▶ 397]	Initializes a new instance of the AdsClient class.
	AdsClient(AdsClientSettings) [▶ 397]	Initializes a new instance of the AdsClient class with the specified settings.
	AdsClient(ILogger) [▶ 397]	Initializes a new instance of the AdsClient class.
	AdsClient(ISession, ILogger) [▶ 398]	Initializes a new instance of the AdsClient class bound to a session.
	AdsClient(ISession, AdsClientSettings, ILogger) [▶ 398]	Initializes a new instance of the AdsClient class.





Properties
















	Name	Description
	Address [▶ 209]	Gets the target AmsAddress [▶ 752] of of the established ADS connection (Destination side).
	ClientAddress [▶ 209]	Get the client AmsAddress [▶ 752] (Source side).
	DefaultValueEncoding [▶ 209]	Gets the default value encoding.
	Id [▶ 210]	Gets the AdsClient Identifier.
	IsConnected [▶ 210]	Gets a value indicating whether the local ADS port was opened successfully. It does not indicate if the target port is available. Use the method ReadState to determine if the target port is available.
	IsDisposed [▶ 211]	Gets a value indicating whether this instance is disposed.
	IsLocal [▶ 211]	Gets a value indicating whether the ADS client is connected to a ADS Server on the local computer.
	Logger [▶ 213]	Gets the logger interface.
	Session [▶ 212]	Gets the session that initiated this IConnection [▶ 79]
	SymbolEncoding [▶ 212]	Gets the symbol encoding.
	Timeout [▶ 213]	Sets the timeout for the ads communication. Unit is in ms.














Methods

















	Name	Description
	AddDeviceNotification(String, Int32, NotificationSettings, Object) [▶ 233]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 390] event.
	AddDeviceNotification(UInt32, UInt32,	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event.

	Name	Description
	Int32, NotificationSettings, Object) [▶ 234]	
	AddDeviceNotificationAsync(String, Int32, NotificationSettings, Object, CancellationTok en) [▶ 236]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 390] event.
	AddDeviceNotificationAsync(UInt32, UInt32, Int32, NotificationSettings, Object, CancellationTok en) [▶ 237]	Adds a device notification as an asynchronous operation.
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type) [▶ 239]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 391] event.
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32.) [▶ 240]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 391] event.
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type) [▶ 241]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 391] event.
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32.) [▶ 242]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive ('AnyType') types are allowed for the parameter type.
	AddDeviceNotificationExAsync(String, NotificationSettings, Object, Type, .Int32., CancellationTok en) [▶ 243]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 391] event.












	Name	Description
	AddDeviceNotificationExAsync(UInt32, UInt32, NotificationSettings, Object, Type, .Int32, CancellationToken) [▶ 245]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 391] event.
	CleanupSymbolTable [▶ 246]	Clears the internal symbol cache.
	Close [▶ 247]	Closes this AdsClient
	Connect(AmsAddress) [▶ 248]	Connects the target
	Connect(AmsPort) [▶ 249]	Connects to the local target ADS Device.
	Connect(Int32) [▶ 247]	Connects to the local target ADS Device.
	Connect(AmsNetId, Int32) [▶ 250]	Connects to the target ADS Device.
	Connect(AmsNetId, AmsPort) [▶ 250]	Connects to the target ADS Device.
	Connect(String, Int32) [▶ 249]	Connects to the target ADS Device.
	ConnectAndWaitAsync [▶ 251]	Connects to the target address and waits until the AdsClient is disconnected asynchronously.
	CreateVariableHandle [▶ 251]	Determines the Symbol handle by its instance path synchronously.
	CreateVariableHandleAsync [▶ 252]	Determines the Symbol handle by its instance path asynchronously.
	DeleteDeviceNotification [▶ 253]	Deletes a registered notification.
	DeleteDeviceNotificationAsync [▶ 254]	Deletes a registered notification asynchronously.
	DeleteVariableHandle [▶ 255]	Releases the specified symbol/variable handle synchronously.
	DeleteVariableHandleAsync [▶ 255]	Releases the specified symbol/variable handle asynchronously.
	Disconnect [▶ 256]	Disconnects this AdsClient from the local ADS router.
	Dispose [▶ 257]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

















	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize [▶ 257]	Finalizes an instance of the AdsClient class. (Overrides Object.Finalize .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	InvokeRpcMethodAsync(IRpcCallableInstance, IRpcMethod, Object, CancellationToken) [▶ 269]	invoke RPC method as an asynchronous operation.
 	InvokeRpcMethodAsync(String, String, Object, CancellationToken) [▶ 264]	Invokes the specified RPC Method asynchronously
	InvokeRpcMethodAsync(IRpcCallableInstance, IRpcMethod, Object, AnyTypeSpecifier, AnyTypeSpecifier, CancellationToken) [▶ 268]	invoke RPC method as an asynchronous operation.
 	InvokeRpcMethodAsync(String, String, Object, AnyTypeSpecifier, AnyTypeSpecifier, CancellationToken) [▶ 266]	Invokes the specified RPC Method asynchronously
	Read(UInt32, Memory.Byte) [▶ 271]	Reads data synchronously from an ADS device and writes to the specified readBuffer.
	Read(UInt32, UInt32, Memory.Byte) [▶ 272]	Reads data synchronously from an ADS device and writes it to the given readBuffer
	ReadAny(UInt32, Type) [▶ 275]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, Type, Int32) [▶ 276]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type) [▶ 278]	Reads data synchronously from an ADS device and writes it to an object.















	Name	Description
	ReadAny(UInt32, UInt32, Type, .Int32.) [▶ 279]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32) [▶ 273]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, .Int32.) [▶ 274]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, UInt32) [▶ 275]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, UInt32, .Int32.) [▶ 277]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, Type, CancellationToken) [▶ 283]	Reads the value of an Anytype (Primitive type) asynchronously.
	ReadAnyAsync(UInt32, Type, .Int32., CancellationToken) [▶ 285]	Reads the value of an Anytype (Primitive type) asynchronously.
	ReadAnyAsync(UInt32, UInt32, Type, CancellationToken) [▶ 287]	Reads the value of an Anytype (Primitive type) asynchronously.
	ReadAnyAsync(UInt32, UInt32, Type, .Int32., CancellationToken) [▶ 287]	Reads the value of an Anytype (Primitive type) asynchronously.
	ReadAnyAsync.T.(UInt32, CancellationToken) [▶ 281]	Reads data synchronously from an ADS device.
	ReadAnyAsync.T.(UInt32, .Int32., CancellationToken) [▶ 282]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T.(UInt32, UInt32, CancellationToken) [▶ 284]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T.(UInt32, UInt32, .Int32., CancellationToken) [▶ 285]	Reads data asynchronously from an ADS device and writes it to an object.














	Name	Description
	ReadAnyString(UInt32, Int32, Encoding) [▶ 289]	Reads a string from the specified symbol/variable.
	ReadAnyString(UInt32, UInt32, Int32, Encoding) [▶ 289]	Reads as string from a specified address.
	ReadAnyStringAsync(UInt32, Int32, Encoding, CancellationToken) [▶ 291]	Reads a string asynchronously from the specified symbol/variable
	ReadAnyStringAsync(UInt32, UInt32, Int32, Encoding, CancellationToken) [▶ 292]	Reads a string from a specified address asynchronously.
	ReadAsync(UInt32, Memory.Byte, CancellationToken) [▶ 294]	Reads the value data of the symbol asynchronously into the readBuffer.
	ReadAsync(UInt32, UInt32, Memory.Byte, CancellationToken) [▶ 295]	Reads the data asynchronously from specified IndexGroup/IndexOffset
	ReadDataType [▶ 296]	Call this method to obtain information about the specified data type.
	ReadDataTypeAsync [▶ 297]	read data type as an asynchronous operation.
	ReadDeviceInfo [▶ 298]	Reads the identification and version number of an ADS server.
	ReadDeviceInfoAsync [▶ 298]	Reads the identification and version number of an ADS server.
	ReadState [▶ 299]	Reads the ADS status and the device status from an ADS server.
	ReadStateAsync [▶ 299]	Reads the ADS status and the device status from an ADS server.
	ReadSymbol [▶ 300]	Call this method to obtain information about the individual symbols (variables) in ADS devices.
	ReadSymbolAsync [▶ 301]	Call this method to obtain information about the individual symbols (variables) in ADS devices.
	ReadValue(ISymbol) [▶ 303]	Reads the value of a symbol and returns it as an object.
	ReadValue(String, Type) [▶ 304]	Reads the value of a symbol and returns the value as object. The parameter type must have the same layout as the ADS symbol.












	Name	Description
	ReadValue.T. (ISymbol) [▶ 303]	Reads the value of a symbol and returns it as an object.
	ReadValue.T.(String) [▶ 302]	Reads the value of a symbol and returns the value. The parameter type must have the same layout as the ADS symbol.
	ReadValueAsync(ISymbol, CancellationToken) [▶ 306]	Reads the value of a symbol asynchronously and returns it as an object. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. Arrays and structures cannot be read.
	ReadValueAsync(String, Type, CancellationToken) [▶ 308]	Reads the value of a symbol asynchronously.
	ReadValueAsync.TV alue.(ISymbol, CancellationToken) [▶ 307]	Reads the value of a symbol asynchronously and returns it as an object.
	ReadValueAsync.TV alue.(String, CancellationToken) [▶ 306]	Reads the value of a symbol asynchronously.
	ReadWrite(UInt32, Memory.Byte., ReadOnlyMemory.B yte.) [▶ 310]	Writes data synchronously to an ADS device and then Reads data from that target.
	ReadWrite(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.B yte.) [▶ 311]	Writes data synchronously to an ADS device and then Reads data from this device into the readBuffer
	ReadWriteAsync(UInt32, Memory.Byte., ReadOnlyMemory.B yte, CancellationToken) [▶ 314]	ReadWrites value data asynchronously to/from the symbol represented by the variableHandle.
	ReadWriteAsync(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.B yte, CancellationToken) [▶ 315]	Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer
	RegisterAdsStateChangedAsync [▶ 316]	Registers for AdsStateChanged [▶ 392] events as an asynchronous operation.






	Name	Description
	RegisterSymbolVersionChanged [▶ 387]	Registers for an AdsSymbolVersionChanged [▶ 394] event as an asynchronous operation.
	RegisterSymbolVersionChangedAsync [▶ 317]	Registers for an AdsSymbolVersionChanged [▶ 394] event as an asynchronous operation.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryAddDeviceNotification(String, Int32, NotificationSettings, Object, UInt32.) [▶ 318]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 390] event.
	TryAddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.) [▶ 319]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event.
	TryAddDeviceNotificationEx(String, NotificationSettings, Object, Type, Int32, UInt32.) [▶ 321]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 391] event.
	TryAddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, Int32, UInt32.) [▶ 322]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event.
	TryCreateVariableHandle [▶ 323]	Determines the Symbol handle by its instance path synchronously.
	TryDeleteDeviceNotification(UInt32) [▶ 325]	Deletes a registered notification.
	TryDeleteDeviceNotification(UInt32, Int32) [▶ 325]	Deletes a registered notification.
	TryDeleteVariableHandle [▶ 326]	Releases the specified symbol/variable handle synchronously.

	Name	Description
 	TryInvokeRpcMethod(String, String, Object, Object.) [▶ 328]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(String, String, Object, Object, Object, Object.) [▶ 330]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(IRpcCallableInstance, IRpcMethod, Object, AnyTypeSpecifier, AnyTypeSpecifier, Object, Object.) [▶ 333]	Tries the invoke RPC method.
 	TryInvokeRpcMethod(String, String, Object, AnyTypeSpecifier, AnyTypeSpecifier, Object, Object.) [▶ 331]	Invokes the RPC Method
	TryRead(UInt32, Memory.Byte, Int32.) [▶ 337]	Reads the value synchronously data of the symbol, that is represented by the variable handle into the readBuffer.
	TryRead(UInt32, UInt32, Memory.Byte, Int32.) [▶ 338]	Reads value data from the specified IndexGroup/IndexOffset to the specified memory location.
	TryReadDataType [▶ 338]	Call this method to obtain information about the specified data type.
	TryReadState [▶ 339]	Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful.
	TryReadSymbol [▶ 340]	Call this method to obtain information about the individual symbols (variables) in ADS devices.
	TryReadValue(ISymbol, Object.) [▶ 342]	Reads the value of a symbol and returns it as an object.
	TryReadValue(String, Type, Object.) [▶ 344]	Reads the value of a symbol and returns the value as object.
	TryReadValue.T.(ISymbol, T.) [▶ 343]	Reads the value of a symbol and returns it as an object.










	Name	Description
	TryReadValue.T. (String, T.) [▶ 341]	Reads the value of a symbol and returns the value as object.
	TryReadWrite(UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.) [▶ 346]	ReadWrites value data synchronously to/from the symbol represented by the variableHandle.
	TryReadWrite(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.) [▶ 347]	Writes data synchronously to an ADS device and reads data from that device.
	TryResurrect [▶ 348]	Resurrects the connection
	TryWrite(UInt32, ReadOnlyMemory.Byte.) [▶ 350]	Writes the value data synchronously that is represented in the writeBuffer to the symbol with the specified variableHandle.
	TryWrite(UInt32, UInt32, ReadOnlyMemory.Byte.) [▶ 351]	Writes data synchronously to an ADS device.
	TryWriteControl(StateInfo) [▶ 352]	Changes the ADS status and the device status of an ADS server.
	TryWriteControl(StateInfo, ReadOnlyMemory.Byte.) [▶ 353]	Changes the ADS status and the device status of an ADS server.
	TryWriteValue(ISymbol, Object) [▶ 356]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	TryWriteValue(String, Object) [▶ 354]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	TryWriteValue.T. (ISymbol, T) [▶ 356]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	TryWriteValue.T. (String, T) [▶ 355]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	UnregisterAdsStateChangedAsync [▶ 357]	unregister ads state changed as an asynchronous operation.
	UnregisterSymbolVersionChanged [▶ 388]	Unregisters the symbol version changed.

	Name	Description
	UnregisterSymbolVersionChangedAsync [▶ 358]	Unregisters from an AdsSymbolVersionChanged [▶ 394] event as an asynchronous operation.
	Write(UInt32, ReadOnlyMemory.BYTE) [▶ 361]	Writes data synchronously to an ADS device.
	Write(UInt32, UInt32) [▶ 360]	Triggers a 'Write' call to the ADS device at the specified address.
	Write(UInt32, UInt32, ReadOnlyMemory.BYTE) [▶ 361]	Writes data synchronously to an ADS device.
	WriteAny(UInt32, Object) [▶ 362]	Writes an object synchronously to an ADS device.
	WriteAny(UInt32, Object, .Int32) [▶ 363]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAny(UInt32, UInt32, Object) [▶ 364]	Writes an object synchronously to an ADS device.
	WriteAny(UInt32, UInt32, Object, .Int32) [▶ 364]	Writes an object synchronously to an ADS device.
	WriteAnyAsync(UInt32, Object, CancellationToken) [▶ 366]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAnyAsync(UInt32, Object, .Int32, CancellationToken) [▶ 367]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAnyAsync(UInt32, UInt32, Object, CancellationToken) [▶ 368]	Write the value of an Anytype (Primitive type) asynchronously.
	WriteAnyAsync(UInt32, UInt32, Object, .Int32, CancellationToken) [▶ 369]	Write the value of an Anytype (Primitive type) asynchronously.
	WriteAnyStringAsync(String, String)	Writes the string (Potentially unsafe!)












	Name	Description
	Int32, Encoding, CancellationToken [▶ 370]	
	WriteAnyStringAsync(UInt32, String, Int32, Encoding, CancellationToken) [▶ 371]	Writes the string (Potentially unsafe!)
	WriteAsync(UInt32, ReadOnlyMemory.Byte, CancellationToken) [▶ 375]	Writes the value data asynchronously that is represented by the writeBuffer to the symbol specified by the variableHandle.
	WriteAsync(UInt32, UInt32, CancellationToken) [▶ 373]	Triggers a write call at the specified IndexGroup/IndexOffset asynchronously.
	WriteAsync(UInt32, UInt32, ReadOnlyMemory.Byte, CancellationToken) [▶ 375]	Writes the data / Value asynchronously into the specified writeBuffer.
	WriteControl(StateInfo) [▶ 376]	Changes the ADS status and the device status of an ADS server.
	WriteControl(StateInfo, ReadOnlyMemory.Byte) [▶ 377]	Changes the ADS status and the device status of an ADS server.
	WriteControlAsync(AdsState, UInt16, CancellationToken) [▶ 379]	Changes the ADS status and device status of the ADS server asynchronously.
	WriteControlAsync(AdsState, UInt16, ReadOnlyMemory.Byte, CancellationToken) [▶ 380]	Writes the state asynchronously
	WriteSymbolAsync [▶ 381]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	WriteValue(ISymbol, Object) [▶ 383]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValue(String, Object) [▶ 382]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

	Name	Description
	WriteValue.T. (ISymbol, T) [▶ 384]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValue.T.(String, T) [▶ 382]	Writes the passed object value to the specified ADS symbol.The parameter type must have the same layout as the ADS symbol.
	WriteValueAsync(ISymbol, Object, CancellationTokentoken) [▶ 386]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValueAsync.T. (ISymbol, T, CancellationTokentoken) [▶ 387]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValueAsync.T. (String, T, CancellationTokentoken) [▶ 385]	Writes the passed object value to the specified ADS symbol.The parameter type must have the same layout as the ADS symbol.






Events













	Name	Description
	AdsNotification [▶ 390]	Occurs when the ADS device sends a notification to the client.
	AdsNotificationError [▶ 391]	Occurs when a exception has occurred during notification management.
	AdsNotificationEx [▶ 391]	Occurs when the ADS devices sends a notification to the client.
	AdsNotificationsInv alidated [▶ 396]	Occurs when Notification Unregistrations / Invalidates are received from the AdsServer
	AdsStateChanged [▶ 392]	Occurs when the ADS state changes.
	AdsSumNotification [▶ 393]	Occurs when Notifications are send (bundled notifications)
	AdsSymbolVersionC hanged [▶ 394]	Occurs when the symbol version has been changed changes.
	ConnectionStateCha nged [▶ 395]	Occurs when the connection state has been changed.
	RouterStateChange d [▶ 395]	(Local) Router state changed event.






Extension Methods















	Name	Description
	PollAdsState(IObservable.Unit.) [▶ 3294]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
 	PollAdsState(IObservable.Unit.) [▶ 1256]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsState(TimeSpan) [▶ 3295]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
 	PollAdsState(TimeSpan) [▶ 1257]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState2(IObservable.Unit.) [▶ 1271]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsState2(TimeSpan) [▶ 1272]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState2Async(IObservable.Unit., Cancellation.Token) [▶ 1274]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsState2Async(TimeSpan, Cancellation.Token) [▶ 1275]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsStateAsync(IObservable.Unit., Cancellation.Token) [▶ 3296]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
 	PollAdsStateAsync(IObservable.Unit., Cancellation.Token) [▶ 1259]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsStateAsync(TimeSpan, Cancellation.Token) [▶ 3298]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)


	Name	Description
 	PollAdsStateAsync(TimeSpan, CancellationToken) [► 1260]	Overloaded. Gets an observable sequence of AdsState [► 729]s via Polling. (Defined by AdsClientExtensions [► 1249].)
	PollDeviceState(IObservable.Unit.) [► 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [► 1157]s via Polling. (Defined by AdsClientExtensions [► 1249].)
 	PollDeviceState(TimeSpan) [► 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [► 1157]s via Polling. (Defined by AdsClientExtensions [► 1249].)
 	PollDeviceStateAsync(IObservable.Unit, CancellationToken) [► 1278]	Overloaded. Gets an observable sequence of ResultReadDeviceState [► 1157]s via Polling. (Defined by AdsClientExtensions [► 1249].)
 	PollDeviceStateAsync(TimeSpan, CancellationToken) [► 1279]	Overloaded. Gets an observable sequence of ResultReadDeviceState [► 1157]s via Polling. (Defined by AdsClientExtensions [► 1249].)
	PollSystemServiceState [► 3282]	Polls the state of the system service. (Defined by SystemServiceExtension [► 3281].)
	PollSystemServiceStateAsync [► 3283]	Polls the system service state asynchronously (Defined by SystemServiceExtension [► 3281].)
	PollValues(ISymbol, Type, IObservable.Unit.) [► 1310]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [► 1281].)
	PollValues(ISymbol, Type, TimeSpan) [► 1311]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [► 1281].)
	PollValues(String, Type, IObservable.Unit.) [► 1300]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [► 1281].)
	PollValues(String, Type, TimeSpan) [► 1301]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [► 1281].)
	PollValues(ISymbol, Type, .Int32., TimeSpan) [► 1316]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [► 1281].)
	PollValues(ISymbol, Type, IObservable.Unit., Func.Exception, Object.) [► 1317]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [► 1281].)

	Name	Description
	PollValues(ISymbol, Type, TimeSpan, Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32, TimeSpan) [▶ 1304]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, IObservable.Unit, Func.Exception, Object.) [▶ 1305]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, TimeSpan, Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32, IObservable.Unit, Func.Exception, Object.) [▶ 1321]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32, TimeSpan, Func.Exception, Object.) [▶ 1322]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32, IObservable.Unit, Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32, TimeSpan, Func.Exception, Object.) [▶ 1307]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, IObservable.Unit) [▶ 1308]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, TimeSpan) [▶ 1309]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, IObservable.Unit) [▶ 1294]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, TimeSpan) [▶ 1295]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	PollValues.T. (ISymbol , IObservable.Unit , Func.Exception , T.) [▶ 1314]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol , TimeSpan , Func.Exception , T.) [▶ 1315]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol , .Int32 , IObservable.Unit .) [▶ 1312]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	PollValues.T. (ISymbol , .Int32 , TimeSpan) [▶ 1313]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String , IObservable.Unit , Func.Exception , T.) [▶ 1298]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String , TimeSpan , Func.Exception , T.) [▶ 1299]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32 , IObservable.Unit .) [▶ 1296]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	PollValues.T. (String , .Int32 , TimeSpan) [▶ 1297]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol , .Int32 , IObservable.Unit , Func.Exception , T.) [▶ 1318]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	PollValues.T. (ISymbol , .Int32 , TimeSpan , Func.Exception , T.) [▶ 1319]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32 , IObservable.Unit , Func.Exception , T.) [▶ 1302]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
 	PollValues.T. (String, .Int32., TimeSpan, Func.Exception, T.) [▶ 1303]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues2.T. (ISymbol, .Int32., IObservable.Unit.) [▶ 1330]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues2.T. (String, .Int32., IObservable.Unit.) [▶ 1329]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	ReadSysServState [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (Defined by SystemServiceExtension [▶ 3281].)
	ReadSysServStateAs ync [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (asynchronous) (Defined by SystemServiceExtension [▶ 3281].)
	ReadWithFallback(U Int32, UInt32, Memory.Byte., UInt32, Boolean.) [▶ 1219]	Overloaded. Ads Read with fallback. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallback(U Int32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., Boolean.) [▶ 1220]	Overloaded. Ads Read with Fallback. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallbackAs ync(UInt32, UInt32, UInt32, Memory.Byte., CancellationToken) [▶ 1221]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallbackAs ync(UInt32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., CancellationToken) [▶ 1222]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)

	Name	Description
	RepeatedRead(UInt32, UInt32, Memory.Byte, Int32, TimeSpan) [► 1224]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [► 1215] .)
	RepeatedRead(UInt32, UInt32, Memory.Byte, Int32, TimeSpan, Func.ResultRead, Boolean.) [► 1225]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [► 1215] .)
	RepeatedReadAsync(UInt32, UInt32, Memory.Byte, Int32, TimeSpan, CancellationToken) [► 1226]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [► 1215] .)
	RepeatedReadAsync(UInt32, UInt32, Memory.Byte, Int32, TimeSpan, Func.ResultRead, Boolean, CancellationToken) [► 1227]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [► 1215] .)
	RestartTwinCATAsync [► 3285]	Sends a SystemService Restart to the target system and waits until the system has finished the state changes (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [► 3281] .)
	SetAdsState [► 3299]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [► 3292] .)
	SetAdsStateAsync [► 3300]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [► 3292] .)
	WaitUntilRestarted [► 3286]	Waits until the Restart is detected on the client (SystemService, Port 10000) (Defined by SystemServiceExtension [► 3281] .)
	WaitUntilRestartedAsync [► 3287]	Waits until the SystemService Restart is detected on the client (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [► 3281] .)
 	WhenAdsStateChanges [► 1261]	Gets an observable sequence of AdsState [► 729] s. (Defined by AdsClientExtensions [► 1249] .)
	WhenNotification(ISymbol) [► 1263]	Overloaded. Gets an observable sequence of Notification [► 1100] s. (Defined by AdsClientExtensions [► 1249] .)
 	WhenNotification(ISymbolCollection) [► 1264]	Overloaded. Gets an observable sequence of Notification [► 1100] objects. (Defined by AdsClientExtensions [► 1249] .)

	Name	Description
 	WhenNotification(IList.ISymbol, NotificationSettings) [▶ 1268]	Overloaded. Gets an observable sequence of Notification [▶ 1100] objects. (Defined by AdsClientExtensions [▶ 1249].)
	WhenNotification(ISymbol, NotificationSettings) [▶ 1265]	Overloaded. Gets an observable sequence of SymbolValueNotification [▶ 1331]s. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenNotification(String, Type, NotificationSettings) [▶ 1324]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [▶ 1281].)
 	WhenNotification.T.(String, NotificationSettings) [▶ 1323]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [▶ 1281].)
	WhenSymbolVersionChanges. [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
	WhenSymbolVersionChanges(IScheduler) [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenValueChanged [▶ 1345]	Observable sequence of Values driven by ADS Notifications on the specified symbol. (Defined by ValueSymbolExtensions [▶ 1333].)
 	WriteValues.T.(String, IObservable.T.) [▶ 1326]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)
 	WriteValues.T.(String, IObservable.T., Action.Exception.) [▶ 1327]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)

Remarks

The class `AdsClient` enables synchronous/asynchronous access to data of an ADS Device.

Examples

The following sample shows how to instantiate and use the `AdsClient` class.

AdsClient Demo (async)

```
using System;
using System Buffers.Binary;
using System.Threading;
using System.Threading.Tasks;
using TwinCAT.Ads;
using TwinCAT.TypeSystem;
```

```

namespace Sample
{
    class ClientAsync
    {
        /// <summary>
        /// Defines the entry point of the application.
        /// </summary>
        /// <param name="args">The arguments.</param>
        static async void Main(string[] args)
        {
            AmsAddress address = ArgParser.Parse(args);
            CancellationTokenSource cancelSource = new CancellationTokenSource();
            CancellationToken cancel = cancelSource.Token;

            using (AdsClient client = new AdsClient())
            {
                // Connect to Address
                client.Connect(address.NetId, address.Port); // Connect to Port (851, first PLC by default)

                // Read the identification and version number of the device
                ResultDeviceInfo resultDeviceInfo = await client.ReadDeviceInfoAsync(cancel);

                if (resultDeviceInfo.Succeeded)
                {
                    DeviceInfo deviceInfo = resultDeviceInfo.DeviceInfo;
                    Version version = deviceInfo.Version.ConvertToStandard();
                    Console.WriteLine(string.Format("DeviceName: {0}", deviceInfo.Name));
                    Console.WriteLine(string.Format("DeviceVersion: {0}", version.ToString(3)));
                }
                // Read the state of the device
                ResultReadDeviceState resultReadDeviceState = await client.ReadStateAsync(cancel);
                AdsState state = AdsState.Invalid;

                if (resultReadDeviceState.Succeeded)
                {
                    StateInfo stateInfo = resultReadDeviceState.State;

                    state = stateInfo.AdsState;
                    short deviceState = stateInfo.DeviceState;
                    Console.WriteLine(string.Format("DeviceState: {0}", deviceState));
                    Console.WriteLine(string.Format("AdsState : {0}", state));
                }

                // Write ADS Commands (write state) to target
                // Set PLC to Run

                if (state == AdsState.Stop)
                {
                    await client.WriteControlAsync(AdsState.Run, 0, cancel);
                }

                //create variable handle for Plc Project Name (automatic generated symbol in PLC)
                ResultHandle resultHandle = await client.CreateVariableHandleAsync("TwinCAT_SystemInfoVarList._AppInfo.ProjectName", cancel);

                if (resultHandle.Succeeded)
                {
                    uint handleProjectName = resultHandle.Handle;
                    uint handleNotification = 0; // Notification Handle for Task1 CycleCount changes

                    try
                    {
                        // Read value from target and Marshal data into string
                        byte[] readData = new byte[256];

                        ResultRead resultRead = await client.ReadAsync(handleProjectName, readData.AsMemory(), cancel);

                        if (resultRead.Succeeded)
                        {
                            PrimitiveTypeMarshaler marshaler = PrimitiveTypeMarshaler.Default;

                            string projectName = null;
                            int unmarshaledBytes = marshaler.Unmarshal(readData, out projectName);
                            Console.WriteLine(string.Format("ProjectName : {0}", projectName));
                        }

                        // _notificationBuffer = new byte[4]; // Sizeof UDINT
                    }
                }
            }
        }
    }
}

```

```

        int size = sizeof(UInt32); // Sizeof UDINT

        client.AdsNotification += client_NotificationEvent; // Register for Notification event

        //Register Notification for Task1 CycleCount symbol (automatic generated symbol in PLC)
        resultHandle = await client.AddDeviceNotificationAsync("TwinCAT_SystemInfoVarList._Task1
nfo[1].CycleCount", size, NotificationSettings.Default, null, cancel);

        if (resultHandle.Succeeded)
        {
            handleNotification = resultHandle.Handle;
            // Sleep 10 Seconds to receive events
            await Task.Delay(10000, cancel); // Async Delay (some time for notifications)
        }
        finally
        {
            // Cleanup all handles
            // Dispose all Streams

            ResultAds result = await client.DeleteDeviceNotificationAsync(handleNotification, cancel
);

            // Always delete all variable handles.
            result = await client.DeleteVariableHandleAsync(handleProjectName, cancel);
        }
        Console.WriteLine("");
        Console.WriteLine("Press [Enter] for leave:");
        Console.ReadLine();
    }
}

```

AdsClient Demo (sync)

```

using System;
using System Buffers.Binary;
using System.Threading;
using System.Threading.Tasks;
using TwinCAT.Ads;
using TwinCAT.TypeSystem;

namespace Sample
{
    class Client
    {
        /// <summary>
        /// Defines the entry point of the application.
        /// </summary>
        /// <param name="args">The arguments.</param>
        static void Main(string[] args)
        {
            AmsAddress address = ArgParser.Parse(args);

            using (AdsClient client = new AdsClient())
            {
                // Connect to Address
                client.Connect(address.NetId, address.Port); // Connect to Port (851, first PLC by default)

                // Read the identification and version number of the device
                DeviceInfo deviceInfo = client.ReadDeviceInfo();

                Version version = deviceInfo.Version.ConvertToStandard();
                Console.WriteLine(string.Format("DeviceName: {0}", deviceInfo.Name));
                Console.WriteLine(string.Format("DeviceVersion: {0}", version.ToString(3)));

                // Read the state of the device
                StateInfo stateInfo = client.ReadState();
                AdsState state = AdsState.Invalid;

                state = stateInfo.AdsState;
                short deviceState = stateInfo.DeviceState;
                Console.WriteLine(string.Format("DeviceState: {0}", deviceState));
                Console.WriteLine(string.Format("AdsState : {0}", state));

                // Write ADS Commands (write state) to target
                // Set PLC to Run

                if (state == AdsState.Stop)

```

```

    {
        client.WriteControl(new StateInfo(AdsState.Run, 0));
    }

    //create variable handle for Plc Project Name (automatic generated symbol in PLC)
    uint handleProjectName = client.CreateVariableHandle("TwinCAT_SystemInfoVarList._AppInfo.Pro
jectName");
    uint handleNotification = 0; // Notification Handle for Task1 CycleCount changes

    try
    {
        // Read value from target and Marshal data into string
        byte[] readData = new byte[256];
        int readBytes = client.Read(handleProjectName, readData.AsMemory());

        PrimitiveTypeMarshaler marshaler = PrimitiveTypeMarshaler.Default;

        string projectName = null;
        int unmarshaledBytes = marshaler.Unmarshal(readData, client.DefaultValueEncoding, out pr
ojectName);
        Console.WriteLine(string.Format("ProjectName : {0}", projectName));

        //_notificationBuffer = new byte[4]; // Sizeof UDINT
        int size = sizeof(UInt32); // Sizeof UDINT

        client.AdsNotification += client_NotificationEvent; // Register for Notification event

        //Register Notification for Task1 CycleCount symbol (automatic generated symbol in PLC)
        handleNotification = client.AddDeviceNotification("TwinCAT_SystemInfoVarList._TaskInfo[1
].CycleCount", size, NotificationSettings.Default, null);

        // Sleep 10 Seconds to receive events
        Thread.Sleep(10000); // Sleep to get notifications
    }
    finally
    {
        // Cleanup all handles
        // Dispose all Streams
        client.DeleteDeviceNotification(handleNotification);

        // Always delete all variable handles.
        client.DeleteVariableHandle(handleProjectName);
    }
    Console.WriteLine("");
    Console.WriteLine("Press [Enter] for leave:");
    Console.ReadLine();
}
}

```

Argument Parser

```

public static class ArgParser
{
    /// <summary>
    /// Parses the arguments.
    /// </summary>
    /// <param name="args">The arguments.</param>
    /// <returns>AmsAddress.</returns>
    public static AmsAddress Parse(string[] args)
    {
        AmsNetId netId = AmsNetId.Local;
        int port = 851;

        if (args != null)
        {
            if (args.Length > 0 && args[0] != null)
                netId = AmsNetId.Parse(args[0]);

            if (args.Length > 1 && args[1] != null)
                port = int.Parse(args[1]);
        }
        return new AmsAddress(netId, port);
    }
}

```

The following sample shows how to call (Remote Procedures / Methods) within the PLC directly from the AdsClient class.

RPC Call Example (async)

```

namespace Sample
{
    using System;
    using System.Diagnostics;
    using System.Threading;
    using TwinCAT.Ads;
    using TwinCAT.TypeSystem;

    class RpcCallAsync
    {
        /// <summary>
        /// Defines the entry point of the application.
        /// </summary>
        /// <param name="args">The arguments.</param>
        static async void Main(string[] args)
        {
            //Parse the AmsAddress from command-line arguments
            AmsAddress address = ArgParser.Parse(args);

            // Create the ADS Client
            using (AdsClient client = new AdsClient())
            {
                // Establish Connection
                client.Connect(address);

                // Call a Method that has the following signature (within MAIN Program)

                /* {attribute 'TcRpcEnable'}
                METHOD PUBLIC M_Add : INT
                VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
                END_VAR
                */

                ResultRpcMethod result1 = await client.InvokeRpcMethodAsync("MAIN", "M_Add", new object[] {
(short)1, (short)4 }, CancellationToken.None);
                short result = (short)result1.ReturnValue;

                // Call a Method that has no parameter and returns VOID
                await client.InvokeRpcMethodAsync("MAIN", "M_Method1", new object[] { },CancellationToken.No
ne);
            }
        }
    }
}

```

RPC Call Example (sync)

```

namespace Sample
{
    using System;
    using System.Diagnostics;
    using TwinCAT.Ads;
    using TwinCAT.TypeSystem;

    class RpcCall
    {
        /// <summary>
        /// Defines the entry point of the application.
        /// </summary>
        /// <param name="args">The arguments.</param>
        static void Main(string[] args)
        {
            //Parse the AmsAddress from command-line arguments
            AmsAddress address = ArgParser.Parse(args);

            // Create the ADS Client
            using (AdsClient client = new AdsClient())
            {
                // Establish Connection
                client.Connect(address);

                // Call a Method that has the following signature (within MAIN Program)

                /* {attribute 'TcRpcEnable'}
                METHOD PUBLIC M_Add : INT
                VAR_INPUT

```

```

        i1 : INT := 0;
        i2 : INT := 0;
        END_VAR
    */

    short result = (short)client.InvokeRpcMethod("MAIN", "M_Add", new object[] { (short)1, (short)4 });


    // Call a Method that has no parameter and returns VOID
    client.InvokeRpcMethod("MAIN", "M_Method1", new object[] { });
}
}
}
}
}

```

Reference

[TwinCAT.Ads Namespace](#) [► 179]












Also see about this

-  [AdsClientExtensions.WhenNotification Method \(IAdsConnection, ISymbolCollection, NotificationSettings\)](#) [► 1266]

6.2.1.1 AdsClient Properties

The [AdsClient](#) [► 183] type exposes the following members.

Properties

	Name	Description
	Address [► 209]	Gets the target AmsAddress [► 752] of of the established ADS connection (Destination side).
	ClientAddress [► 209]	Get the client AmsAddress [► 752] (Source side).
	DefaultValueEncoding [► 209]	Gets the default value encoding.
	Id [► 210]	Gets the AdsClient [► 183] Identifier.
	IsConnected [► 210]	Gets a value indicating whether the local ADS port was opened successfully. It does not indicate if the target port is available. Use the method ReadState to determine if the target port is available.
	IsDisposed [► 211]	Gets a value indicating whether this instance is disposed.
	IsLocal [► 211]	Gets a value indicating whether the ADS client is connected to a ADS Server on the local computer.
	Logger [► 213]	Gets the logger interface.
	Session [► 212]	Gets the session that initiated this IConnection [► 79]
	SymbolEncoding [► 212]	Gets the symbol encoding.
	Timeout [► 213]	Sets the timeout for the ads communication. Unit is in ms.

Reference

[AdsClient Class](#) [► 183]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.1.1 **AdsClient.Address** Property

Gets the target [AmsAddress](#) [▶ 752] of of the established ADS connection (Destination side).

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsAddress Address { get; }
```

Property Value

Type: [AmsAddress](#) [▶ 752]

The address.

Implements

[IAdsConnection.Address](#) [▶ 895]

Reference

[AdsClient Class](#) [▶ 183]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.1.2 **AdsClient.ClientAddress** Property

Get the client [AmsAddress](#) [▶ 752] (Source side).

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsAddress ClientAddress { get; }
```

Property Value

Type: [AmsAddress](#) [▶ 752]

The client address.

Implements

[IAdsConnection.ClientAddress](#) [▶ 895]

Reference

[AdsClient Class](#) [▶ 183]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.1.3 **AdsClient.DefaultValueEncoding** Property

Gets the default value encoding.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Encoding DefaultValueEncoding { get; }
```

Property Value

Type: [Encoding](#)

The default value encoding.

Implements

[IConnection.DefaultValueEncoding](#) [► 81]

Reference

[AdsClient Class](#) [► 183]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.1.4 AdsClient.Id Property

Gets the [AdsClient](#) [► 183] Identifier.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Id { get; }
```

Property Value

Type: [Int32](#)

The identifier.

Implements

[IConnection.Id](#) [► 81]

Reference

[AdsClient Class](#) [► 183]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.1.5 AdsClient.IsConnected Property

Gets a value indicating whether the local ADS port was opened successfully. It does not indicate if the target port is available. Use the method `ReadState` to determine if the target port is available.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsConnected { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is connected; otherwise, false.

Implements

[IConnection.IsConnected](#) [[▶ 82](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.1.6 AdsClient.IsDisposed Property

Gets a value indicating whether this instance is disposed.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsDisposed { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is disposed; otherwise, false.

Implements

[IAdsDisposableConnection.IsDisposed](#) [[▶ 935](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.1.7 AdsClient.IsLocal Property

Gets a value indicating whether the ADS client is connected to a ADS Server on the local computer.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsLocal { get; }
```

Property ValueType: [Boolean](#)

true if this instance is local; otherwise, false.

Implements[IAdsConnection.IsLocal](#) [[▶ 896](#)]**Exceptions**

Exception	Condition
NotImplementedException	

Reference[AdsClient Class](#) [[▶ 183](#)][TwinCAT.Ads Namespace](#) [[▶ 179](#)]**6.2.1.1.8 AdsClient.Session Property**Gets the session that initiated this [IConnection](#) [[▶ 79](#)]**Namespace:** [TwinCAT.Ads](#) [[▶ 179](#)]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public ISession? Session { get; }
```

Property ValueType: [ISession](#) [[▶ 92](#)]

The session or NULL

Implements[IConnection.Session](#) [[▶ 82](#)]**Remarks**

The Session can be null on standalone connections.

Reference[AdsClient Class](#) [[▶ 183](#)][TwinCAT.Ads Namespace](#) [[▶ 179](#)]**6.2.1.1.9 AdsClient.SymbolEncoding Property**

Gets the symbol encoding.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Encoding SymbolEncoding { get; }
```

Property Value

Type: [Encoding](#)

The symbol encoding.

Reference

[AdsClient Class](#) [► 183]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.1.10 AdsClient.Timeout Property

Sets the timeout for the ads communication. Unit is in ms.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Timeout { get; set; }
```

Property Value

Type: [Int32](#)

Implements

[IConnection.Timeout](#) [► 83]

Reference

[AdsClient Class](#) [► 183]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.1.11 AdsClient.Logger Property

Gets the logger interface.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ILogger? Logger { get; }
```

Property Value

Type: [ILogger](#)
The logger.

Reference

[AdsClient Class](#) [[▶ 183](#)]














[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

















6.2.1.2 AdsClient Methods














The [AdsClient](#) [[▶ 183](#)] type exposes the following members.









Methods


















	Name	Description
	AddDeviceNotification(String, Int32, NotificationSettings, Object) [▶ 233]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 390] event.
	AddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object) [▶ 234]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event.
	AddDeviceNotificationAsync(String, Int32, NotificationSettings, Object, CancellationToken) [▶ 236]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 390] event.
	AddDeviceNotificationAsync(UInt32, UInt32, Int32, NotificationSettings, Object, CancellationToken) [▶ 237]	Adds a device notification as an asynchronous operation.
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type) [▶ 239]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 391] event.
	AddDeviceNotificationEx(String,	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 391] event.












	Name	Description
	NotificationSettings , Object , Type , .Int32) [▶ 240]	
	AddDeviceNotificationEx (UInt32 , UInt32 , NotificationSettings , Object , Type) [▶ 241]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 391] event.
	AddDeviceNotificationEx (UInt32 , UInt32 , NotificationSettings , Object , Type , .Int32) [▶ 242]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive ('AnyType') types are allowed for the parameter type.
	AddDeviceNotificationExAsync (String , NotificationSettings , Object , Type , .Int32 , CancellationToken) [▶ 243]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 391] event.
	AddDeviceNotificationExAsync (UInt32 , UInt32 , NotificationSettings , Object , Type , .Int32 , CancellationToken) [▶ 245]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 391] event.
	CleanupSymbolTable [▶ 246]	Clears the internal symbol cache.
	Close [▶ 247]	Closes this AdsClient [▶ 183]
	Connect (AmsAddress) [▶ 248]	Connects the target
	Connect (AmsPort) [▶ 249]	Connects to the local target ADS Device.
	Connect (Int32) [▶ 247]	Connects to the local target ADS Device.
	Connect (AmsNetId , Int32) [▶ 250]	Connects to the target ADS Device.
	Connect (AmsNetId , AmsPort) [▶ 250]	Connects to the target ADS Device.
	Connect (String , Int32) [▶ 249]	Connects to the target ADS Device.
	ConnectAndWaitAsync [▶ 251]	Connects to the target address and waits until the AdsClient [▶ 183] is disconnected asynchronously.














	Name	Description
	CreateVariableHandle [▶ 251]	Determines the Symbol handle by its instance path synchronously.
	CreateVariableHandleAsync [▶ 252]	Determines the Symbol handle by its instance path asynchronously.
	DeleteDeviceNotification [▶ 253]	Deletes a registered notification.
	DeleteDeviceNotificationAsync [▶ 254]	Deletes a registered notification asynchronously.
	DeleteVariableHandle [▶ 255]	Releases the specified symbol/variable handle synchronously.
	DeleteVariableHandleAsync [▶ 255]	Releases the specified symbol/variable handle asynchronously.
	Disconnect [▶ 256]	Disconnects this AdsClient [▶ 183] from the local ADS router.
	Dispose [▶ 257]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize [▶ 257]	Finalizes an instance of the AdsClient [▶ 183] class. (Overrides Object.Finalize .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	InvokeRpcMethodAsync(IRpcCallableInstance, IRpcMethod, .Object, CancellationTokens) [▶ 269]	invoke RPC method as an asynchronous operation.
 	InvokeRpcMethodAsync(String, String, .Object, CancellationTokens) [▶ 264]	Invokes the specified RPC Method asynchronously
	InvokeRpcMethodAsync(IRpcCallableInstance, IRpcMethod, .Object, .AnyTypeSpecifier, .AnyTypeSpecifier, CancellationTokens) [▶ 268]	invoke RPC method as an asynchronous operation.
















	Name	Description
 	InvokeRpcMethodAsync(String, String, Object, AnyTypeSpecifier, AnyTypeSpecifier, CancellationToken) [▶ 266]	Invokes the specified RPC Method asynchronously
	Read(UInt32, Memory.Byte.) [▶ 271]	Reads data synchronously from an ADS device and writes to the specified readBuffer.
	Read(UInt32, UInt32, Memory.Byte.) [▶ 272]	Reads data synchronously from an ADS device and writes it to the given readBuffer
	ReadAny(UInt32, Type) [▶ 275]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, Type, Int32.) [▶ 276]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type) [▶ 278]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type, Int32.) [▶ 279]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32) [▶ 273]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, Int32.) [▶ 274]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, UInt32) [▶ 275]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, UInt32, Int32.) [▶ 277]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, Type, CancellationToken) [▶ 283]	Reads the value of an Anytype (Primitive type) asynchronously.
	ReadAnyAsync(UInt32, Type, Int32., CancellationToken) [▶ 285]	Reads the value of an Anytype (Primitive type) asynchronously.
	ReadAnyAsync(UInt32, UInt32, Type, CancellationToken) [▶ 287]	Reads the value of an Anytype (Primitive type) asynchronously.
















	Name	Description
	ReadAnyAsync(UInt32, UInt32, Type, .Int32., CancellationToken) [▶ 287]	Reads the value of an Anytype (Primitive type) asynchronously.
	ReadAnyAsync.T.(UInt32, CancellationToken) [▶ 281]	Reads data synchronously from an ADS device.
	ReadAnyAsync.T.(UInt32, .Int32., CancellationToken) [▶ 282]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T.(UInt32, UInt32, CancellationToken) [▶ 284]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T.(UInt32, UInt32, .Int32., CancellationToken) [▶ 285]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyString(UInt32, Int32, Encoding) [▶ 289]	Reads a string from the specified symbol/variable.
	ReadAnyString(UInt32, UInt32, Int32, Encoding) [▶ 289]	Reads as string from a specified address.
	ReadAnyStringAsync(UInt32, Int32, Encoding, CancellationToken) [▶ 291]	Reads a string asynchronously from the specified symbol/variable
	ReadAnyStringAsync(UInt32, UInt32, Int32, Encoding, CancellationToken) [▶ 292]	Reads a string from a specified address asynchronously.
	ReadAsync(UInt32, Memory.Byte, CancellationToken) [▶ 294]	Reads the value data of the symbol asynchronously into the readBuffer.




	Name	Description
	ReadAsync(UInt32, UInt32, Memory.Byte, CancellationToken) [▶ 295]	Reads the data asynchronously from specified IndexGroup/IndexOffset
	ReadDataType [▶ 296]	Call this method to obtain information about the specified data type.
	ReadDataTypeAsync [▶ 297]	read data type as an asynchronous operation.
	ReadDeviceInfo [▶ 298]	Reads the identification and version number of an ADS server.
	ReadDeviceInfoAsync [▶ 298]	Reads the identification and version number of an ADS server.
	ReadState [▶ 299]	Reads the ADS status and the device status from an ADS server.
	ReadStateAsync [▶ 299]	Reads the ADS status and the device status from an ADS server.
	ReadSymbol [▶ 300]	Call this method to obtain information about the individual symbols (variables) in ADS devices.
	ReadSymbolAsync [▶ 301]	Call this method to obtain information about the individual symbols (variables) in ADS devices.
	ReadValue(ISymbol) [▶ 303]	Reads the value of a symbol and returns it as an object.
	ReadValue(String, Type) [▶ 304]	Reads the value of a symbol and returns the value as object. The parameter type must have the same layout as the ADS symbol.
	ReadValue.T.(ISymbol) [▶ 303]	Reads the value of a symbol and returns it as an object.
	ReadValue.T.(String) [▶ 302]	Reads the value of a symbol and returns the value. The parameter type must have the same layout as the ADS symbol.
	ReadValueAsync(ISymbol, CancellationToken) [▶ 306]	Reads the value of a symbol asynchronously and returns it as an object. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. Arrays and structures cannot be read.
	ReadValueAsync(String, Type, CancellationToken) [▶ 308]	Reads the value of a symbol asynchronously.
	ReadValueAsync.T.Value.(ISymbol, CancellationToken) [▶ 307]	Reads the value of a symbol asynchronously and returns it as an object.
	ReadValueAsync.T.Value.(String, CancellationToken) [▶ 306]	Reads the value of a symbol asynchronously.












	Name	Description
	ReadWrite(UInt32, Memory.Byte., ReadOnlyMemory.Byte.) [▶ 310]	Writes data synchronously to an ADS device and then Reads data from that target.
	ReadWrite(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte.) [▶ 311]	Writes data synchronously to an ADS device and then Reads data from this device into the readBuffer
	ReadWriteAsync(UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken) [▶ 314]	ReadWrites value data asynchronously to/from the symbol represented by the variableHandle.
	ReadWriteAsync(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken) [▶ 315]	Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer
	RegisterAdsStateChangedAsync [▶ 316]	Registers for AdsStateChanged [▶ 392] events as an asynchronous operation.
	RegisterSymbolVersionChanged [▶ 387]	Registers for an AdsSymbolVersionChanged [▶ 394] event as an asynchronous operation.
	RegisterSymbolVersionChangedAsync [▶ 317]	Registers for an AdsSymbolVersionChanged [▶ 394] event as an asynchronous operation.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryAddDeviceNotification(String, Int32, NotificationSettings, Object, UInt32.) [▶ 318]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 390] event.
	TryAddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.) [▶ 319]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event.
	TryAddDeviceNotificationEx(String,	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 391] event.

	Name	Description
	NotificationSettings, Object, Type, .Int32., UInt32.) [▶ 321]	
	TryAddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., UInt32.) [▶ 322]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event.
	TryCreateVariableHandle [▶ 323]	Determines the Symbol handle by its instance path synchronously.
	TryDeleteDeviceNotification(UInt32) [▶ 325]	Deletes a registered notification.
	TryDeleteDeviceNotification(UInt32, Int32) [▶ 325]	Deletes a registered notification.
	TryDeleteVariableHandle [▶ 326]	Releases the specified symbol/variable handle synchronously.
 	TryInvokeRpcMethod(String, String, .Object., Object.) [▶ 328]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(String, String, .Object., Object., Object.) [▶ 330]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(IRpcCallableInstance, IRpcMethod, .Object., .AnyTypeSpecifier, .AnyTypeSpecifier, .Object., Object.) [▶ 333]	Tries the invoke RPC method.
 	TryInvokeRpcMethod(String, String, .Object., .AnyTypeSpecifier, .AnyTypeSpecifier, .Object., Object.) [▶ 331]	Invokes the RPC Method


	Name	Description
	TryRead(UInt32, Memory.Byte., Int32.) [▶ 337]	Reads the value synchronously data of the symbol, that is represented by the variable handle into the readBuffer.
	TryRead(UInt32, UInt32, Memory.Byte., Int32.) [▶ 338]	Reads value data from the specified IndexGroup/IndexOffset to the specified memory location.
	TryReadDataType [▶ 338]	Call this method to obtain information about the specified data type.
	TryReadState [▶ 339]	Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful.
	TryReadSymbol [▶ 340]	Call this method to obtain information about the individual symbols (variables) in ADS devices.
	TryReadValue(ISymbol, Object.) [▶ 342]	Reads the value of a symbol and returns it as an object.
	TryReadValue(String, Type, Object.) [▶ 344]	Reads the value of a symbol and returns the value as object.
	TryReadValue.T. (ISymbol, T.) [▶ 343]	Reads the value of a symbol and returns it as an object.
	TryReadValue.T. (String, T.) [▶ 341]	Reads the value of a symbol and returns the value as object.
	TryReadWrite(UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.) [▶ 346]	ReadWrites value data synchronously to/from the symbol represented by the variableHandle.
	TryReadWrite(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.) [▶ 347]	Writes data synchronously to an ADS device and reads data from that device.
	TryResurrect [▶ 348]	Resurrects the connection
	TryWrite(UInt32, ReadOnlyMemory.Byte.) [▶ 350]	Writes the value data synchronously that is represented in the writeBuffer to the symbol with the specified variableHandle.
	TryWrite(UInt32, UInt32, ReadOnlyMemory.Byte.) [▶ 351]	Writes data synchronously to an ADS device.
	TryWriteControl(StationInfo) [▶ 352]	Changes the ADS status and the device status of an ADS server.












	Name	Description
	TryWriteControl(Sta telInfo, ReadOnlyMemory.B yte.) [▶ 353]	Changes the ADS status and the device status of an ADS server.
	TryWriteValue(ISym bol, Object) [▶ 356]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	TryWriteValue(Strin g, Object) [▶ 354]	Writes the passed object value to the specified ADS symbol.The parameter type must have the same layout as the ADS symbol.
	TryWriteValue.T. (ISymbol, T) [▶ 356]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	TryWriteValue.T. (String, T) [▶ 355]	Writes the passed object value to the specified ADS symbol.The parameter type must have the same layout as the ADS symbol.
	UnregisterAdsState ChangedAsync [▶ 357]	unregister ads state changed as an asynchronous operation.
	UnregisterSymbolV ersionChanged [▶ 388]	Unregisters the symbol version changed.
	UnregisterSymbolV ersionChangedAsyn c [▶ 358]	Unregisters from an AdsSymbolVersionChanged [▶ 394] event as an asynchronous operation.
	Write(UInt32, ReadOnlyMemory.B yte.) [▶ 361]	Writes data synchronously to an ADS device.
	Write(UInt32, UInt32) [▶ 360]	Triggers a 'Write' call to the ADS device at the specified address.
	Write(UInt32, UInt32, ReadOnlyMemory.B yte.) [▶ 361]	Writes data synchronously to an ADS device.
	WriteAny(UInt32, Object) [▶ 362]	Writes an object synchronously to an ADS device.
	WriteAny(UInt32, Object, .Int32.) [▶ 363]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAny(UInt32, UInt32, Object) [▶ 364]	Writes an object synchronously to an ADS device.
	WriteAny(UInt32, UInt32, Object, .Int32.) [▶ 364]	Writes an object synchronously to an ADS device.

	Name	Description
	WriteAnyAsync(UInt32, Object, CancellationToken) [▶ 366]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAnyAsync(UInt32, Object, .Int32, CancellationToken) [▶ 367]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAnyAsync(UInt32, UInt32, Object, CancellationToken) [▶ 368]	Write the value of an Anytype (Primitive type) asynchronously.
	WriteAnyAsync(UInt32, UInt32, Object, .Int32, CancellationToken) [▶ 369]	Write the value of an Anytype (Primitive type) asynchronously.
	WriteAnyStringAsync(String, String, Int32, Encoding, CancellationToken) [▶ 370]	Writes the string (Potentially unsafe!)
	WriteAnyStringAsync(UInt32, String, Int32, Encoding, CancellationToken) [▶ 371]	Writes the string (Potentially unsafe!)
	WriteAsync(UInt32, ReadOnlyMemory.Byte, CancellationToken) [▶ 375]	Writes the value data asynchronously that is represented by the writeBuffer to the symbol specified by the variableHandle.
	WriteAsync(UInt32, UInt32, CancellationToken) [▶ 373]	Triggers a write call at the specified IndexGroup/IndexOffset asynchronously.
	WriteAsync(UInt32, UInt32, ReadOnlyMemory.Byte, CancellationToken) [▶ 375]	Writes the data / Value asynchronously into the specified writeBuffer.
	WriteControl(StateInfo) [▶ 376]	Changes the ADS status and the device status of an ADS server.










	Name	Description
	WriteControl(StateInfo, ReadOnlyMemory<Byte>) [▶ 377]	Changes the ADS status and the device status of an ADS server.
	WriteControlAsync(AdsState, UInt16, CancellationToken) [▶ 379]	Changes the ADS status and device status of the ADS server asynchronously.
	WriteControlAsync(AdsState, UInt16, ReadOnlyMemory<Byte>, CancellationToken) [▶ 380]	Writes the state asynchronously
	WriteSymbolAsync [▶ 381]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	WriteValue(ISymbol, Object) [▶ 383]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValue(String, Object) [▶ 382]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	WriteValue.T.(ISymbol, T) [▶ 384]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValue.T.(String, T) [▶ 382]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	WriteValueAsync(ISymbol, Object, CancellationToken) [▶ 386]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValueAsync.T.(ISymbol, T, CancellationToken) [▶ 387]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValueAsync.T.(String, T, CancellationToken) [▶ 385]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Extension Methods






	Name	Description
	PollAdsState(IObservable<Unit>) [▶ 3294]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
















	Name	Description
	PollAdsState(IObservable.Unit.) [▸ 1256]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
	PollAdsState(TimeSpan) [▸ 3295]	Overloaded. Gets an observable sequence of ResultReadAdsState [▸ 1146]s via Polling. (Defined by ConnectionStateExtension [▸ 3292].)
	PollAdsState(TimeSpan) [▸ 1257]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
	PollAdsState2(IObservable.Unit.) [▸ 1271]	Overloaded. Gets an observable sequence of ResultReadAdsState [▸ 1146]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
	PollAdsState2(TimeSpan) [▸ 1272]	Overloaded. Gets an observable sequence of ResultReadAdsState [▸ 1146]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
	PollAdsState2Async(IObservable.Unit., Cancellation.Token) [▸ 1274]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
	PollAdsState2Async(TimeSpan, Cancellation.Token) [▸ 1275]	Overloaded. Gets an observable sequence of ResultReadAdsState [▸ 1146]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
	PollAdsStateAsync(IObservable.Unit., Cancellation.Token) [▸ 3296]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by ConnectionStateExtension [▸ 3292].)
	PollAdsStateAsync(IObservable.Unit., Cancellation.Token) [▸ 1259]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
	PollAdsStateAsync(TimeSpan, Cancellation.Token) [▸ 3298]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by ConnectionStateExtension [▸ 3292].)
	PollAdsStateAsync(TimeSpan, Cancellation.Token) [▸ 1260]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
	PollDeviceState(IObservable.Unit.) [▸ 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▸ 1157]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)








	Name	Description
	PollDeviceState(TimeSpan) [▶ 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157] s via Polling. (Defined by AdsClientExtensions [▶ 1249] .)
	PollDeviceStateAsync(IObservable.Unit, Cancellation.Token) [▶ 1278]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157] s via Polling. (Defined by AdsClientExtensions [▶ 1249] .)
	PollDeviceStateAsync(TimeSpan, Cancellation.Token) [▶ 1279]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157] s via Polling. (Defined by AdsClientExtensions [▶ 1249] .)
	PollSystemServiceState [▶ 3282]	Polls the state of the system service. (Defined by SystemServiceExtension [▶ 3281] .)
	PollSystemServiceStateAsync [▶ 3283]	Polls the system service state asynchronously (Defined by SystemServiceExtension [▶ 3281] .)
	PollValues(ISymbol, Type, IObservable.Unit) [▶ 1310]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(ISymbol, Type, TimeSpan) [▶ 1311]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(String, Type, IObservable.Unit) [▶ 1300]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(String, Type, TimeSpan) [▶ 1301]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(ISymbol, Type, .Int32., TimeSpan) [▶ 1316]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(ISymbol, Type, IObservable.Unit, Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(ISymbol, Type, TimeSpan, Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(String, Type, .Int32., TimeSpan) [▶ 1304]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)

	Name	Description
	PollValues(String, Type, IObservable.Unit, Func.Exception, Object.) [▶ 1305]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, TimeSpan, Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32, IObservable.Unit, Func.Exception, Object.) [▶ 1321]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32, TimeSpan, Func.Exception, Object.) [▶ 1322]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32, IObservable.Unit, Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32, TimeSpan, Func.Exception, Object.) [▶ 1307]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, IObservable.Unit.) [▶ 1308]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, TimeSpan) [▶ 1309]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, IObservable.Unit.) [▶ 1294]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, TimeSpan) [▶ 1295]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, IObservable.Unit, Func.Exception, T.) [▶ 1314]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	<code>PollValues.T.</code> (<code>ISymbol</code> , <code>TimeSpan</code> , <code>Func.Exception</code> , <code>T</code>) [▶ 1315]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	<code>PollValues.T.</code> (<code>ISymbol</code> , <code>.Int32</code> , <code>IObservable.Unit</code>) [▶ 1312]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	<code>PollValues.T.</code> (<code>ISymbol</code> , <code>.Int32</code> , <code>TimeSpan</code>) [▶ 1313]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	<code>PollValues.T.(String</code> , <code>IObservable.Unit</code> , <code>Func.Exception</code> , <code>T</code>) [▶ 1298]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	<code>PollValues.T.(String</code> , <code>TimeSpan</code> , <code>Func.Exception</code> , <code>T</code>) [▶ 1299]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	<code>PollValues.T.</code> (<code>String</code> , <code>.Int32</code> , <code>IObservable.Unit</code>) [▶ 1296]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	<code>PollValues.T.</code> (<code>String</code> , <code>.Int32</code> , <code>TimeSpan</code>) [▶ 1297]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	<code>PollValues.T.</code> (<code>ISymbol</code> , <code>.Int32</code> , <code>IObservable.Unit</code> , <code>Func.Exception</code> , <code>T</code>) [▶ 1318]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	<code>PollValues.T.</code> (<code>ISymbol</code> , <code>.Int32</code> , <code>TimeSpan</code> , <code>Func.Exception</code> , <code>T</code>) [▶ 1319]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	<code>PollValues.T.</code> (<code>String</code> , <code>.Int32</code> , <code>IObservable.Unit</code> , <code>Func.Exception</code> , <code>T</code>) [▶ 1302]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	<code>PollValues.T.</code> (<code>String</code> , <code>.Int32</code> , <code>TimeSpan</code> , <code>Func.Exception</code> , <code>T</code>) [▶ 1303]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	PollValues2.T. (ISymbol , .Int32 , IObservable.Unit .) [▶ 1330]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues2.T. (String , .Int32 , IObservable.Unit .) [▶ 1329]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	ReadSysServState [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (Defined by SystemServiceExtension [▶ 3281].)
	ReadSysServStateAs ync [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (asynchronous) (Defined by SystemServiceExtension [▶ 3281].)
	ReadWithFallback(U Int32, UInt32, Memory.Byte, UInt32, Boolean) [▶ 1219]	Overloaded. Ads Read with fallback. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallback(U Int32, UInt32, UInt32, Memory.Byte, Func.ResultRead, Boolean, Boolean) [▶ 1220]	Overloaded. Ads Read with Fallback. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallbackAs ync(UInt32, UInt32, UInt32, Memory.Byte, CancellationToken) [▶ 1221]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallbackAs ync(UInt32, UInt32, UInt32, Memory.Byte, Func.ResultRead, Boolean, CancellationToken) [▶ 1222]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedRead(UInt 32, UInt32, Memory.Byte, Int32, TimeSpan) [▶ 1224]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedRead(UInt 32, UInt32, Memory.Byte, Int32,	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [▶ 1215].)

	Name	Description
	TimeSpan , Func.ResultRead , Boolean) [▶ 1225]	
	RepeatedReadAsync (UInt32 , UInt32 , Memory.Byte , Int32 , TimeSpan , CancellationToken) [▶ 1226]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedReadAsync (UInt32 , UInt32 , Memory.Byte , Int32 , TimeSpan , Func.ResultRead , Boolean , CancellationToken) [▶ 1227]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	RestartTwinCATAsync [▶ 3285]	Sends a SystemService Restart to the target system and waits until the system has finished the state changes (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [▶ 3281].)
	SetAdsState [▶ 3299]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [▶ 3292].)
	SetAdsStateAsync [▶ 3300]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [▶ 3292].)
	WaitUntilRestarted [▶ 3286]	Waits until the Restart is detected on the client (SystemService, Port 10000) (Defined by SystemServiceExtension [▶ 3281].)
	WaitUntilRestartedAsync [▶ 3287]	Waits until the SystemService Restart is detected on the client (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [▶ 3281].)
 	WhenAdsStateChanges [▶ 1261]	Gets an observable sequence of AdsState [▶ 729]s. (Defined by AdsClientExtensions [▶ 1249].)
	WhenNotification(ISymbol) [▶ 1263]	Overloaded. Gets an observable sequence of Notification [▶ 1100]s. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenNotification(ISymbolCollection) [▶ 1264]	Overloaded. Gets an observable sequence of Notification [▶ 1100] objects. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenNotification(IList.ISymbol, NotificationSettings) [▶ 1268]	Overloaded. Gets an observable sequence of Notification [▶ 1100] objects. (Defined by AdsClientExtensions [▶ 1249].)
	WhenNotification(ISymbol, NotificationSettings) [▶ 1265]	Overloaded. Gets an observable sequence of SymbolValueNotification [▶ 1331]s. (Defined by AdsClientExtensions [▶ 1249].)


	Name	Description
	WhenNotification(String, Type, NotificationSettings) [▶ 1324]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [▶ 1281].)
	WhenNotification.T.(String, NotificationSettings) [▶ 1323]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [▶ 1281].)
	WhenSymbolVersionChanges. [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
	WhenSymbolVersionChanges(IScheduler) [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
	WhenValueChanged [▶ 1345]	Observable sequence of Values driven by ADS Notifications on the specified symbol. (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues.T.(String, IObservable.T.) [▶ 1326]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)
	WriteValues.T.(String, IObservable.T., Action.Exception.) [▶ 1327]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)

Reference

[AdsClient Class](#) [▶ 183]



[TwinCAT.Ads Namespace](#) [▶ 179]

Also see about this

-  [AdsClientExtensions.WhenNotification Method \(IAdsConnection, ISymbolCollection, NotificationSettings\)](#) [▶ 1266]

6.2.1.2.1 AdsClient.AddDeviceNotification Method

Overload List

	Name	Description
	AddDeviceNotification(String, Int32, NotificationSettings, Object) [▶ 233]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 390] event.
	AddDeviceNotification(UInt32, UInt32,	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event.

	Name	Description
	Int32, NotificationSettings, Object) [▶ 234]	

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.1.1 AdsClient.AddDeviceNotification Method (String, Int32, NotificationSettings, Object)

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotification](#) [[▶ 390](#)] event.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint AddDeviceNotification(
    string symbolPath,
    int dataSize,
    NotificationSettings settings,
    Object? userData
)
```

Parameters

symbolPath	Type: System.String Symbol / Instance path of the ADS variable.
dataSize	Type: System.Int32 Maximum amount of data in bytes to receive with this ADS Notification.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)

Return Value

Type: [UInt32](#)
The notification handle.

Implements

[IAdsNotifications.AddDeviceNotification\(String, Int32, NotificationSettings, Object\)](#) [[▶ 971](#)]

Remarks

The

dataSize

Parameter defines the amount of bytes, that will be attached to the [AdsNotification](#) [[▶ 390](#)] as value. Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\)](#) [[▶ 253](#)] should always called when the notification is not used anymore.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[AddDeviceNotification Overload](#) [[▶ 232](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[AdsClient.AdsNotification](#) [[▶ 390](#)]

[AdsClient.DeleteDeviceNotification\(UInt32\)](#) [[▶ 253](#)]

[AddDeviceNotification Overload](#) [[▶ 971](#)]

[AddDeviceNotificationAsync Overload](#) [[▶ 973](#)]

[TryAddDeviceNotification Overload](#) [[▶ 985](#)]

6.2.1.2.1.2 AdsClient.AddDeviceNotification Method (UInt32, UInt32, Int32, NotificationSettings, Object)

Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint AddDeviceNotification(
    uint indexGroup,
    uint indexOffset,
    int dataSize,
    NotificationSettings settings,
    Object? userData
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
dataSize	Type: System.Int32 Maximum amount of data in bytes to receive with this ADS Notification.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)

Return Value

Type: [UInt32](#)

The notification handle.

Implements

[IAdsNotifications.AddDeviceNotification\(UInt32, UInt32, Int32, NotificationSettings, Object\)](#) [[▶ 972](#)]

Remarks

The
dataSize



Parameter defines the amount of bytes, that will be attached to the [AdsNotification](#) [[▶ 390](#)] as value. Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\)](#) [[▶ 253](#)] should always called when the notification is not used anymore.

Reference

- [AdsClient Class](#) [[▶ 183](#)]
- [AddDeviceNotification Overload](#) [[▶ 232](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]
- [AdsClient.DeleteDeviceNotification\(UInt32\)](#) [[▶ 253](#)]
- [AdsClient.AdsNotification](#) [[▶ 390](#)]
- [AdsClient.AdsNotificationError](#) [[▶ 391](#)]
- [AddDeviceNotification Overload](#) [[▶ 971](#)]
- [TryAddDeviceNotification Overload](#) [[▶ 985](#)]
- [AddDeviceNotificationAsync Overload](#) [[▶ 973](#)]

6.2.1.2.2 AdsClient.AddDeviceNotificationAsync Method

Overload List

	Name	Description
	AddDeviceNotificationAsync(String, Int32, NotificationSettings, Object, CancellationTok en) [▶ 236]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 390] event.
	AddDeviceNotificationAsync(UInt32, UInt32, Int32, NotificationSettings, Object, CancellationTok en) [▶ 237]	Adds a device notification as an asynchronous operation.

Reference

- [AdsClient Class](#) [[▶ 183](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.2.1 **AdsClient.AddDeviceNotificationAsync Method (String, Int32, NotificationSettings, Object, CancellationToken)**

Connects a variable to the ADS client asynchronously. The ADS client will be notified by the [AdsNotification \[▸ 390\]](#) event.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultHandle> AddDeviceNotificationAsync (
    string symbolPath,
    int dataSize,
    NotificationSettings settings,
    Object? userData,
    CancellationToken cancel
)
```

Parameters

symbolPath	Type: System.String The symbol/instance path of the ADS variable.
dataSize	Type: System.Int32 Maximum amount of data in bytes to receive with this ADS Notification.
settings	Type: TwinCAT.Ads.NotificationSettings [▸ 1106] The notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
cancel	Type: System.Threading.CancellationToken The Cancellation token.

Return Value

Type: [Task.ResultHandle \[▸ 1136\]](#).

A task that represents the asynchronous 'AddDeviceNotification' operation. The [ResultHandle \[▸ 1136\]](#) type parameter contains the created handle ([Handle \[▸ 1138\]](#)) and the [ErrorCode \[▸ 1120\]](#) after execution.

Implements

[IAdsNotifications.AddDeviceNotificationAsync\(String, Int32, NotificationSettings, Object, CancellationToken\) \[▸ 974\]](#)

Remarks

The

dataSize

Parameter defines the amount of bytes, that will be attached to the [AdsNotification \[▸ 390\]](#) as value. Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotificationAsync\(UInt32, CancellationToken\) \[▸ 254\]](#) should always be called when the notification is not used anymore.

Reference

[AdsClient Class \[▸ 183\]](#)

[AddDeviceNotificationAsync Overload \[▸ 235\]](#)

[TwinCAT.Ads Namespace](#) [► 179]

[AdsClient.AdsNotification](#) [► 390]

[AdsClient.DeleteDeviceNotificationAsync\(UInt32, CancellationTokens\)](#) [► 254]

[AddDeviceNotification Overload](#) [► 971]

[AddDeviceNotificationAsync Overload](#) [► 973]

[TryAddDeviceNotification Overload](#) [► 985]

6.2.1.2.2 AdsClient.AddDeviceNotificationAsync Method (UInt32, UInt32, Int32, NotificationSettings, Object, CancellationTokens)

Adds a device notification as an asynchronous operation.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultHandle> AddDeviceNotificationAsync (
    uint indexGroup,
    uint indexOffset,
    int dataSize,
    NotificationSettings settings,
    Object? userData,
    CancellationTokens cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group number of the requested ADS service.
indexOffset	Type: System.UInt32 The index offset number of the requested ADS service.
dataSize	Type: System.Int32 Maximum amount of data in bytes to receive with this ADS Notification.
settings	Type: TwinCAT.Ads.NotificationSettings [► 1106] The notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
cancel	Type: System.Threading.CancellationTokens The Cancellation token.

Return Value

Type: [Task.ResultHandle](#) [► 1136].

A task that represents the asynchronous 'AddDeviceNotification' operation. The [ResultHandle](#) [► 1136] type parameter contains the created handle (Handle) and the [ErrorCode](#) [► 1120] after execution.

Implements

[IAdsNotifications.AddDeviceNotificationAsync\(UInt32, UInt32, Int32, NotificationSettings, Object, CancellationTokens\)](#) [► 975]

Remarks

The
dataSize

Parameter defines the amount of bytes, that will be attached to the [AdsNotification](#) [► 390] as value. Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [► 984] should always be called when the notification is not used anymore.

Reference

[AdsClient Class](#) [► 183]

[AddDeviceNotificationAsync Overload](#) [► 235]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsNotifications.AdsNotification](#) [► 991]





[IAdsNotifications.DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [► 984]

[AddDeviceNotification Overload](#) [► 971]

[AddDeviceNotificationAsync Overload](#) [► 973]

[TryAddDeviceNotification Overload](#) [► 985]

6.2.1.2.3 AdsClient.AddDeviceNotificationEx Method**Overload List**

	Name	Description
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type) [► 239]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [► 391] event.
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32.) [► 240]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [► 391] event.
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type) [► 241]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [► 391] event.
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32.) [► 242]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive ('AnyType') types are allowed for the parameter type.

Reference

[AdsClient Class](#) [► 183]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.2.3.1 AdsClient.AddDeviceNotificationEx Method (String, NotificationSettings, Object, Type)

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotificationEx](#) [► 391] event.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint AddDeviceNotificationEx(
    string symbolPath,
    NotificationSettings settings,
    Object? userData,
    Type type
)
```

Parameters

symbolPath	Type: System.String Symbol/Instance path of the ADS variable.
settings	Type: TwinCAT.Ads.NotificationSettings [► 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
type	Type: System.Type Type of the object stored in the event argument ('AnyType')

Return Value

Type: [UInt32](#)
The notification handle.

Implements

[IAdsNotifications.AddDeviceNotificationEx\(String, NotificationSettings, Object, Type\)](#) [► 977]

Remarks

Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\)](#) [► 253] should always called when the notification is not used anymore.

Reference

[AdsClient Class](#) [► 183]

[AddDeviceNotificationEx Overload](#) [► 238]

[TwinCAT.Ads Namespace](#) [► 179]

[AdsClient.AdsNotificationEx](#) [► 391]

[AdsClient.DeleteDeviceNotification\(UInt32\)](#) [► 253]

[AddDeviceNotificationEx Overload \[► 976\]](#)

[AddDeviceNotificationExAsync Overload \[► 981\]](#)

[TryAddDeviceNotificationEx Overload \[► 988\]](#)

6.2.1.2.3.2 **AdsClient.AddDeviceNotificationEx Method (String, NotificationSettings, Object, Type, .Int32.)**

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotificationEx \[► 391\]](#) event.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint AddDeviceNotificationEx(
    string symbolPath,
    NotificationSettings settings,
    Object? userData,
    Type type,
    int[]? args
)
```

Parameters

symbolPath	Type: System.String Symbol/Instance path of the ADS variable.
settings	Type: TwinCAT.Ads.NotificationSettings [► 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
type	Type: System.Type Type of the object stored in the event argument ('AnyType')
args	Type: .System.Int32 . Additional arguments (for 'AnyType')

Return Value

Type: [UInt32](#)
The notification handle.

Implements

[IAdsNotifications.AddDeviceNotificationEx\(String, NotificationSettings, Object, Type, .Int32.\) \[► 977\]](#)

Remarks

Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\) \[► 253\]](#) should always called when the notification is not used anymore.

Reference

[AdsClient Class \[► 183\]](#)

[AddDeviceNotificationEx Overload \[► 238\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

[AdsClient.AdsNotificationEx](#) [► 391]

[AdsClient.DeleteDeviceNotification\(UInt32\)](#) [► 253]

[AddDeviceNotificationEx Overload](#) [► 976]

[AddDeviceNotificationExAsync Overload](#) [► 981]

[TryAddDeviceNotificationEx Overload](#) [► 988]

6.2.1.2.3.3 **AdsClient.AddDeviceNotificationEx Method (UInt32, UInt32, NotificationSettings, Object, Type)**

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotificationEx](#) [► 391] event.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint AddDeviceNotificationEx(
    uint indexGroup,
    uint indexOffset,
    NotificationSettings settings,
    Object? userData,
    Type type
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
settings	Type: TwinCAT.Ads.NotificationSettings [► 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
type	Type: System.Type Type of the object stored in the event argument ('AnyType')

Return Value

Type: [UInt32](#)

The notification handle.

Implements

[IAdsNotifications.AddDeviceNotificationEx\(UInt32, UInt32, NotificationSettings, Object, Type\)](#) [► 978]

Remarks

Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\)](#) [► 253] should always called when the notification is not used anymore.

Reference

[AdsClient Class](#) [► 183]

[AddDeviceNotificationEx Overload \[▶ 238\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

[AdsClient.DeleteDeviceNotification\(UInt32\) \[▶ 253\]](#)

[AdsClient.AdsNotificationEx \[▶ 391\]](#)

[AdsClient.AdsNotificationError \[▶ 391\]](#)

[AddDeviceNotificationEx Overload \[▶ 976\]](#)

[TryAddDeviceNotificationEx Overload \[▶ 988\]](#)

[AddDeviceNotificationExAsync Overload \[▶ 981\]](#)

6.2.1.2.3.4 **AdsClient.AddDeviceNotificationEx Method (UInt32, UInt32, NotificationSettings, Object, Type, .Int32.)**

Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive ('AnyType') types are allowed for the parameter type.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint AddDeviceNotificationEx(
    uint indexGroup,
    uint indexOffset,
    NotificationSettings settings,
    Object? userData,
    Type type,
    int[]? args
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data.
type	Type: System.Type Type of the object stored in the event argument.
args	Type: .System.Int32 . Additional arguments for 'AnyType' types.

Return Value

Type: [UInt32](#)
The notification handle.

Implements

[IAdsNotifications.AddDeviceNotificationEx\(UInt32, UInt32, NotificationSettings, Object, Type, .Int32.\) \[▸ 980\]](#)

Remarks

Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\) \[▸ 253\]](#) should always called when the notification is not used anymore.

Reference

[AdsClient Class \[▸ 183\]](#)

[AddDeviceNotificationEx Overload \[▸ 238\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

[AdsClient.AdsNotificationEx \[▸ 391\]](#)



[AddDeviceNotificationEx Overload \[▸ 976\]](#)

[AddDeviceNotificationExAsync Overload \[▸ 981\]](#)

[TryAddDeviceNotificationEx Overload \[▸ 988\]](#)

6.2.1.2.4 AdsClient.AddDeviceNotificationExAsync Method

Overload List

	Name	Description
	AddDeviceNotificationExAsync(String, NotificationSettings, Object, Type, .Int32., CancellationToken) [▸ 243]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▸ 391] event.
	AddDeviceNotificationExAsync(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., CancellationToken) [▸ 245]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▸ 391] event.

Reference

[AdsClient Class \[▸ 183\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.1.2.4.1 AdsClient.AddDeviceNotificationExAsync Method (String, NotificationSettings, Object, Type, .Int32., CancellationToken)

Connects a variable to the ADS client asynchronously. The ADS client will be notified by the [AdsNotificationEx \[▸ 391\]](#) event.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultHandle> AddDeviceNotificationExAsync(
    string symbolPath,
    NotificationSettings settings,
    Object? userData,
    Type type,
    int[]? args,
    CancellationToken cancel
)
```

Parameters

symbolPath	Type: System.String The symbol/instance path of the ADS variable.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
type	Type: System.Type Type of the object stored in the event argument ('AnyType')
args	Type: .System.Int32 . Additional arguments (for 'AnyType')
cancel	Type: System.Threading.CancellationToken The Cancellation token.

Return Value

Type: [Task.ResultHandle](#) [[▶ 1136](#)].

A task that represents the asynchronous 'AddDeviceNotification' operation. The [ResultHandle](#) [[▶ 1136](#)] type parameter contains the created handle ([Handle](#) [[▶ 1138](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsNotifications.AddDeviceNotificationExAsync\(String, NotificationSettings, Object, Type, .Int32., CancellationToken\)](#) [[▶ 981](#)]

Remarks

Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [[▶ 254](#)] should always be called when the notification is not used anymore.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[AddDeviceNotificationExAsync Overload](#) [[▶ 243](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[AdsClient.AdsNotificationEx](#) [[▶ 391](#)]

[AdsClient.DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [[▶ 254](#)]

[AddDeviceNotificationEx Overload](#) [[▶ 976](#)]

[AddDeviceNotificationExAsync Overload \[► 981\]](#)

[TryAddDeviceNotificationEx Overload \[► 988\]](#)

6.2.1.2.4.2 **AdsClient.AddDeviceNotificationExAsync Method (UInt32, UInt32, NotificationSettings, Object, Type, .Int32., CancellationToken)**

Connects a variable to the ADS client asynchronously. The ADS client will be notified by the [AdsNotificationEx \[► 391\]](#) event.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultHandle> AddDeviceNotificationExAsync (
    uint indexGroup,
    uint indexOffset,
    NotificationSettings settings,
    Object? userData,
    Type anyType,
    int[]? args,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
settings	Type: TwinCAT.Ads.NotificationSettings [► 1106] The settings.
userData	Type: System.Object This object can be used to store user specific data.
anyType	Type: System.Type Type of the object stored in the event argument, only Primitive 'AnyTypes' allowed.
args	Type: .System.Int32. Additional arguments (for 'AnyType')
cancel	Type: System.Threading.CancellationToken The Cancellation token.

Return Value

Type: [Task.ResultHandle \[► 1136\]](#).

A task that represents the asynchronous 'AddDeviceNotification' operation. The [ResultHandle \[► 1136\]](#) type parameter contains the created handle ([Handle \[► 1138\]](#)) and the [ErrorCode \[► 1120\]](#) after execution.

Implements

[IAdsNotifications.AddDeviceNotificationExAsync\(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., CancellationToken\) \[► 982\]](#)

Remarks

If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive types (AnyType) are supported by this method. Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [▶ 254] should always called when the notification is not used anymore.

Reference

[AdsClient Class](#) [▶ 183]

[AddDeviceNotificationExAsync Overload](#) [▶ 243]

[TwinCAT.Ads Namespace](#) [▶ 179]

[AdsClient.DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [▶ 254]

[AdsClient.AdsNotificationEx](#) [▶ 391]

[AdsClient.AdsNotificationError](#) [▶ 391]

[AddDeviceNotificationEx Overload](#) [▶ 976]

[TryAddDeviceNotificationEx Overload](#) [▶ 988]

[AddDeviceNotificationExAsync Overload](#) [▶ 981]

6.2.1.2.5 AdsClient.CleanupSymbolTable Method

Clears the internal symbol cache.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public void CleanupSymbolTable()
```

Implements

[IAdsSymbolicAccess.CleanupSymbolTable.](#) [▶ 1070]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Remarks

Previously stored symbol information is cleared. As a consequence the symbol information must be obtained from the ADS server again if accessed, which which needs an extra ADS round trip.

Reference

[AdsClient Class](#) [▶ 183]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.2.6 **AdsClient.Close Method**

Closes this [AdsClient](#) [▶ 183]

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Close()
```

Implements

[IConnection.Close.](#) [▶ 83]







Reference

[AdsClient Class](#) [▶ 183]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.2.7 **AdsClient.Connect Method**

Overload List

	Name	Description
	Connect(AmsAddress) [▶ 248]	Connects the target
	Connect(AmsPort) [▶ 249]	Connects to the local target ADS Device.
	Connect(Int32) [▶ 247]	Connects to the local target ADS Device.
	Connect(AmsNetId, Int32) [▶ 250]	Connects to the target ADS Device.
	Connect(AmsNetId, AmsPort) [▶ 250]	Connects to the target ADS Device.
	Connect(String, Int32) [▶ 249]	Connects to the target ADS Device.

Reference

[AdsClient Class](#) [▶ 183]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.2.7.1 **AdsClient.Connect Method (Int32)**

Connects to the local target ADS Device.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Connect(
    int port
)
```

Parameters

port	Type: System.Int32 The port number of the local ADS target device to connect to.
------	---

Implements

[IAdsConnectAddress.Connect\(Int32\)](#) [[▶ 873](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[Connect Overload](#) [[▶ 247](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.7.2 AdsClient.Connect Method (AmsAddress)

Connects the target

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Connect(
    AmsAddress address
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [▶ 752] The address.
---------	--

Implements

[IAdsConnectAddress.Connect\(AmsAddress\)](#) [[▶ 873](#)]

Exceptions

Exception	Condition
ObjectDisposedException	

Reference

[AdsClient Class](#) [[▶ 183](#)]

[Connect Overload](#) [[▶ 247](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.7.3 AdsClient.Connect Method (AmsPort)

Connects to the local target ADS Device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Connect(
    AmsPort port
)
```

Parameters

port	Type: TwinCAT.Ads.AmsPort [▶ 795] The port number of the local ADS target device to connect to.
------	--

Reference

[AdsClient Class](#) [[▶ 183](#)]

[Connect Overload](#) [[▶ 247](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.7.4 AdsClient.Connect Method (String, Int32)

Connects to the target ADS Device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Connect(
    string netId,
    int port
)
```

Parameters

netId	Type: System.String The AmsNetId [▶ 767] of the ADS target device specified as string.
port	Type: System.Int32 The port number of the ADS target device.

Implements

[IAdsConnectAddress.Connect\(String, Int32\)](#) [[▶ 874](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[Connect Overload](#) [[▶ 247](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.7.5 AdsClient.Connect Method (AmsNetId, Int32)

Connects to the target ADS Device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Connect(
    AmsNetId netId,
    int port
)
```

Parameters

netId	Type: TwinCAT.Ads.AmsNetId [▶ 767] The AmsNetId of the target device.
port	Type: System.Int32 The Ams Port number on the target device to connect to.

Implements

[IAdsConnectAddress.Connect\(AmsNetId, Int32\)](#) [[▶ 874](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[Connect Overload](#) [[▶ 247](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.7.6 AdsClient.Connect Method (AmsNetId, AmsPort)

Connects to the target ADS Device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Connect(
    AmsNetId netId,
    AmsPort port
)
```

Parameters

netId	Type: TwinCAT.Ads.AmsNetId [▶ 767] The AmsNetId [▶ 767] of the ADS target device specified as string.
port	Type: TwinCAT.Ads.AmsPort [▶ 795] The port number of the ADS target device.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[Connect Overload \[▶ 247\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.1.2.8 AdsClient.ConnectAndWaitAsync Method

Connects to the target address and waits until the [AdsClient \[▶ 183\]](#) is disconnected asynchronously.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task ConnectAndWaitAsync(  
    AmsAddress address,  
    CancellationToken cancel  
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [▶ 752] The target address.
cancel	Type: System.Threading.CancellationToken Cancellation Token.

Return Value

Type: [Task](#)

Returns a task object that represents the `ConnectAndWaitAsync(AmsAddress, CancellationToken)` operation as result.

Remarks

This method is used for scenarios, where the [AdsClient \[▶ 183\]](#) disconnects from other code asynchronously. When this method returns, the connection is already terminated and only additional cleanup code should be processed.

Reference

[AdsClient Class \[▶ 183\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.1.2.9 AdsClient.CreateVariableHandle Method

Determines the Symbol handle by its instance path synchronously.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint CreateVariableHandle(  
    string symbolPath  
)
```

Parameters

symbolPath	Type: System.String SymbolName / InstancePath.
------------	---

Return Value

Type: [UInt32](#)
The symbols/variable handle

Implements

[IAdsHandle.CreateVariableHandle\(String\)](#) [[▶ 956](#)]

Remarks

It is a good practice to release all variable handles after use to regain internal resources in the TwinCAT subsystem. The composite method to this [CreateVariableHandle\(String\)](#) is the [DeleteVariableHandle\(UInt32\)](#) [[▶ 255](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[AdsClient.DeleteVariableHandle\(UInt32\)](#) [[▶ 255](#)]

[AdsClient.CreateVariableHandleAsync\(String, CancellationToken\)](#) [[▶ 252](#)]

[AdsClient.TryCreateVariableHandle\(String, UInt32.\)](#) [[▶ 323](#)]

6.2.1.2.10 AdsClient.CreateVariableHandleAsync Method

Determines the Symbol handle by its instance path asynchronously.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultHandle> CreateVariableHandleAsync (
    string symbolPath,
    CancellationToken cancel
)
```

Parameters

symbolPath	Type: System.String SymbolName / InstancePath.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultHandle](#) [[▶ 1136](#)].

A task that represents the asynchronous 'CreateVariableHandle' operation. The [ResultHandle](#) [[▶ 1136](#)] parameter contains the variable handle ([Handle](#) [[▶ 1138](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsHandle.CreateVariableHandleAsync\(String, CancellationToken\) \[► 956\]](#)

Remarks

It is a good practice to release all variable handles after use to regain internal resources in the TwinCAT subsystem. The composite method to this [CreateVariableHandleAsync\(String, CancellationToken\)](#) is the [DeleteVariableHandleAsync\(UInt32, CancellationToken\) \[► 255\]](#)

Reference

[AdsClient Class \[► 183\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

[AdsClient.DeleteVariableHandleAsync\(UInt32, CancellationToken\) \[► 255\]](#)

[AdsClient.TryCreateVariableHandle\(String, UInt32.\) \[► 323\]](#)

[AdsClient.CreateVariableHandle\(String\) \[► 251\]](#)

6.2.1.2.11 AdsClient.DeleteDeviceNotification Method

Deletes a registered notification.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void DeleteDeviceNotification(  
    uint notificationHandle  
)
```

Parameters

notificationHandle	Type: System.UInt32 Notification handle.
--------------------	---

Implements

[IAdsNotifications.DeleteDeviceNotification\(UInt32\) \[► 983\]](#)

Remarks

This is the complementary method to [AddDeviceNotification Overload \[► 971\]](#) overloads and should be called when the notification is not needed anymore the free TwinCAT realtime resources.

Reference

[AdsClient Class \[► 183\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

[AddDeviceNotification Overload \[► 971\]](#)

[AdsClient.AdsNotification \[► 390\]](#)

[TryAddDeviceNotification Overload \[► 985\]](#)

[AddDeviceNotificationAsync Overload \[► 973\]](#)

6.2.1.2.12 AdsClient.DeleteDeviceNotificationAsync Method

Deletes a registered notification asynchronously.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAds> DeleteDeviceNotificationAsync (
    uint notificationHandle,
    CancellationToken cancel
)
```

Parameters

notificationHandle	Type: System.UInt32 Notification handle.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds \[► 1116\]](#).

A task that represents the asynchronous 'DeleteDeviceNotification' operation. The [ErrorCode \[► 1120\]](#) property contains the ADS error code after execution.

Implements

[IAdsNotifications.DeleteDeviceNotificationAsync\(UInt32, CancellationToken\) \[► 984\]](#)

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [► 67]	

Remarks

This is the complementary method to [AddDeviceNotificationAsync Overload \[► 973\]](#) overloads and should be called when the notification is not needed anymore the free TwinCAT realtime resources.

Reference

[AdsClient Class \[► 183\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

[AddDeviceNotificationAsync Overload \[► 973\]](#)

[IAdsNotifications.AdsNotification \[► 991\]](#)

[TryAddDeviceNotification Overload \[▶ 985\]](#)

[AddDeviceNotification Overload \[▶ 971\]](#)

6.2.1.2.13 **AdsClient.DeleteVariableHandle Method**

Releases the specified symbol/variable handle synchronously.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void DeleteVariableHandle(  
    uint variableHandle  
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable
----------------	---

Return Value

Type:
The ADS error code.

Implements

[IAdsHandle.DeleteVariableHandle\(UInt32\) \[▶ 957\]](#)

Remarks

It is a good practice to release all variable handles after use to regain internal resources in the TwinCAT subsystem. The composite method to this [TryDeleteVariableHandle\(UInt32\) \[▶ 326\]](#) is the [TryCreateVariableHandle\(String, UInt32.\) \[▶ 323\]](#)

Reference

[AdsClient Class \[▶ 183\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

[AdsClient.CreateVariableHandle\(String\) \[▶ 251\]](#)

[AdsClient.TryDeleteVariableHandle\(UInt32\) \[▶ 326\]](#)

[AdsClient.DeleteVariableHandleAsync\(UInt32, CancellationToken\) \[▶ 255\]](#)

6.2.1.2.14 **AdsClient.DeleteVariableHandleAsync Method**

Releases the specified symbol/variable handle asynchronously.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAds> DeleteVariableHandleAsync(
    uint variableHandle,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds](#) [► 1116].

A task that represents the asynchronous 'DeleteVariableHandle' operation. The [ResultAds](#) [► 1116] parameter contains the [ErrorCode](#) [► 1120] after execution.

Implements

[IAdsHandle.DeleteVariableHandleAsync\(UInt32, CancellationToken\)](#) [► 958]

Remarks

It is a good practice to release all variable handles after use to regain internal resources in the TwinCAT subsystem. The composite method to this [DeleteVariableHandleAsync\(UInt32, CancellationToken\)](#) is the [CreateVariableHandleAsync\(String, CancellationToken\)](#) [► 252]

Reference

[AdsClient Class](#) [► 183]

[TwinCAT.Ads Namespace](#) [► 179]

[AdsClient.CreateVariableHandleAsync\(String, CancellationToken\)](#) [► 252]

[AdsClient.TryDeleteVariableHandle\(UInt32\)](#) [► 326]

[AdsClient.DeleteVariableHandle\(UInt32\)](#) [► 255]

6.2.1.2.15 AdsClient.Disconnect Method

Disconnects this [AdsClient](#) [► 183] from the local ADS router.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Disconnect()
```

Return Value

Type: [Boolean](#)

true if disconnected, false otherwise.

Implements

[IConnection.Disconnect.](#) [[▶ 84](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.16 **AdsClient.Dispose Method**

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Dispose()
```

Implements

[IDisposable.Dispose.](#)

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.17 **AdsClient.Finalize Method**

Finalizes an instance of the [AdsClient](#) [[▶ 183](#)] class.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected override void Finalize()
```

Implements

[Object.Finalize.](#)







Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.18 AdsClient.InvokeRpcMethod Method

Overload List

	Name	Description
 	InvokeRpcMethod(String, .Object.) [▶ 258]	Invokes the specified RPC Method
 	InvokeRpcMethod(String, .Object., .Object.) [▶ 260]	Invokes the specified RPC Method
 	InvokeRpcMethod(String, .Object., .AnyTypeSpecifier, .AnyTypeSpecifier, .Object.) [▶ 261]	Invokes the specified RPC Method

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.18.1 AdsClient.InvokeRpcMethod Method (String, String, .Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Object InvokeRpcMethod(
    string symbolPath,
    string methodName,
    Object[] inParameters
)
```

Parameters

symbolPath	Type: System.String The symbol path.
methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The input parameters or NULL

Return Value

Type: [Object](#)

The return value of the Method (as object).

Implements

[IAdsRpcInvoke.InvokeRpcMethod\(String, String, Object.\)](#) [[▶ 1015](#)]

Remarks

This method only supports primitive data types as inParameters. Any available outparameters will be ignored. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] { (short) 3, (short) 4 });

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```

Reference

[AdsClient Class](#) [[▶ 183](#)]

[InvokeRpcMethod Overload](#) [[▶ 258](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.18.2 **AdsClient.InvokeRpcMethod Method (String, String, .Object., .Object.)**

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Object InvokeRpcMethod(
    string symbolPath,
    string methodName,
    Object[] inParameters,
    out Object[] outParameters
)
```

Parameters

symbolPath	Type: System.String The symbol path.
methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The input parameters or NULL
outParameters	Type: .System.Object . The output parameters.

Return Value

Type: [Object](#)

The return value of the Method (as object).

Implements

[IAdsRpcInvoke.InvokeRpcMethod\(String, String, .Object., .Object.\)](#) [[▶ 1017](#)]

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [[▶ 2094](#)] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
```

```

/// Defines the entry point of the application.
/// </summary>
/// <param name="args">The arguments.</param>
static void Main(string[] args)
{
    // Get the AdsAddress from command-line arguments
    AmsAddress address = ArgParser.Parse(args);

    using (AdsClient client = new AdsClient())
    {
        //client.Synchronize = false;

        // Connect to the target device
        client.Connect(address);

        SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
        ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

        // Get the Symbols (Dynamic Symbols)

        IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

        // Call a Method that has the following signature (within MAIN Program)
        /* {attribute 'TcRpcEnable'}
        METHOD PUBLIC M_Add : INT
        VAR_INPUT
            i1 : INT := 0;
            i2 : INT := 0;
        END_VAR
        */

        short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

        // Call a Method that has no parameter and returns VOID
        main.InvokeRpcMethod("M_Method1", new object[] {});

        //Browsing RpcMethods
        foreach(IRpcMethod method in main.RpcMethods)
        {
            string methodName = method.Name;

            foreach(IRpcMethodParameter parameter in method.Parameters)
            {
                string parameterName = parameter.Name;
                string parameterType = parameter.TypeName;
            }
        }
    }
}

```

Reference

[AdsClient Class \[► 183\]](#)

[InvokeRpcMethod Overload \[► 258\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.18.3 AdsClient.InvokeRpcMethod Method (String, String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object..)

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Object InvokeRpcMethod(
    string symbolPath,
    string methodName,
    Object[] inParameters,
    AnyTypeSpecifier[] outSpecifiers,
    AnyTypeSpecifier retSpecifier,
    out Object[] outParameters
)
```

Parameters

symbolPath	Type: System.String The symbol path.
methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094]. The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094]. The ret specifier (specifying the return value) or NULL.
outParameters	Type: .System.Object .. The out parameters.

Return Value

Type: [Object](#)
The return value of the Method (as object).

Implements

[IAdsRpcInvoke.InvokeRpcMethod\(String, String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object..\)](#)
[▶ 1018](#)

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;
        }
    }
}
```

```

// Connect to the target device
client.Connect(address);

SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.VirtualTree);
ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

// Get the Symbols (Dynamic Symbols)

IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

// Call a Method that has the following signature (within MAIN Program)
/* {attribute 'TcRpcEnable'}
METHOD PUBLIC M_Add : INT
VAR_INPUT
    i1 : INT := 0;
    i2 : INT := 0;
END_VAR
*/

short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

// Call a Method that has no parameter and returns VOID
main.InvokeRpcMethod("M_Method1", new object[] {});

//Browsing RpcMethods
foreach(IRpcMethod method in main.RpcMethods)
{
    string methodName = method.Name;

    foreach(IRpcMethodParameter parameter in method.Parameters)
    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
}

```

Reference

[AdsClient Class \[▶ 183\]](#)




[InvokeRpcMethod Overload \[▶ 258\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.1.2.19 AdsClient.InvokeRpcMethodAsync Method

Overload List

	Name	Description
	InvokeRpcMethodAsync(IRpcCallableInstance, IRpcMethod, .Object t, Cancellation token) [▶ 269]	invoke RPC method as an asynchronous operation.
	InvokeRpcMethodAsync(String, String, .Object, Cancellation token) [▶ 264]	Invokes the specified RPC Method asynchronously

	Name	Description
	InvokeRpcMethodAsync(IRpcCallableInstance, IRpcMethod, .Object, .AnyTypeSpecifier, .AnyTypeSpecifier, CancellationToken) [▶ 268]	invoke RPC method as an asynchronous operation.
 	InvokeRpcMethodAsync(String, String, .Object, .AnyTypeSpecifier, .AnyTypeSpecifier, CancellationToken) [▶ 266]	Invokes the specified RPC Method asynchronously

Reference

[AdsClient Class](#) [▶ 183]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.2.19.1 AdsClient.InvokeRpcMethodAsync Method (String, String, .Object, CancellationToken)

Invokes the specified RPC Method asynchronously

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultRpcMethod> InvokeRpcMethodAsync (
    string symbolPath,
    string methodName,
    Object[]? inParameters,
    CancellationToken cancel
)
```

Parameters

symbolPath	Type: System.String The symbol/Instance path of the symbol.
methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
cancel	Type: System.Threading.CancellationTok The cancellation token

Return Value

Type: [Task.ResultRpcMethod](#) [▸ 1176].

A task that represents the asynchronous 'InvokeRpcMethod' operation. The [ResultRpcMethod](#) [▸ 1176] results contains the return value together with the output parameters.

Implements

[IAdsRpcInvoke.InvokeRpcMethodAsync\(String, String, .Object., CancellationToken\)](#) [▸ 1021]

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [▸ 2094] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```

```

    }
}

```

Reference

[AdsClient Class \[► 183\]](#)

[InvokeRpcMethodAsync Overload \[► 263\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.19.2 AdsClient.InvokeRpcMethodAsync Method (String, String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, CancellationToken)

Invokes the specified RPC Method asynchronously

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public Task<ResultRpcMethod> InvokeRpcMethodAsync (
    string symbolPath,
    string methodName,
    Object[]? inParameters,
    AnyTypeSpecifier[]? outSpecifiers,
    AnyTypeSpecifier? retSpecifier,
    CancellationToken cancel
)

```

Parameters

symbolPath	Type: System.String The symbol/Instance path of the symbol.
methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [► 2094] . The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [► 2094] The ret specifier (specifying the return value) or NULL.
cancel	Type: System.Threading.CancellationToken The cancellation token

Return Value

Type: [Task.ResultRpcMethod \[► 1176\]](#).

A task that represents the asynchronous 'InvokeRpcMethod' operation. The [ResultRpcMethod \[► 1176\]](#) results contains the return value together with the output parameters.

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set. [ReturnValue \[► 1179\]](#) and the [ErrorCode \[► 1120\]](#) of the ADS communication after execution.

Implements

[IAdsRpcInvoke.InvokeRpcMethodAsync\(String, String, Object, AnyTypeSpecifier, AnyTypeSpecifier, CancellationToken\)](#) [[▶ 1022](#)]

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```

Reference

[AdsClient Class](#) [[▶ 183](#)]

[InvokeRpcMethodAsync Overload](#) [[▶ 263](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.19.3 **AdsClient.InvokeRpcMethodAsync Method (IRpcCallableInstance, IRpcMethod, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, CancellationToken)**

invoke RPC method as an asynchronous operation.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultRpcMethod> InvokeRpcMethodAsync (
    IRpcCallableInstance symbol,
    IRpcMethod rpcMethod,
    Object[]? inParameters,
    AnyTypeSpecifier[]? outSpec,
    AnyTypeSpecifier? returnSpec,
    CancellationToken cancel
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IRpcCallableInstance [► 2606] The symbol.
rpcMethod	Type: TwinCAT.TypeSystem.IRpcMethod [► 2625] The RPC method.
inParameters	Type: .System.Object . The in parameters.
outSpec	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [► 2094]. The out spec.
returnSpec	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [► 2094] The return spec.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultRpcMethod](#) [► 1176].

Task<ResultRpcMethod>.

Implements

[IAdsRpcInvoke.InvokeRpcMethodAsync\(IRpcCallableInstance, IRpcMethod, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, CancellationToken\)](#) [► 1024]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [► 67]	
ArgumentNullException	symbol
ArgumentNullException	rpcMethod

Reference

[AdsClient Class \[► 183\]](#)

[InvokeRpcMethodAsync Overload \[► 263\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.19.4 AdsClient.InvokeRpcMethodAsync Method (IRpcCallableInstance, IRpcMethod, .Object., CancellationToken)

invoke RPC method as an asynchronous operation.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultRpcMethod> InvokeRpcMethodAsync (
    IRpcCallableInstance symbol,
    IRpcMethod rpcMethod,
    Object[]? inParameters,
    CancellationToken cancel
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IRpcCallableInstance [► 2606] The symbol.
rpcMethod	Type: TwinCAT.TypeSystem.IRpcMethod [► 2625] The RPC method.
inParameters	Type: .System.Object. The in parameters.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultRpcMethod \[► 1176\]](#).

Task<ResultRpcMethod>.

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [► 67]	
ArgumentNullException	symbol
ArgumentNullException	rpcMethod

Reference



[AdsClient Class \[► 183\]](#)

[InvokeRpcMethodAsync Overload \[► 263\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.20 AdsClient.Read Method

Overload List

	Name	Description
	Read(UInt32, Memory.Byte.) [▶ 271]	Reads data synchronously from an ADS device and writes to the specified readBuffer.
	Read(UInt32, UInt32, Memory.Byte.) [▶ 272]	Reads data synchronously from an ADS device and writes it to the given readBuffer

Reference

[AdsClient Class](#) [▶ 183]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.2.20.1 AdsClient.Read Method (UInt32, Memory`1)

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public int Read(
    uint variableHandle,
    Memory readBuffer
)
```

Parameters

variableHandle Type: [System.UInt32](#)

readBuffer Type: [Memory](#)

Return Value

Type: [Int32](#)

Reference

[AdsClient Class](#) [▶ 183]

[Read Overload](#) [▶ 270]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.2.20.2 AdsClient.Read Method (UInt32, UInt32, Memory`1)

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public int Read(
    uint indexGroup,
    uint indexOffset,
    Memory readBuffer
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
readBuffer	Type: Memory

Return Value

Type: [Int32](#)

Reference

[AdsClient Class](#) [► 183]

[Read Overload](#) [► 270]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.2.20.3 AdsClient.Read Method (UInt32, Memory.Byte.)

Reads data synchronously from an ADS device and writes to the specified readBuffer.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Read(
    uint variableHandle,
    Memory<byte> readBuffer
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable
readBuffer	Type: System.Memory.Byte. The read buffer / data

Return Value

Type: [Int32](#)

Number of successfully returned data bytes.

Implements

[IAdsHandle.Read\(UInt32, Memory.Byte.\)](#) [► 959]

Reference[AdsClient Class](#) [[▶ 183](#)][Read Overload](#) [[▶ 270](#)][TwinCAT.Ads Namespace](#) [[▶ 179](#)][AdsClient.TryRead\(UInt32, Memory.Byte., Int32.\)](#) [[▶ 337](#)][AdsClient.ReadAsync\(UInt32, Memory.Byte., CancellationToken\)](#) [[▶ 294](#)]**6.2.1.2.20.4 AdsClient.Read Method (UInt32, UInt32, Memory.Byte.)**

Reads data synchronously from an ADS device and writes it to the given readBuffer

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public int Read(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
readBuffer	Type: System.Memory.Byte. Memory location, where to read the data.









Return ValueType: [Int32](#)

Number of successfully returned (read) data bytes.

Implements[IAdsReadWrite2.Read\(UInt32, UInt32, Memory.Byte.\)](#) [[▶ 1002](#)]**Reference**[AdsClient Class](#) [[▶ 183](#)][Read Overload](#) [[▶ 270](#)][TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.21 AdsClient.ReadAny Method

Overload List

	Name	Description
	ReadAny.T.(UInt32) [▶ 273]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, Type) [▶ 275]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, .Int32.) [▶ 274]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, UInt32) [▶ 275]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, Type, .Int32.) [▶ 276]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type) [▶ 278]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, UInt32, .Int32.) [▶ 277]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type, .Int32.) [▶ 279]	Reads data synchronously from an ADS device and writes it to an object.

Reference

[AdsClient Class](#) [▶ 183]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.2.21.1 AdsClient.ReadAny.T. Method (UInt32)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T ReadAny<T>(
    uint variableHandle
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
----------------	--

Type Parameters

T	The type of the value to read.
---	--------------------------------

Return Value

Type: T

The value of the read symbol.

Implements[IAdsAnyAccess.ReadAny.T.\(UInt32\) \[► 810\]](#)**Reference**[AdsClient Class \[► 183\]](#)[ReadAny Overload \[► 273\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.1.2.21.2 AdsClient.ReadAny.T. Method (UInt32, .Int32.)**

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public T ReadAny<T>(
    uint variableHandle,
    int[]? args
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
args	Type: .System.Int32 . Additional arguments.

Type Parameters

T	The type of the value to read.
---	--------------------------------

Return Value

Type: T

The value of the read symbol.

Implements[IAdsAnyAccess.ReadAny.T.\(UInt32, .Int32.\) \[► 811\]](#)**Remarks**

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference

[AdsClient Class](#) [[▶ 183](#)]

[ReadAny Overload](#) [[▶ 273](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.21.3 **AdsClient.ReadAny Method (UInt32, Type)**

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object ReadAny(
    uint variableHandle,
    Type type
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
type	Type: System.Type Type of the object to be read.

Return Value

Type: [Object](#)
The read object.

Implements

[IAdsAnyAccess.ReadAny\(UInt32, Type\)](#) [[▶ 812](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[ReadAny Overload](#) [[▶ 273](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.21.4 **AdsClient.ReadAny.T. Method (UInt32, UInt32)**

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T ReadAny<T>(
    uint indexGroup,
    uint indexOffset
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.

Type Parameters

T	The type of the object to be read.
---	------------------------------------

Return Value

Type: T
The read value.

Implements

[IAdsAnyAccess.ReadAny.T.\(UInt32, UInt32\)](#) [[▶ 812](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[ReadAny Overload](#) [[▶ 273](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.21.5 AdsClient.ReadAny Method (UInt32, Type, .Int32.)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object ReadAny(
    uint variableHandle,
    Type type,
    int[]? args
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
type	Type: System.Type Type of the object to be read.
args	Type: .System.Int32 . Additional arguments.

Return Value

Type: [Object](#)
 The read value.

Implements

[IAdsAnyAccess.ReadAny\(UInt32, Type, .Int32.\)](#) [[▶ 813](#)]

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference

[AdsClient Class](#) [[▶ 183](#)]

[ReadAny Overload](#) [[▶ 273](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.21.6 AdsClient.ReadAny.T. Method (UInt32, UInt32, .Int32.)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T ReadAny<T>(
    uint indexGroup,
    uint indexOffset,
    int[]? args
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
args	Type: .System.Int32 . Additional arguments.

Type Parameters

T	The type of the object to be read.
---	------------------------------------

Return Value

Type: T
The read value.

Implements

[IAdsAnyAccess.ReadAny.T.\(UInt32, UInt32, Int32.\)](#) [[▶ 814](#)]

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference

[AdsClient Class](#) [[▶ 183](#)]

[ReadAny Overload](#) [[▶ 273](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.21.7 AdsClient.ReadAny Method (UInt32, UInt32, Type)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Object ReadAny(
    uint indexGroup,
    uint indexOffset,
    Type type
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
type	Type: System.Type Type of the object to be read.

Return Value

Type: [Object](#)
The read value.

Implements

[IAdsAnyAccess.ReadAny\(UInt32, UInt32, Type\) \[► 815\]](#)

Reference

[AdsClient Class \[► 183\]](#)

[ReadAny Overload \[► 273\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.21.8 AdsClient.ReadAny Method (UInt32, UInt32, Type, .Int32.)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object ReadAny(
    uint indexGroup,
    uint indexOffset,
    Type type,
    int[]? args
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
type	Type: System.Type Type of the object to be read.
args	Type: .System.Int32 . Additional arguments.

Return Value

Type: [Object](#)
The read value.

Implements

[IAdsAnyAccess.ReadAny\(UInt32, UInt32, Type, .Int32.\) \[► 815\]](#)

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [► 67]	

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	







Reference



[AdsClient Class](#) [[▶ 183](#)]

[ReadAny Overload](#) [[▶ 273](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.22 AdsClient.ReadAnyAsync Method**Overload List**

	Name	Description
	ReadAnyAsync.T. (UInt32 , CancellationToken) [▶ 281]	Reads data synchronously from an ADS device.
	ReadAnyAsync(UInt32 , Type , CancellationToken) [▶ 283]	Reads the value of an Anytype (Primitive type) asynchronously.
	ReadAnyAsync.T. (UInt32 , .Int32 , CancellationToken) [▶ 282]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T. (UInt32 , UInt32 , CancellationToken) [▶ 284]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32 , Type , .Int32 , CancellationToken) [▶ 285]	Reads the value of an Anytype (Primitive type) asynchronously.
	ReadAnyAsync(UInt32 , UInt32 , Type , CancellationToken) [▶ 287]	Reads the value of an Anytype (Primitive type) asynchronously.

	Name	Description
	ReadAnyAsync.T. (UInt32, UInt32, .Int32., CancellationToken) [▶ 285]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt 32, UInt32, Type, .Int32., CancellationToken) [▶ 287]	Reads the value of an Anytype (Primitive type) asynchronously.

Reference

[AdsClient Class \[▶ 183\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.1.2.22.1 AdsClient.ReadAnyAsync.T. Method (UInt32, CancellationToken)

Reads data synchronously from an ADS device.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultValue<T>> ReadAnyAsync<T>(
    uint variableHandle,
    Cancellation token cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable/symbol handle.
cancel	Type: System.Threading.Cancellation token The cancellation token.

Type Parameters

T	The Type of the value to be read.
---	-----------------------------------

Return Value

Type: [Task.ResultValue \[▶ 1181\].T..](#)

A task that represents the asynchronous read operation. The [ResultAnyValue \[▶ 1129\]](#) parameter contains the read value ([Value \[▶ 1185\]](#)) and the [ErrorCode \[▶ 1120\]](#) after execution.

Implements

[IAdsAnyAccess.ReadAnyAsync.T.\(UInt32, CancellationToken\) \[▶ 817\]](#)

Remarks

As object types only primitive types are supported.

Reference

[AdsClient Class](#) [► 183]

[ReadAnyAsync Overload](#) [► 280]

[TwinCAT.Ads Namespace](#) [► 179]

Also see about this

 [ResultAnyValue.Value Property](#) [► 1131]

6.2.1.2.22.2 **AdsClient.ReadAnyAsync.T. Method (UInt32, .Int32., CancellationToken)**

Reads data asynchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultValue<T>> ReadAnyAsync<T>(
    uint variableHandle,
    int[]? args,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
args	Type: .System.Int32 . Additional arguments.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Type Parameters

T	Type of the object to be read
---	-------------------------------

Return Value

Type: [Task.ResultValue](#) [► 1181].T..

A task that represents the asynchronous read operation. The [ResultValue.TValue](#). [► 1181] parameter contains the read value ([Value](#) [► 1185]) and the [ErrorCode](#) [► 1120] after execution.

Implements

[IAdsAnyAccess.ReadAnyAsync.T.\(UInt32, .Int32., CancellationToken\)](#) [► 818]

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference

[AdsClient Class](#) [[▶ 183](#)]

[ReadAnyAsync Overload](#) [[▶ 280](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.22.3 AdsClient.ReadAnyAsync Method (UInt32, Type, CancellationToken)

Reads the value of an Anytype (Primitive type) asynchronously.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAnyValue> ReadAnyAsync(
    uint variableHandle,
    Type type,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
type	Type: System.Type The type as AnyType (primitive types).
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAnyValue](#) [[▶ 1129](#)].

A task that represents the asynchronous 'ReadState' operation. The [ResultAnyValue](#) [[▶ 1129](#)] parameter contains the value [Value](#) [[▶ 1185](#)] and the [ErrorCode](#) [[▶ 1120](#)] of the ADS communication after execution.

Implements

[IAdsAnyAccess.ReadAnyAsync\(UInt32, Type, CancellationToken\)](#) [[▶ 819](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[ReadAnyAsync Overload \[► 280\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

Also see about this

 [ResultAnyValue.Value Property \[► 1131\]](#)

6.2.1.2.22.4 AdsClient.ReadAnyAsync.T. Method (UInt32, UInt32, CancellationToken)

Reads data asynchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultValue<T>> ReadAnyAsync<T>(
    uint indexGroup,
    uint indexOffset,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Type Parameters

T

Return Value

Type: [Task.ResultValue \[► 1181\].T..](#)
The asynchronous result.

Return Value

Type: [Task.ResultValue \[► 1181\].T..](#)

A task that represents the asynchronous read operation. The [ResultValue.TValue. \[► 1181\]](#) parameter contains the read value ([Value \[► 1185\]](#)) and the [ErrorCode \[► 1120\]](#) after execution.

Implements

[IAdsAnyAccess.ReadAnyAsync.T.\(UInt32, UInt32, CancellationToken\) \[► 820\]](#)

Reference

[AdsClient Class \[► 183\]](#)

[ReadAnyAsync Overload \[► 280\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.22.5 **AdsClient.ReadAnyAsync Method (UInt32, Type, .Int32., CancellationTokentoken)**

Reads the value of an Anytype (Primitive type) asynchronously.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAnyValue> ReadAnyAsync (
    uint variableHandle,
    Type type,
    int[]? args,
    CancellationTokentoken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
type	Type: System.Type The type as AnyType (primitive types).
args	Type: .System.Int32. The type arguments (AnyType)
cancel	Type: System.Threading.CancellationTokentoken The cancellation token.

Return Value

Type: [Task.ResultAnyValue](#) [[▶ 1129](#)].

A task that represents the asynchronous 'ReadState' operation. The [ResultAnyValue](#) [[▶ 1129](#)] parameter contains the value [Value](#) [[▶ 1185](#)] and the [ErrorCode](#) [[▶ 1120](#)] of the ADS communication after execution.

Implements

[IAdsAnyAccess.ReadAnyAsync\(UInt32, Type, .Int32., CancellationTokentoken\)](#) [[▶ 821](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[ReadAnyAsync Overload](#) [[▶ 280](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

 [ResultAnyValue.Value Property](#) [[▶ 1131](#)]

6.2.1.2.22.6 **AdsClient.ReadAnyAsync.T. Method (UInt32, UInt32, .Int32., CancellationTokentoken)**

Reads data asynchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultValue<T>> ReadAnyAsync<T>(
    uint indexGroup,
    uint indexOffset,
    int[]? args,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
args	Type: .System.Int32 . Additional arguments.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Type Parameters

T	The type of the result value.
---	-------------------------------

Return Value

Type: [Task.ResultValue](#) [[▶ 1181](#)].T..

A task that represents the asynchronous read operation. The [ResultValue.TValue](#). [[▶ 1181](#)] parameter contains the read value ([Value](#) [[▶ 1185](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsAnyAccess.ReadAnyAsync.T\(UInt32, UInt32, .Int32., CancellationToken\)](#) [[▶ 822](#)]

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference

[AdsClient Class](#) [[▶ 183](#)]

[ReadAnyAsync Overload](#) [[▶ 280](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.22.7 **AdsClient.ReadAnyAsync Method (UInt32, UInt32, Type, CancellationToken)**

Reads the value of an Anytype (Primitive type) asynchronously.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAnyValue> ReadAnyAsync (
    uint indexGroup,
    uint indexOffset,
    Type type,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
type	Type: System.Type The type as AnyType (primitive types).
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAnyValue](#) [[▶ 1129](#)].

A task that represents the asynchronous 'ReadState' operation. The [ResultAnyValue](#) [[▶ 1129](#)] parameter contains the value [Value](#) [[▶ 1185](#)] and the [ErrorCode](#) [[▶ 1120](#)] of the ADS communication after execution.

Implements

[IAdsAnyAccess.ReadAnyAsync\(UInt32, UInt32, Type, CancellationToken\)](#) [[▶ 823](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[ReadAnyAsync Overload](#) [[▶ 280](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

 [ResultAnyValue.Value Property](#) [[▶ 1131](#)]

6.2.1.2.22.8 **AdsClient.ReadAnyAsync Method (UInt32, UInt32, Type, .Int32., CancellationToken)**

Reads the value of an Anytype (Primitive type) asynchronously.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAnyValue> ReadAnyAsync(
    uint indexGroup,
    uint indexOffset,
    Type type,
    int[]? args,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
type	Type: System.Type The type as AnyType (primitive types).
args	Type: System.Int32 . The type arguments (AnyType)
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAnyValue](#) [[▶ 1129](#)].

A task that represents the asynchronous 'ReadState' operation. The [ResultAnyValue](#) [[▶ 1129](#)] parameter contains the value [Value](#) [[▶ 1185](#)] and the [ErrorCode](#) [[▶ 1120](#)] of the ADS communication after execution.

Implements

[IAdsAnyAccess.ReadAnyAsync\(UInt32, UInt32, Type, .Int32., CancellationToken\)](#) [[▶ 823](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[ReadAnyAsync Overload](#) [[▶ 280](#)]


[TwinCAT.Ads Namespace](#) [[▶ 179](#)]


Also see about this

 [ResultAnyValue.Value Property](#) [[▶ 1131](#)]

6.2.1.2.23 AdsClient.ReadAnyString Method

Overload List

	Name	Description
	ReadAnyString(UInt32, Int32, Encoding) [▶ 289]	Reads a string from the specified symbol/variable.

	Name	Description
	ReadAnyString(UInt32, UInt32, Int32, Encoding) [▶ 289]	Reads as string from a specified address.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.23.1 AdsClient.ReadAnyString Method (UInt32, Int32, Encoding)

Reads a string from the specified symbol/variable.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string ReadAnyString(
    uint variableHandle,
    int len,
    Encoding encoding
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
len	Type: System.Int32 The length.
encoding	Type: System.Text.Encoding The encoding.

Return Value

Type: [String](#)
The string value.

Implements

[IAdsAnyAccess.ReadAnyString\(UInt32, Int32, Encoding\)](#) [[▶ 825](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[ReadAnyString Overload](#) [[▶ 288](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.23.2 AdsClient.ReadAnyString Method (UInt32, UInt32, Int32, Encoding)

Reads as string from a specified address.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string ReadAnyString(
    uint indexGroup,
    uint indexOffset,
    int len,
    Encoding encoding
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
len	Type: System.Int32 The string length to be read.
encoding	Type: System.Text.Encoding The encoding.

Return Value

Type: [String](#)
System.String.

Implements

[IAdsAnyAccess.ReadAnyString\(UInt32, UInt32, Int32, Encoding\)](#) [► 825]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [► 67]	

Reference


[AdsClient Class](#) [► 183]


[ReadAnyString Overload](#) [► 288]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.2.24 AdsClient.ReadAnyStringAsync Method

Overload List

	Name	Description
	ReadAnyStringAsync (UInt32, Int32,	Reads a string asynchronously from the specified symbol/variable

	Name	Description
	Encoding, CancellationToken [▶ 291]	
	ReadAnyStringAsync (UInt32 , UInt32 , Int32 , Encoding , CancellationToken) [▶ 292]	Reads a string from a specified address asynchronously.

Reference

[AdsClient Class](#) [▶ 183]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.2.24.1 **AdsClient.ReadAnyStringAsync Method (UInt32, Int32, Encoding, CancellationToken)**

Reads a string asynchronously from the specified symbol/variable

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAnyValue> ReadAnyStringAsync (
    uint variableHandle,
    int len,
    Encoding encoding,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
len	Type: System.Int32 The length.
encoding	Type: System.Text.Encoding The encoding.
cancel	Type: System.Threading.CancellationTok The cancellation token.

Return Value

Type: [Task.ResultAnyValue](#) [▶ 1129].

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [▶ 1129] parameter contains the read string ([Value](#) [▶ 1185]) and the [ErrorCode](#) [▶ 1120] after execution.

Implements

[IAdsAnyAccess.ReadAnyStringAsync\(UInt32, Int32, Encoding, CancellationToken\)](#) [▶ 827]

Reference[AdsClient Class \[► 183\]](#)[ReadAnyStringAsync Overload \[► 290\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**Also see about this**[ResultAnyValue.Value Property \[► 1131\]](#)**6.2.1.2.24.2 AdsClient.ReadAnyStringAsync Method (UInt32, UInt32, Int32, Encoding, CancellationToken)**

Reads a string from a specified address asynchronously.

Namespace: [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public Task<ResultAnyValue> ReadAnyStringAsync (
    uint indexGroup,
    uint indexOffset,
    int len,
    Encoding encoding,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
len	Type: System.Int32 The string length to be read.
encoding	Type: System.Text.Encoding The encoding.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return ValueType: [Task.ResultAnyValue \[► 1129\]](#).

A task that represents the asynchronous read operation. The [ResultAnyValue \[► 1129\]](#) parameter contains the read value ([Value \[► 1185\]](#)) and the [ErrorCode \[► 1120\]](#) after execution.

Implements[IAdsAnyAccess.ReadAnyStringAsync\(UInt32, UInt32, Int32, Encoding, CancellationToken\) \[► 827\]](#)**Reference**[AdsClient Class \[► 183\]](#)[ReadAnyStringAsync Overload \[► 290\]](#)



[TwinCAT.Ads Namespace \[► 179\]](#)

Also see about this

 [ResultAnyValue.Value Property \[► 1131\]](#)

6.2.1.2.25 AdsClient.ReadAsync Method

Overload List

	Name	Description
	ReadAsync(UInt32, Memory.Byte., CancellationToken) [► 294]	Reads the value data of the symbol asynchronously into the readBuffer.
	ReadAsync(UInt32, UInt32, Memory.Byte., CancellationToken) [► 295]	Reads the data asynchronously from specified IndexGroup/IndexOffset

Reference

[AdsClient Class \[► 183\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.25.1 AdsClient.ReadAsync Method (UInt32, Memory`1, Void)

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Task<ResultRead> ReadAsync(
    uint variableHandle,
    Memory readBuffer,
    void cancel
)
```

Parameters

variableHandle Type: [System.UInt32](#)

readBuffer Type: [Memory](#)

cancel Type: [System.Void](#)

Return Value

Type: [Task.ResultRead \[► 1143\]](#).

Reference

[AdsClient Class \[► 183\]](#)

[ReadAsync Overload \[► 293\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.25.2 AdsClient.ReadAsync Method (UInt32, UInt32, Memory`1, Void)

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Task<ResultRead> ReadAsync(  
    uint indexGroup,  
    uint indexOffset,  
    Memory readBuffer,  
    void cancel  
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
readBuffer	Type: Memory
cancel	Type: System.Void

Return Value

Type: [Task.ResultRead \[► 1143\]](#).

Reference

[AdsClient Class \[► 183\]](#)

[ReadAsync Overload \[► 293\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.25.3 AdsClient.ReadAsync Method (UInt32, Memory<Byte>, CancellationToken)

Reads the value data of the symbol asynchronously into the readBuffer.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultRead> ReadAsync(  
    uint variableHandle,  
    Memory<byte> readBuffer,  
    CancellationToken cancel  
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
readBuffer	Type: System.Memory.Byte . The read buffer/data.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultRead](#) [[▶ 1143](#)].

A task that represents the asynchronous read operation. The [ResultRead](#) [[▶ 1143](#)] parameter contains the total number of bytes read into the buffer ([ReadBytes](#) [[▶ 1145](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution..

Implements

[IAdsHandle.ReadAsync\(UInt32, Memory.Byte., CancellationToken\)](#) [[▶ 959](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[ReadAsync Overload](#) [[▶ 293](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.25.4 AdsClient.ReadAsync Method (UInt32, UInt32, Memory.Byte., CancellationToken)

Reads the data asynchronously from specified IndexGroup/IndexOffset

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultRead> ReadAsync(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
readBuffer	Type: System.Memory.Byte . The read buffer, memory area where the data is written.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultRead](#) [► 1143].

A task that represents the asynchronous read operation. The [ResultRead](#) [► 1143] parameter contains the total number of bytes read into the buffer ([ReadBytes](#) [► 1145]) and the [ErrorCode](#) [► 1120] after execution..

Implements

[IAdsReadWrite.ReadAsync\(UInt32, UInt32, Memory.Byte., CancellationToken\)](#) [► 996]

Reference

[AdsClient Class](#) [► 183]

[ReadAsync Overload](#) [► 293]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.2.26 AdsClient.ReadDataType Method

Call this method to obtain information about the specified data type.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataType ReadDataType(  
    string typeName  
)
```

Parameters

typeName	Type: System.String Name of the data type (without namespace)
----------	--

Return Value

Type: [IDataType](#) [► 2475]

An containing the requested type.

Implements

[IAdsSymbolicAccess.ReadDataType\(String\)](#) [► 1070]

Exceptions

Exception	Condition
ArgumentOutOfRangeException	typeName
ObjectDisposedException	
ClientNotConnectedException [► 67]	

Reference

[AdsClient Class \[► 183\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

[IAdsSymbolicAccess.TryReadDataType\(String, IDataType.\) \[► 1079\]](#)

[IAdsSymbolicAccess.ReadDataTypeAsync\(String, CancellationToken\) \[► 1070\]](#)

6.2.1.2.27 AdsClient.ReadDataTypeAsync Method

read data type as an asynchronous operation.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultValue<IDatatype>> ReadDataTypeAsync (
    string typeName,
    CancellationToken cancel
)
```

Parameters

typeName	Type: System.String Name of the data type.
cancel	Type: System.Threading.CancellationToken The cancel token.

Return Value

Type: [Task.ResultValue \[► 1181\].IDatatype \[► 2475\]](#)..

A task that represents the asynchronous 'ReadDataType' operation. The [ResultValue.TValue. \[► 1181\]](#) parameter contains the read value ([Value \[► 1185\]](#)) and the [ErrorCode \[► 1120\]](#) after execution.

Implements

[IAdsSymbolicAccess.ReadDataTypeAsync\(String, CancellationToken\) \[► 1070\]](#)

Exceptions

Exception	Condition
ArgumentOutOfRangeException	typeName
ObjectDisposedException	
ClientNotConnectedException [► 67]	

Reference

[AdsClient Class \[► 183\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

[IAdsSymbolicAccess.ReadDataType\(String\)](#) [► 1070]

[IAdsSymbolicAccess.TryReadDataType\(String, IDataType.\)](#) [► 1079]

6.2.1.2.28 **AdsClient.ReadDeviceInfo Method**

Reads the identification and version number of an ADS server.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DeviceInfo ReadDeviceInfo()
```

Return Value

Type: [DeviceInfo](#) [► 801]

DeviceInfo struct containing the name of the device and the version information.

Implements

[IAdsConnection.ReadDeviceInfo.](#) [► 914]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [► 67]	

Reference

[AdsClient Class](#) [► 183]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.2.29 **AdsClient.ReadDeviceInfoAsync Method**

Reads the identification and version number of an ADS server.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultDeviceInfo> ReadDeviceInfoAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultDeviceInfo](#) [► 1132].

A task that represents the asynchronous 'ReadDeviceState' operation. The [ResultDeviceInfo](#) [► 1132] parameter contains the value [DeviceInfo](#) [► 1134] and the [ErrorCode](#) [► 1120] of the ADS communication after execution.

Implements

[IAdsConnection.ReadDeviceInfoAsync\(CancellationTokens\)](#) [► 914]

Reference

[AdsClient Class](#) [► 183]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.2.30 **AdsClient.ReadState Method**

Reads the ADS status and the device status from an ADS server.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public StateInfo ReadState()
```

Return Value

Type: [StateInfo](#) [► 1200]

The ADS statue and device status.

Implements

[IAdsStateProvider.ReadState.](#) [► 1058]

Remarks

Not all ADS Servers support the State ADS Request.

Reference

[AdsClient Class](#) [► 183]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.2.31 **AdsClient.ReadStateAsync Method**

Reads the ADS status and the device status from an ADS server.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultReadDeviceState> ReadStateAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token
--------	--

Return Value

Type: [Task.ResultReadDeviceState](#) [[▶ 1157](#)].

A task that represents the asynchronous 'ReadState' operation. The [ResultReadDeviceState](#) [[▶ 1157](#)] parameter contains the state ([State](#) [[▶ 1159](#)]) as long as the [ErrorCode](#) [[▶ 1120](#)] of the ADS communication after execution.

Implements

[IAdsStateProvider.ReadStateAsync\(CancellationToken\)](#) [[▶ 1058](#)]

Exceptions

Exception	Condition
ObjectDisposedException	

Remarks

Not all ADS Servers support the State ADS Request

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.32 AdsClient.ReadSymbol Method

Call this method to obtain information about the individual symbols (variables) in ADS devices.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IAdsSymbol ReadSymbol(
    string name
)
```

Parameters

name	Type: System.String Name of the symbol.
------	--

Return Value

Type: [IAdsSymbol](#) [[▶ 1755](#)]

A [IAdsSymbol2](#) containing the requested symbol information or null if symbol could not be found.

Implements

[IAdsSymbolicAccess.ReadSymbol\(String\)](#) [[▶ 1071](#)]

Exceptions

Exception	Condition
AdsErrorException [▶ 682]	Thrown when the ADS call fails.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.33 AdsClient.ReadSymbolAsync Method

Call this method to obtain information about the individual symbols (variables) in ADS devices.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultValue<IAdsSymbol>> ReadSymbolAsync (
    string name,
    CancellationToken cancel
)
```

Parameters

name	Type: System.String Name of the symbol.
cancel	Type: System.Threading.CancellationToken The cancel token.

Return Value

Type: [Task.ResultValue](#) [[▶ 1181](#)].[IAdsSymbol](#) [[▶ 1755](#)].

A task that represents the asynchronous 'ReadSymbolInfo' operation. The [ResultValue.TValue](#). [[▶ 1181](#)] parameter contains the read value (Value) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsSymbolicAccess.ReadSymbolAsync\(String, CancellationToken\)](#) [[▶ 1072](#)]





Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.34 AdsClient.ReadValue Method

Overload List

	Name	Description
	ReadValue(ISymbol) [▶ 303]	Reads the value of a symbol and returns it as an object.
	ReadValue.T. (ISymbol) [▶ 303]	Reads the value of a symbol and returns it as an object.
	ReadValue.T.(String) [▶ 302]	Reads the value of a symbol and returns the value. The parameter type must have the same layout as the ADS symbol.
	ReadValue(String, Type) [▶ 304]	Reads the value of a symbol and returns the value as object. The parameter type must have the same layout as the ADS symbol.

Reference

[AdsClient Class](#) [▶ 183]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.2.34.1 AdsClient.ReadValue.T. Method (String)

Reads the value of a symbol and returns the value. The parameter type must have the same layout as the ADS symbol.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T ReadValue<T>(
    string name
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
------	--

Type Parameters

T	The value type
---	----------------

Return Value

Type: T
Value of the symbol

Implements

[IAdsSymbolicAccess.ReadValue.T.\(String\)](#) [▶ 1073]

Reference

[AdsClient Class](#) [▶ 183]

[ReadValue Overload](#) [▶ 302]

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.34.2 AdsClient.ReadValue Method (ISymbol)

Reads the value of a symbol and returns it as an object.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object ReadValue(  
    ISymbol symbol  
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol that should be read.
--------	--

Return Value

Type: [Object](#)

The value of the symbol as an object.

Implements

[IAdsSymbolicAccess.ReadValue\(ISymbol\) \[► 1073\]](#)

Remarks

Supported types are limited to 'AnyTypes', what includes all primitive types (UInt32, Int32, Bool etc.), strings, and Arrays that are compositions of 'AnyTypes'. Structs are not supported.

Reference

[AdsClient Class \[► 183\]](#)

[ReadValue Overload \[► 302\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.34.3 AdsClient.ReadValue.T Method (ISymbol)

Reads the value of a symbol and returns it as an object.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T ReadValue<T>( ISymbol symbol  
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▸ 2691] The symbol that should be read.
--------	--

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: T
The value of the symbol.

Implements

[IAdsSymbolicAccess.ReadValue.T.\(ISymbol\)](#) [[▸ 1074](#)]

Remarks

Supported types are limited to 'AnyTypes', what includes all primitive types (UInt32, Int32, Bool etc.), strings, and Arrays that are compositions of 'AnyTypes'.

Reference

[AdsClient Class](#) [[▸ 183](#)]

[ReadValue Overload](#) [[▸ 302](#)]

[TwinCAT.Ads Namespace](#) [[▸ 179](#)]

6.2.1.2.34.4 AdsClient.ReadValue Method (String, Type)

Reads the value of a symbol and returns the value as object. The parameter type must have the same layout as the ADS symbol.

Namespace: [TwinCAT.Ads](#) [[▸ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Object ReadValue(
    string name,
    Type type
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
type	Type: System.Type Managed type of the ADS symbol.

Return Value

Type: [Object](#)
Value of the symbol

Implements

[IAdsSymbolicAccess.ReadValue\(String, Type\) \[► 1075\]](#)

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [► 67]	

Reference





[AdsClient Class \[► 183\]](#)

[ReadValue Overload \[► 302\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.35 AdsClient.ReadValueAsync Method

Overload List

	Name	Description
	ReadValueAsync(ISymbol, CancellationTok en) [► 306]	Reads the value of a symbol asynchronously and returns it as an object. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. Arrays and structures cannot be read.
	ReadValueAsync.TV alue.(ISymbol, CancellationTok en) [► 307]	Reads the value of a symbol asynchronously and returns it as an object.
	ReadValueAsync.TV alue.(String, CancellationTok en) [► 306]	Reads the value of a symbol asynchronously.
	ReadValueAsync(Str ing, Type, CancellationTok en) [► 308]	Reads the value of a symbol asynchronously.

Reference

[AdsClient Class \[► 183\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.35.1 **AdsClient.ReadValueAsync.TValue. Method (String, CancellationToken)**

Reads the value of a symbol asynchronously.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultValue<TValue>> ReadValueAsync<TValue>(
    string name,
    CancellationToken cancel
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
cancel	Type: System.Threading.CancellationToken The cancel token.

Type Parameters

TValue	The value type.
--------	-----------------

Return Value

Type: [Task.ResultValue](#) [[▶ 1181](#)].TValue..

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [[▶ 1129](#)] parameter contains the read value (Value) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Remarks

The parameter type must have the same layout as the ADS symbol.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[ReadValueAsync Overload](#) [[▶ 305](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.35.2 **AdsClient.ReadValueAsync Method (ISymbol, CancellationToken)**

Reads the value of a symbol asynchronously and returns it as an object. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. Arrays and structures cannot be read.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAnyValue> ReadValueAsync(
    ISymbol symbol,
    CancellationToken cancel
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol that should be read.
cancel	Type: System.Threading.CancellationToken The cancel token.

Return Value

Type: [Task.ResultAnyValue](#) [► 1129].

The [ResultAnyValue](#) [► 1129] as customized task object.

Implements

[IAdsSymbolicAccess.ReadValueAsync\(ISymbol, CancellationToken\)](#) [► 1077]

Reference

[AdsClient Class](#) [► 183]

[ReadValueAsync Overload](#) [► 305]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.2.35.3 **AdsClient.ReadValueAsync.TValue. Method (ISymbol, CancellationToken)**

Reads the value of a symbol asynchronously and returns it as an object.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultValue<TValue>> ReadValueAsync<TValue>(
    ISymbol symbol,
    CancellationToken cancel
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol that should be read.
cancel	Type: System.Threading.CancellationToken The cancel token.

Type Parameters

TValue	The value type.
--------	-----------------

Return Value

Type: [Task.ResultValue](#) [► 1181].TValue..

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [► 1129] parameter contains the read value (Value) and the [ErrorCode](#) [► 1120] after execution.

Exceptions

Exception	Condition
NotImplementedException	

Remarks

Supported types are limited to 'AnyTypes', what includes all primitive types (UInt32, Int32, Bool etc.), strings, and Arrays that are compositions of 'AnyTypes'.

Reference

[AdsClient Class](#) [► 183]

[ReadValueAsync Overload](#) [► 305]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.2.35.4 AdsClient.ReadValueAsync Method (String, Type, CancellationToken)

Reads the value of a symbol asynchronously.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAnyValue> ReadValueAsync (
    string name,
    Type type,
    CancellationToken cancel
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
type	Type: System.Type Managed type of the ADS symbol.
cancel	Type: System.Threading.CancellationToken The cancel token.

Return Value

Type: [Task.ResultAnyValue](#) [► 1129].

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [► 1129] parameter contains the read value (Value) and the [ErrorCode](#) [► 1120] after execution.

Implements

[IAdsSymbolicAccess.ReadValueAsync\(String, Type, CancellationToken\)](#) [► 1079]

Remarks

The parameter type must have the same layout as the ADS symbol.

Reference



[AdsClient Class \[▶ 183\]](#)

[ReadValueAsync Overload \[▶ 305\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.1.2.36 AdsClient.ReadWrite Method

Overload List

	Name	Description
	ReadWrite(UInt32, Memory.Byte, ReadOnlyMemory.Byte.) [▶ 310]	Writes data synchronously to an ADS device and then Reads data from that target.
	ReadWrite(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte.) [▶ 311]	Writes data synchronously to an ADS device and then Reads data from this device into the readBuffer

Reference

[AdsClient Class \[▶ 183\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.1.2.36.1 AdsClient.ReadWrite Method (UInt32, Memory`1, Void)

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934 addedca3e72bc0ea15da1c14

Syntax

C#

```
public int ReadWrite(
    uint variableHandle,
    Memory readBuffer,
    void writeBuffer
)
```

Parameters

- variableHandle Type: [System.UInt32](#)
- readBuffer Type: [Memory](#)
- writeBuffer Type: [System.Void](#)

Return Value

Type: [Int32](#)

Reference

[AdsClient Class \[► 183\]](#)

[ReadWrite Overload \[► 309\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.36.2 AdsClient.ReadWrite Method (UInt32, UInt32, Memory`1, Void)

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public int ReadWrite(  
    uint indexGroup,  
    uint indexOffset,  
    Memory readBuffer,  
    void writeBuffer  
)
```

Parameters

indexGroup Type: [System.UInt32](#)

indexOffset Type: [System.UInt32](#)

readBuffer Type: [Memory](#)

writeBuffer Type: [System.Void](#)

Return Value

Type: [Int32](#)

Reference

[AdsClient Class \[► 183\]](#)

[ReadWrite Overload \[► 309\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.36.3 AdsClient.ReadWrite Method (UInt32, Memory.Byte., ReadOnlyMemory.Byte.)

Writes data synchronously to an ADS device and then Reads data from that target.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ReadWrite(  
    uint variableHandle,  
    Memory<byte> readBuffer,  
    ReadOnlyMemory<byte> writeBuffer  
)
```

Parameters

variableHandle	Type: System.UInt32 Variable handle.
readBuffer	Type: System.Memory.Byte . The read buffer.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.

Return Value

Type: [Int32](#)

Number of successfully returned data bytes.

Implements

[IAdsHandle.ReadWrite\(UInt32, Memory.Byte., ReadOnlyMemory.Byte.\)](#) [[▶ 960](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[ReadWrite Overload](#) [[▶ 309](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[AdsClient.TryReadWrite\(UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.\)](#) [[▶ 346](#)]

[AdsClient.ReadWriteAsync\(UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken\)](#) [[▶ 314](#)]

6.2.1.2.36.4 AdsClient.ReadWrite Method (UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte.)

Writes data synchronously to an ADS device and then Reads data from this device into the readBuffer

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ReadWrite(  
    uint indexGroup,  
    uint indexOffset,  
    Memory<byte> readBuffer,  
    ReadOnlyMemory<byte> writeBuffer  
)
```



Parameters

indexGroup	Type: System.UInt32 The index group number of the requested ADS service.
indexOffset	Type: System.UInt32 The index offset number of the requested ADS service.
readBuffer	Type: System.Memory.Byte . The read buffer.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.

Return ValueType: [Int32](#)

Number of successfully returned (read) data bytes.

Implements[IAdsReadWrite2.ReadWrite\(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte.\)](#) [[▶ 1002](#)]**Reference**[AdsClient Class](#) [[▶ 183](#)][ReadWrite Overload](#) [[▶ 309](#)][TwinCAT.Ads Namespace](#) [[▶ 179](#)]**6.2.1.2.37 AdsClient.ReadWriteAsync Method****Overload List**

	Name	Description
	ReadWriteAsync(UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken) [▶ 314]	ReadWrites value data asynchronously to/from the symbol represented by the variableHandle.
	ReadWriteAsync(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken) [▶ 315]	Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer

Reference[AdsClient Class](#) [[▶ 183](#)][TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.37.1 AdsClient.ReadWriteAsync Method (UInt32, Memory`1, Void, Byte)

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Task<ResultReadWrite> ReadWriteAsync(  
    uint variableHandle,  
    Memory readBuffer,  
    void writeBuffer,  
    byte cancel  
)
```

Parameters

variableHandle	Type: System.UInt32
readBuffer	Type: Memory
writeBuffer	Type: System.Void
cancel	Type: System.Byte

Return Value

Type: [Task.ResultReadWrite](#) [[▶ 1163](#)].

Reference

[AdsClient Class](#) [[▶ 183](#)]

[ReadWriteAsync Overload](#) [[▶ 312](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.37.2 AdsClient.ReadWriteAsync Method (UInt32, UInt32, Memory`1, Void, Byte)

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Task<ResultReadWrite> ReadWriteAsync(  
    uint indexGroup,  
    uint indexOffset,  
    Memory readBuffer,  
    void writeBuffer,  
    byte cancel  
)
```

Parameters

indexGroup	Type: System.UInt32
------------	-------------------------------------

indexOffset	Type: System.UInt32
readBuffer	Type: Memory
writeBuffer	Type: System.Void
cancel	Type: System.Byte

Return Value

Type: [Task.ResultReadWrite](#) [[▶ 1163](#)].

Reference

[AdsClient Class](#) [[▶ 183](#)]

[ReadWriteAsync Overload](#) [[▶ 312](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.37.3 **AdsClient.ReadWriteAsync Method (UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken)**

ReadWrites value data asynchronously to/from the symbol represented by the variableHandle.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultReadWrite> ReadWriteAsync(
    uint variableHandle,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 Variable handle.
readBuffer	Type: System.Memory.Byte . The read data / value
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write data / value.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultReadWrite](#) [[▶ 1163](#)].

A task that represents the asynchronous 'ReadWrite' operation. The [ResultReadWrite](#) [[▶ 1163](#)] parameter contains the total number of bytes read into the buffer ([ReadBytes](#) [[▶ 1145](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsHandle.ReadWriteAsync\(UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken\) \[► 961\]](#)

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [► 67]	

Reference

[AdsClient Class \[► 183\]](#)

[ReadWriteAsync Overload \[► 312\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.37.4 **AdsClient.ReadWriteAsync Method (UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken)**

Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultReadWrite> ReadWriteAsync (
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
readBuffer	Type: System.Memory.Byte. The read buffer.
writeBuffer	Type: System.ReadOnlyMemory.Byte. The write buffer.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultReadWrite \[► 1163\]](#).

A task that represents the asynchronous 'ReadWrite' operation. The [ResultReadWrite \[► 1163\]](#) parameter contains the total number of bytes read into the buffer (ReadBytes) and the [ErrorCode \[► 1120\]](#) after execution.

Implements

[IAdsReadWrite.ReadWriteAsync\(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken\)](#) [[▶ 997](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[ReadWriteAsync Overload](#) [[▶ 312](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.38 AdsClient.RegisterAdsStateChangedAsync Method

Registers for [AdsStateChanged](#) [[▶ 392](#)] events as an asynchronous operation.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAds> RegisterAdsStateChangedAsync (
    EventHandler<AdsStateChangedEventArgs> handler,
    CancellationToken cancel
)
```

Parameters

handler	Type: System.EventHandler.AdsStateChangedEventArgs [▶ 729]. The handler function to be registered for AdsStateChanged calls.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds](#) [[▶ 1116](#)].

A task that represents the asynchronous 'RegisterAdsStateChanged' operation. The [ResultAds](#) [[▶ 1116](#)] parameter contains the state the [ErrorCode](#) [[▶ 1120](#)] of the ADS communication after execution.

Implements

[IAdsStateProvider.RegisterAdsStateChangedAsync\(EventHandler.AdsStateChangedEventArgs., CancellationToken\)](#) [[▶ 1059](#)]

Exceptions

Exception	Condition
ObjectDisposedException	

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.39 **AdsClient.RegisterSymbolVersionChangedAsync Method**

Registers for an [AdsSymbolVersionChanged](#) [▶ 394] event as an asynchronous operation.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAds> RegisterSymbolVersionChangedAsync (
    EventHandler<AdsSymbolVersionChangedEventArgs> handler,
    CancellationToken cancel
)
```

Parameters

handler	Type: System.EventHandler.AdsSymbolVersionChangedEventArgs [▶ 739]. The handler function to register.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds](#) [▶ 1116].

A task that represents the asynchronous 'RegisterSymbolVersionChanged' operation. The [ResultAds](#) [▶ 1116] parameter contains the value [ErrorCode](#) [▶ 1120] of the ADS communication after execution.

Implements

[IAdsSymbolChangedProvider.RegisterSymbolVersionChangedAsync\(EventHandler.AdsSymbolVersionChangedEventArgs, CancellationToken\)](#) [▶ 1062]

Exceptions

Exception	Condition
ObjectDisposedException	


Reference


[AdsClient Class](#) [▶ 183]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.2.40 **AdsClient.TryAddDeviceNotification Method**

Overload List

	Name	Description
	TryAddDeviceNotification(String, Int32, NotificationSettings, Object, UInt32.) [▶ 318]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 390] event.

	Name	Description
	TryAddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.) [▶ 319]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.40.1 **AdsClient.TryAddDeviceNotification Method (String, Int32, NotificationSettings, Object, UInt32.)**

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotification](#) [[▶ 390](#)] event.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryAddDeviceNotification(
    string symbolPath,
    int dataSize,
    NotificationSettings settings,
    Object? userData,
    out uint notificationHandle
)
```

Parameters

symbolPath	Type: System.String The symbol/instance path of the ADS variable.
dataSize	Type: System.Int32 Maximum amount of data in bytes to receive with this ADS Notification.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The notification settings.
userData	Type: System.Object This object can be used to store user specific data.
notificationHandle	Type: System.UInt32 . The notification handle.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The ADS ErrorCode.

Implements

[IAdsNotifications.TryAddDeviceNotification\(String, Int32, NotificationSettings, Object, UInt32.\)](#) [[▶ 985](#)]

Remarks

The
dataSize

Parameter defines the amount of bytes, that will be attached to the [AdsNotification](#) [[▶ 390](#)] as value. Because notifications allocate TwinCAT system resources, a complementary call to [TryDeleteDeviceNotification\(UInt32\)](#) [[▶ 325](#)] should always be called when the notification is not used anymore.

Reference

- [AdsClient Class](#) [[▶ 183](#)]
- [TryAddDeviceNotification Overload](#) [[▶ 317](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]
- [AdsClient.AdsNotification](#) [[▶ 390](#)]
- [AdsClient.TryDeleteDeviceNotification\(UInt32\)](#) [[▶ 325](#)]
- [AddDeviceNotification Overload](#) [[▶ 971](#)]
- [AddDeviceNotificationAsync Overload](#) [[▶ 973](#)]
- [TryAddDeviceNotification Overload](#) [[▶ 985](#)]

6.2.1.2.40.2 AdsClient.TryAddDeviceNotification Method (UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.)

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotification](#) [[▶ 991](#)] event.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryAddDeviceNotification(
    uint indexGroup,
    uint indexOffset,
    int dataSize,
    NotificationSettings settings,
    Object? userData,
    out uint handle
)
```

Parameters

indexGroup	Type: System.UInt32 The index group number of the requested ADS service.
indexOffset	Type: System.UInt32 The index offset number of the requested ADS service.
dataSize	Type: System.Int32 Maximum amount of data in bytes to receive with this ADS Notification.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)

handle	Type: System.UInt32 . The notification handle.
--------	---

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS error code.

Implements

[IAdsNotifications.TryAddDeviceNotification\(UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.\)](#) [[▶ 986](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Remarks

The
dataSize

Parameter defines the amount of bytes, that will be attached to the [AdsNotification](#) [[▶ 390](#)] as value. Because notifications allocate TwinCAT system resources, a complementary call to [TryDeleteDeviceNotification\(UInt32\)](#) [[▶ 990](#)] should always called when the notification is not used anymore.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TryAddDeviceNotification Overload](#) [[▶ 317](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsNotifications.TryDeleteDeviceNotification\(UInt32\)](#) [[▶ 990](#)]

[IAdsNotifications.AdsNotification](#) [[▶ 991](#)]

[IAdsNotifications.AdsNotificationError](#) [[▶ 993](#)]


[AddDeviceNotification Overload](#) [[▶ 971](#)]


[TryAddDeviceNotification Overload](#) [[▶ 985](#)]

[AddDeviceNotificationAsync Overload](#) [[▶ 973](#)]

6.2.1.2.41 AdsClient.TryAddDeviceNotificationEx Method

Overload List

	Name	Description
	TryAddDeviceNotificationEx(String,	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 391] event.

	Name	Description
	NotificationSettings, Object, Type, .Int32., UInt32. [▶ 321]	
	TryAddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., UInt32.) [▶ 322]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.41.1 AdsClient.TryAddDeviceNotificationEx Method (String, NotificationSettings, Object, Type, .Int32., UInt32.)

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotificationEx](#) [[▶ 391](#)] event.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryAddDeviceNotificationEx(
    string symbolPath,
    NotificationSettings settings,
    Object? userData,
    Type type,
    int[]? args,
    out uint notificationHandle
)
```

Parameters

symbolPath	Type: System.String Symbol/Instance path of the ADS variable.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
type	Type: System.Type Type of the object stored in the event argument ('AnyType')
args	Type: .System.Int32. Additional arguments (for 'AnyType')
notificationHandle	Type: System.UInt32. The notification handle

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS error code.

Implements

[IAdsNotifications.TryAddDeviceNotificationEx\(String, NotificationSettings, Object, Type, .Int32., UInt32.\)](#)
[\[► 988\]](#)

Remarks

Because notifications allocate TwinCAT system resources, a complementary call to [TryDeleteDeviceNotification\(UInt32\)](#) [\[► 325\]](#) should always called when the notification is not used anymore.

Reference

[AdsClient Class](#) [\[► 183\]](#)

[TryAddDeviceNotificationEx Overload](#) [\[► 320\]](#)

[TwinCAT.Ads Namespace](#) [\[► 179\]](#)

[AdsClient.AdsNotificationEx](#) [\[► 391\]](#)

[AdsClient.DeleteDeviceNotification\(UInt32\)](#) [\[► 253\]](#)

[AddDeviceNotificationEx Overload](#) [\[► 976\]](#)

[AddDeviceNotificationExAsync Overload](#) [\[► 981\]](#)

[TryAddDeviceNotificationEx Overload](#) [\[► 988\]](#)

6.2.1.2.41.2 **AdsClient.TryAddDeviceNotificationEx Method (UInt32, UInt32, NotificationSettings, Object, Type, .Int32., UInt32.)**

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotificationEx](#) [\[► 993\]](#) event.

Namespace: [TwinCAT.Ads](#) [\[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryAddDeviceNotificationEx(
    uint indexGroup,
    uint indexOffset,
    NotificationSettings settings,
    Object? userData,
    Type anyType,
    int[]? args,
    out uint handle
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
settings	Type: TwinCAT.Ads.NotificationSettings [► 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)

anyType	Type: System.Type Type of the object stored in the event argument ('AnyType')
args	Type: .System.Int32 . The 'AnyType' arguments.
handle	Type: System.UInt32 . The notification handle.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS Error code.

Implements

[IAdsNotifications.TryAddDeviceNotificationEx\(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., UInt32.\)](#) [[▶ 989](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Remarks

If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive types (AnyType) are supported by this method. Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\)](#) [[▶ 983](#)] should always called when the notification is not used anymore.

Reference

- [AdsClient Class](#) [[▶ 183](#)]
- [TryAddDeviceNotificationEx Overload](#) [[▶ 320](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]
- [IAdsNotifications.DeleteDeviceNotification\(UInt32\)](#) [[▶ 983](#)]
- [IAdsNotifications.AdsNotificationEx](#) [[▶ 993](#)]
- [IAdsNotifications.AdsNotificationError](#) [[▶ 993](#)]
- [AddDeviceNotificationEx Overload](#) [[▶ 976](#)]
- [TryAddDeviceNotificationEx Overload](#) [[▶ 988](#)]
- [AddDeviceNotificationExAsync Overload](#) [[▶ 981](#)]

6.2.1.2.42 AdsClient.TryCreateVariableHandle Method

Determines the Symbol handle by its instance path synchronously.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryCreateVariableHandle(
    string symbolPath,
    out uint variableHandle
)
```

Parameters

symbolPath	Type: System.String SymbolName / InstancePath.
variableHandle	Type: System.UInt32 . The symbols handle.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS error code.

Implements

[IAdsHandle.TryCreateVariableHandle\(String, UInt32.\)](#) [[▶ 961](#)]

Remarks

It is a good practice to release all variable handles after use to regain internal resources in the TwinCAT subsystem. The composite method to this [TryCreateVariableHandle\(String, UInt32.\)](#) is the [TryDeleteVariableHandle\(UInt32\)](#) [[▶ 326](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]



[AdsClient.TryDeleteVariableHandle\(UInt32\)](#) [[▶ 326](#)]

[AdsClient.CreateVariableHandleAsync\(String, CancellationToken\)](#) [[▶ 252](#)]

[AdsClient.CreateVariableHandle\(String\)](#) [[▶ 251](#)]

6.2.1.2.43 AdsClient.TryDeleteDeviceNotification Method

Overload List

	Name	Description
	TryDeleteDeviceNotification(UInt32) [▶ 325]	Deletes a registered notification.
	TryDeleteDeviceNotification(UInt32, Int32) [▶ 325]	Deletes a registered notification.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.43.1 **AdsClient.TryDeleteDeviceNotification Method (UInt32)**

Deletes a registered notification.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryDeleteDeviceNotification(  
    uint notificationHandle  
)
```

Parameters

notificationHandle	Type: System.UInt32 Notification handle.
--------------------	---

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The ADS error code.

Implements

[IAdsNotifications.TryDeleteDeviceNotification\(UInt32\)](#) [[▶ 990](#)]

Remarks

This is the complementary method to [TryAddDeviceNotification Overload](#) [[▶ 985](#)] overloads and should be called when the notification is not needed anymore the free TwinCAT realtime resources.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TryDeleteDeviceNotification Overload](#) [[▶ 324](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[AddDeviceNotification Overload](#) [[▶ 971](#)]

[IAdsNotifications.AdsNotification](#) [[▶ 991](#)]

[TryAddDeviceNotification Overload](#) [[▶ 985](#)]

[AddDeviceNotificationAsync Overload](#) [[▶ 973](#)]

6.2.1.2.43.2 **AdsClient.TryDeleteDeviceNotification Method (UInt32, Int32)**

Deletes a registered notification.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryDeleteDeviceNotification(
    uint notificationHandle,
    int timeout
)
```

Parameters

notificationHandle	Type: System.UInt32 Notification handle.
timeout	Type: System.Int32 The timeout.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS error code.

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Remarks

This is the complementary method to [TryAddDeviceNotification Overload](#) [[▶ 985](#)] overloads and should be called when the notification is not needed anymore the free TwinCAT realtime resources.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TryDeleteDeviceNotification Overload](#) [[▶ 324](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[AddDeviceNotification Overload](#) [[▶ 971](#)]

[IAdsNotifications.AdsNotification](#) [[▶ 991](#)]

[TryAddDeviceNotification Overload](#) [[▶ 985](#)]

[AddDeviceNotificationAsync Overload](#) [[▶ 973](#)]

6.2.1.2.44 AdsClient.TryDeleteVariableHandle Method

Releases the specified symbol/variable handle synchronously.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryDeleteVariableHandle(
    uint variableHandle
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable
----------------	---

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS error code.

Implements

[IAdsHandle.TryDeleteVariableHandle\(UInt32\)](#) [[▶ 962](#)]

Remarks

It is a good practice to release all variable handles after use to regain internal resources in the TwinCAT subsystem. The composite method to this TryDeleteVariableHandle(UInt32) is the [TryCreateVariableHandle\(String, UInt32.\)](#) [[▶ 323](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]




[AdsClient.TryCreateVariableHandle\(String, UInt32.\)](#) [[▶ 323](#)]


[AdsClient.DeleteVariableHandleAsync\(UInt32, CancellationToken\)](#) [[▶ 255](#)]

[AdsClient.DeleteVariableHandle\(UInt32\)](#) [[▶ 255](#)]

6.2.1.2.45 **AdsClient.TryInvokeRpcMethod Method**

Overload List

	Name	Description
	TryInvokeRpcMethod(String, String, Object, Object.) [▶ 328]	Invokes the specified RPC Method
	TryInvokeRpcMethod(String, String, Object, Object., Object.) [▶ 330]	Invokes the specified RPC Method
	TryInvokeRpcMethod(IRpcCallableInstance,	Tries the invoke RPC method.

	Name	Description
	IRpcMethod , .Object , .AnyTypeSpecifier " AnyTypeSpecifier , .Object , .Object) [▶ 333]	
	TryInvokeRpcMethod (String , String , .Object , .AnyTypeSpecifier , AnyTypeSpecifier , .Object , .Object) [▶ 331]	Invokes the RPC Method

Reference

[AdsClient Class](#) [▶ 183]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.2.45.1 **AdsClient.TryInvokeRpcMethod Method (String, String, .Object., Object.)**

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryInvokeRpcMethod(
    string symbolPath,
    string methodName,
    Object[]? inParameters,
    out Object?? retValue
)
```

Parameters

symbolPath	Type: System.String The symbol path.
methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
retValue	Type: System.Object . The return value of the RPC method as object.

Return Value

Type: [AdsErrorCode](#) [▶ 664]

The ADS Error Code.

Implements

[IAdsRpcInvoke.TryInvokeRpcMethod\(String, String, Object, Object.\)](#) [[▶ 1026](#)]

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [[▶ 2094](#)] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TryInvokeRpcMethod Overload](#) [[▶ 327](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.45.2 **AdsClient.TryInvokeRpcMethod Method (String, String, .Object., .Object., Object.)**

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryInvokeRpcMethod(
    string symbolPath,
    string methodName,
    Object[]? inParameters,
    out Object[]? outParameters,
    out Object?? retValue
)
```

Parameters

symbolPath	Type: System.String The symbol path.
methodName	Type: System.String The method name.
inParameters	Type: .System.Object. The parameters.
outParameters	Type: .System.Object.. The out parameters.
retValue	Type: System.Object. The return value of the RPC method as object.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The ADS Error Code.

Implements

[IAdsRpcInvoke.TryInvokeRpcMethod\(String, String, .Object., .Object., Object.\)](#) [[▶ 1028](#)]

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [[▶ 2094](#)] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
```

```

/// <param name="args">The arguments.</param>
static void Main(string[] args)
{
// Get the AdsAddress from command-line arguments
AmsAddress address = ArgParser.Parse(args);

using (AdsClient client = new AdsClient())
{
    //client.Synchronize = false;

    // Connect to the target device
    client.Connect(address);

    SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
    ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

    // Get the Symbols (Dynamic Symbols)

    IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
the PLC Program

    // Call a Method that has the following signature (within MAIN Program)
    /* {attribute 'TcRpcEnable'}
    METHOD PUBLIC M_Add : INT
    VAR_INPUT
        i1 : INT := 0;
        i2 : INT := 0;
    END_VAR
    */

    short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

    // Call a Method that has no parameter and returns VOID
    main.InvokeRpcMethod("M_Method1", new object[] {});

    //Browsing RpcMethods
    foreach(IRpcMethod method in main.RpcMethods)
    {
        string methodName = method.Name;

        foreach(IRpcMethodParameter parameter in method.Parameters)
        {
            string parameterName = parameter.Name;
            string parameterType = parameter.TypeName;
        }
    }
}
}
}

```

Reference

[AdsClient Class \[► 183\]](#)

[TryInvokeRpcMethod Overload \[► 327\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.45.3 **AdsClient.TryInvokeRpcMethod Method (String, String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object., Object.)**

Invokes the RPC Method

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryInvokeRpcMethod(
    string symbolPath,
    string methodName,
    Object[]? inParameters,
    AnyTypeSpecifier[]? outSpecifiers,
    AnyTypeSpecifier? retSpecifier,
    out Object[]? outParameters,
    out Object?? retValue
)
```

Parameters

symbolPath	Type: System.String The symbol.
methodName	Type: System.String Name of the method.
inParameters	Type: .System.Object . The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094]. The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094] The ret specifier (specifying the return value) or NULL.
outParameters	Type: .System.Object . The out parameters.
retValue	Type: System.Object . The return value of the RPC method./>

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
AdsErrorCode.

Implements

[IAdsRpcInvoke.TryInvokeRpcMethod\(String, String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object.. Object.\)](#) [[▶ 1030](#)]

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
    }
}
```

```

AmsAddress address = ArgParser.Parse(args);

using (AdsClient client = new AdsClient())
{
    //client.Synchronize = false;

    // Connect to the target device
    client.Connect(address);

    SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
    ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

    // Get the Symbols (Dynamic Symbols)

    IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
the PLC Program

    // Call a Method that has the following signature (within MAIN Program)
    /* {attribute 'TcRpcEnable'}
    METHOD PUBLIC M_Add : INT
    VAR_INPUT
        i1 : INT := 0;
        i2 : INT := 0;
    END_VAR
    */

    short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

    // Call a Method that has no parameter and returns VOID
    main.InvokeRpcMethod("M_Method1", new object[] {});

    //Browsing RpcMethods
    foreach(IRpcMethod method in main.RpcMethods)
    {
        string methodName = method.Name;

        foreach(IRpcMethodParameter parameter in method.Parameters)
        {
            string parameterName = parameter.Name;
            string parameterType = parameter.TypeName;
        }
    }
}
}

```

Reference

[AdsClient Class \[► 183\]](#)

[TryInvokeRpcMethod Overload \[► 327\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.45.4 AdsClient.TryInvokeRpcMethod Method (IRpcCallableInstance, IRpcMethod, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object., Object.)

Tries the invoke RPC method.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public AdsErrorCode TryInvokeRpcMethod(
    IRpcCallableInstance symbol,
    IRpcMethod rpcMethod,
    Object[]? inParameters,
    AnyTypeSpecifier[]? outSpec,
    AnyTypeSpecifier? returnSpec,

```

```

    out Object[]? outParameters,
    out Object?? returnValue
)

```

Parameters

symbol	Type: TwinCAT.TypeSystem.IRpcCallableInstance [▶ 2606] The symbol.
rpcMethod	Type: TwinCAT.TypeSystem.IRpcMethod [▶ 2625] The RPC method.
inParameters	Type: .System.Object . The in parameters.
outSpec	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094]. The out spec.
returnSpec	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094] The return spec.
outParameters	Type: .System.Object .. The out parameters.
returnValue	Type: System.Object . The return value.

Return Value

Type: [AdsErrorCode](#) [▶ 664]
AdsErrorCode.

Implements

[IRpcCallableInstance.InvokeRpcMethod\(IRpcCallableInstance, IRpcMethod, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object., Object.\)](#) [▶ 1031]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	
ArgumentNullException	symbol
ArgumentNullException	rpcMethod

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outParameters, are allowed to be empty or NULL.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```

class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)

```

```

{
// Get the AdsAddress from command-line arguments
AmsAddress address = ArgParser.Parse(args);

using (AdsClient client = new AdsClient())
{
    //client.Synchronize = false;

    // Connect to the target device
    client.Connect(address);

    SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
    ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

    // Get the Symbols (Dynamic Symbols)

    IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
the PLC Program

    // Call a Method that has the following signature (within MAIN Program)
    /* {attribute 'TcRpcEnable'}
    METHOD PUBLIC M_Add : INT
    VAR_INPUT
        i1 : INT := 0;
        i2 : INT := 0;
    END_VAR
    */

    short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

    // Call a Method that has no parameter and returns VOID
    main.InvokeRpcMethod("M_Method1", new object[] {});

    //Browsing RpcMethods
    foreach(IRpcMethod method in main.RpcMethods)
    {
        string methodName = method.Name;

        foreach(IRpcMethodParameter parameter in method.Parameters)
        {
            string parameterName = parameter.Name;
            string parameterType = parameter.TypeName;
        }
    }
}
}

```

Reference



[AdsClient Class \[▶ 183\]](#)

[TryInvokeRpcMethod Overload \[▶ 327\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.1.2.46 AdsClient.TryRead Method

Overload List

	Name	Description
	TryRead(UInt32, Memory.Byte., Int32.) [▶ 337]	Reads the value synchronously data of the symbol, that is represented by the variable handle into the readBuffer.
	TryRead(UInt32, UInt32, Memory.Byte., Int32.) [▶ 338]	Reads value data from the specified IndexGroup/IndexOffset to the specified memory location.

Reference

[AdsClient Class](#) [► 183]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.2.46.1 AdsClient.TryRead Method (UInt32, Memory`1, Void)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public AdsErrorCode TryRead(  
    uint variableHandle,  
    Memory readBuffer,  
    void readBytes  
)
```

Parameters

variableHandle Type: [System.UInt32](#)

readBuffer Type: [Memory](#)

readBytes Type: [System.Void](#)

Return Value

Type: [AdsErrorCode](#) [► 664]

Reference

[AdsClient Class](#) [► 183]

[TryRead Overload](#) [► 335]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.2.46.2 AdsClient.TryRead Method (UInt32, UInt32, Memory`1, Void)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public AdsErrorCode TryRead(  
    uint indexGroup,  
    uint indexOffset,  
    Memory memory,  
    void readBytes  
)
```


Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
memory	Type: Memory
readBytes	Type: System.Void

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TryRead Overload](#) [[▶ 335](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.46.3 AdsClient.TryRead Method (UInt32, Memory.Byte., Int32.)

Reads the value synchronously data of the symbol, that is represented by the variable handle into the readBuffer.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public AdsErrorCode TryRead(
    uint variableHandle,
    Memory<byte> readBuffer,
    out int readBytes
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
readBuffer	Type: System.Memory.Byte. The read buffer/data
readBytes	Type: System.Int32. Number of read bytes.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The ADS error code.

Implements

[IAdsHandle.TryRead\(UInt32, Memory.Byte., Int32.\)](#) [[▶ 963](#)]

Reference[AdsClient Class](#) [[▶ 183](#)][TryRead Overload](#) [[▶ 335](#)][TwinCAT.Ads Namespace](#) [[▶ 179](#)]**6.2.1.2.46.4 AdsClient.TryRead Method (UInt32, UInt32, Memory.Byte., Int32.)**

Reads value data from the specified IndexGroup/IndexOffset to the specified memory location.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public AdsErrorCode TryRead(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> memory,
    out int readBytes
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
memory	Type: System.Memory.Byte. The memory location
readBytes	Type: System.Int32. The read bytes.

Return ValueType: [AdsErrorCode](#) [[▶ 664](#)]

TwinCAT.Ads.AdsErrorCode.

Implements[IAdsReadWrite.TryRead\(UInt32, UInt32, Memory.Byte., Int32.\)](#) [[▶ 998](#)]**Reference**[AdsClient Class](#) [[▶ 183](#)][TryRead Overload](#) [[▶ 335](#)][TwinCAT.Ads Namespace](#) [[▶ 179](#)]**6.2.1.2.47 AdsClient.TryReadDataType Method**

Call this method to obtain information about the specified data type.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryReadDataType(
    string typeName,
    out IDataTypeInfo dataType
)
```

Parameters

typeName	Type: System.String Name of the symbol.
dataType	Type: TwinCAT.TypeSystem.IDataTypeInfo [▶ 2475]. The symbol.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

A [IDataTypeInfo](#) [[▶ 2475](#)] containing the requested symbol information or null if symbol could not be found.

Implements

[IAdsSymbolicAccess.TryReadDataType\(String, IDataTypeInfo.\)](#) [[▶ 1079](#)]

Exceptions

Exception	Condition
ArgumentOutOfRangeException	typeName
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsSymbolicAccess.ReadDataType\(String\)](#) [[▶ 1070](#)]

6.2.1.2.48 **AdsClient.TryReadState Method**

Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryReadState(
    out StateInfo stateInfo
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200]. The ADS statue and device status.
-----------	--

Return Value

Type: [AdsErrorCode](#) [▶ 664]

[AdsErrorCode](#) [▶ 664] of the ADS read state call. Check for [NoError](#) [▶ 664] to see if call was successful.

Implements

[IAdsStateProvider.TryReadState\(StateInfo.\)](#) [▶ 1059]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Remarks

Not all ADS Servers support the State ADS Request

Reference

[AdsClient Class](#) [▶ 183]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.2.49 AdsClient.TryReadSymbol Method

Call this method to obtain information about the individual symbols (variables) in ADS devices.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public AdsErrorCode TryReadSymbol(
    string name,
    out IAdsSymbol?? symbol
)
```

Parameters

name	Type: System.String Name of the symbol.
symbol	Type: TwinCAT.Ads.TypeSystem.IAdsSymbol [▶ 1755]. The symbol.

Return Value

Type: [AdsErrorCode](#) [▶ 664]

A [IAdsSymbol](#) [▶ 1755] containing the requested symbol information or null if symbol could not be found.

Implements

[IAdsSymbolicAccess.TryReadSymbol\(String, IAdsSymbol.\)](#) [[▶ 1080](#)]





Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.50 AdsClient.TryReadValue Method

Overload List

	Name	Description
	TryReadValue(ISymbol, Object.) [▶ 342]	Reads the value of a symbol and returns it as an object.
	TryReadValue.T.(ISymbol, T.) [▶ 343]	Reads the value of a symbol and returns it as an object.
	TryReadValue.T.(String, T.) [▶ 341]	Reads the value of a symbol and returns the value as object.
	TryReadValue(String, Type, Object.) [▶ 344]	Reads the value of a symbol and returns the value as object.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.50.1 AdsClient.TryReadValue.T. Method (String, T.)

Reads the value of a symbol and returns the value as object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryReadValue<T>(
    string name,
    out T value
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
value	Type: T. The read value of the Symbol.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The [AdsErrorCode](#) [[▶ 664](#)].

Implements

[IAdsSymbolicAccess.TryReadValue.T.\(String, T.\)](#) [[▶ 1081](#)]

Exceptions

Exception	Condition
NotImplementedException	

Remarks

The parameter type must have the same layout as the ADS symbol.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TryReadValue Overload](#) [[▶ 341](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.50.2 AdsClient.TryReadValue Method (ISymbol, Object.)

Reads the value of a symbol and returns it as an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public AdsErrorCode TryReadValue(
    ISymbol symbol,
    out Object?? value
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol that should be read.
value	Type: System.Object . The value.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The ADS Error Code

Implements

[IAdsSymbolicAccess.TryReadValue\(ISymbol, Object.\)](#) [[▶ 1082](#)]

Remarks

Supported types are limited to 'AnyTypes', what includes all primitive types (UInt32, Int32, Bool etc.), strings, and Arrays that are compositions of 'AnyTypes'. Structs are not supported.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TryReadValue Overload](#) [[▶ 341](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.50.3 AdsClient.TryReadValue.T. Method (ISymbol, T.)

Reads the value of a symbol and returns it as an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryReadValue<T>(
    ISymbol symbol,
    out T value
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol that should be read.
value	Type: T. The value.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The ADS Error Code

Implements

[IAdsSymbolicAccess.TryReadValue.T.\(ISymbol, T.\)](#) [[▶ 1083](#)]

Remarks

Supported types are limited to 'AnyTypes', what includes all primitive types (UInt32, Int32, Bool etc.), strings, and Arrays that are compositions of 'AnyTypes'.

Reference[AdsClient Class \[► 183\]](#)[TryReadValue Overload \[► 341\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.1.2.50.4 AdsClient.TryReadValue Method (String, Type, Object.)**

Reads the value of a symbol and returns the value as object.

Namespace: [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public AdsErrorCode TryReadValue(
    string name,
    Type type,
    out Object?? value
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
type	Type: System.Type Managed type of the ADS symbol.
value	Type: System.Object . The read value of the Symbol.



Return ValueType: [AdsErrorCode \[► 664\]](#)The [AdsErrorCode \[► 664\]](#).**Implements**[IAdsSymbolicAccess.TryReadValue\(String, Type, Object.\) \[► 1083\]](#)**Remarks**

The parameter type must have the same layout as the ADS symbol.

Reference[AdsClient Class \[► 183\]](#)[TryReadValue Overload \[► 341\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.51 **AdsClient.TryReadWrite Method**

Overload List

	Name	Description
	TryReadWrite(UInt32, Memory.Byte, ReadOnlyMemory.Byte, Int32.) [▶ 346]	ReadWrites value data synchronously to/from the symbol represented by the variableHandle.
	TryReadWrite(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte, Int32.) [▶ 347]	Writes data synchronously to an ADS device and reads data from that device.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.51.1 **AdsClient.TryReadWrite Method (UInt32, Memory`1, Void, Byte)**

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public AdsErrorCode TryReadWrite(
    uint variableHandle,
    Memory readBuffer,
    void writeBuffer,
    byte readBytes
)
```

Parameters

- variableHandle Type: [System.UInt32](#)
- readBuffer Type: [Memory](#)
- writeBuffer Type: [System.Void](#)
- readBytes Type: [System.Byte](#)

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TryReadWrite Overload \[► 345\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.51.2 **AdsClient.TryReadWrite Method (UInt32, UInt32, Memory`1, Void, Byte)**

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public AdsErrorCode TryReadWrite(  
    uint indexGroup,  
    uint indexOffset,  
    Memory readBuffer,  
    void writeBuffer,  
    byte readBytes  
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
readBuffer	Type: Memory
writeBuffer	Type: System.Void
readBytes	Type: System.Byte

Return Value

Type: [AdsErrorCode \[► 664\]](#)

Reference

[AdsClient Class \[► 183\]](#)

[TryReadWrite Overload \[► 345\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.51.3 **AdsClient.TryReadWrite Method (UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.)**

ReadWrites value data synchronously to/from the symbol represented by the variableHandle.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryReadWrite(
    uint variableHandle,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer,
    out int readBytes
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
readBuffer	Type: System.Memory.Byte . The read buffer / read data.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer / write data.
readBytes	Type: System.Int32 . Number of read bytes.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS error code.

Implements

[IAdsHandle.TryReadWrite\(UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.\)](#) [[▶ 963](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TryReadWrite Overload](#) [[▶ 345](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.51.4 AdsClient.TryReadWrite Method (UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.)

Writes data synchronously to an ADS device and reads data from that device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryReadWrite(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer,
    out int readBytes
)
```

Parameters

indexGroup	Type: System.UInt32 The index group number of the requested ADS service.
indexOffset	Type: System.UInt32 The index offset number of the requested ADS service.
readBuffer	Type: System.Memory.Byte . The read buffer.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.
readBytes	Type: System.Int32 . The read bytes.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS Error code.

Implements

[IAdsReadWrite.TryReadWrite\(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.\)](#) [[▶ 998](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TryReadWrite Overload](#) [[▶ 345](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.52 AdsClient.TryResurrect Method

Resurrects the connection

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public bool TryResurrect(
    out AdsException?? error
)
```

Parameters

error	Type: TwinCAT.AdsException [▶ 61]. The error.
-------	--



Return Value

Type: [Boolean](#)
true if resurrection was accepted, false otherwise.

Exceptions

Exception	Condition
ObjectDisposedException	

Reference[AdsClient Class \[► 183\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.1.2.53 AdsClient.TryWrite Method****Overload List**

	Name	Description
	TryWrite(UInt32, ReadOnlyMemory.BYTE) [► 350]	Writes the value data synchronously that is represented in the writeBuffer to the symbol with the specified variableHandle.
	TryWrite(UInt32, UInt32, ReadOnlyMemory.BYTE) [► 351]	Writes data synchronously to an ADS device.

Reference[AdsClient Class \[► 183\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.1.2.53.1 AdsClient.TryWrite Method (UInt32, ReadOnlyMemory`1)****Namespace:** [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934 addedca3e72bc0ea15da1c14

Syntax**C#**

```
public AdsErrorCode TryWrite(
    uint variableHandle,
    ReadOnlyMemory writeBuffer
)
```

ParametersvariableHandle Type: [System.UInt32](#)writeBuffer Type: [ReadOnlyMemory](#)**Return Value**Type: [AdsErrorCode \[► 664\]](#)**Reference**[AdsClient Class \[► 183\]](#)[TryWrite Overload \[► 349\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.53.2 **AdsClient.TryWrite Method (UInt32, UInt32, ReadOnlyMemory`1)**

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934 addedca3e72bc0ea15da1c14

Syntax

C#

```
public AdsErrorCode TryWrite(
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory writeBuffer
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
writeBuffer	Type: ReadOnlyMemory

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TryWrite Overload](#) [[▶ 349](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.53.3 **AdsClient.TryWrite Method (UInt32, ReadOnlyMemory.Byte.)**

Writes the value data synchronously that is represented in the writeBuffer to the symbol with the specified variableHandle.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWrite(
    uint variableHandle,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
writeBuffer	Type: System.ReadOnlyMemory.Byte. The write buffer / value.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
 The ADS error code.

Implements

[IAdsHandle.TryWrite\(UInt32, ReadOnlyMemory.Byte.\)](#) [[▶ 964](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TryWrite Overload](#) [[▶ 349](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.53.4 **AdsClient.TryWrite Method (UInt32, UInt32, ReadOnlyMemory.Byte.)**

Writes data synchronously to an ADS device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWrite(
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
writeBuffer	Type: System.ReadOnlyMemory.Byte. The data buffer to be written.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
 The ADS error code.

Implements

[IAdsReadWrite.TryWrite\(UInt32, UInt32, ReadOnlyMemory.Byte.\)](#) [[▶ 999](#)]

Reference



[AdsClient Class](#) [[▶ 183](#)]

[TryWrite Overload](#) [[▶ 349](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.54 AdsClient.TryWriteControl Method

Overload List

	Name	Description
	TryWriteControl(StateInfo) [▶ 352]	Changes the ADS status and the device status of an ADS server.
	TryWriteControl(StateInfo, ReadOnlyMemory.BYTE) [▶ 353]	Changes the ADS status and the device status of an ADS server.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.54.1 AdsClient.TryWriteControl Method (StateInfo)

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWriteControl(
    StateInfo stateInfo
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200] New ADS status and device status.
-----------	---

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

AdsErrorCode.

Implements

[IAdsStateControl.TryWriteControl\(StateInfo\)](#) [[▶ 1042](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TryWriteControl Overload](#) [[▶ 352](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.54.2 **AdsClient.TryWriteControl Method (StateInfo, ReadOnlyMemory`1)**

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934 addedca3e72bc0ea15da1c14

Syntax

C#

```
public AdsErrorCode TryWriteControl(
    StateInfo stateInfo,
    ReadOnlyMemory writeBuffer
)
```

Parameters

stateInfo Type: [TwinCAT.Ads.StateInfo \[▸ 1200\]](#)

writeBuffer Type: [ReadOnlyMemory](#)

Return Value

Type: [AdsErrorCode \[▸ 664\]](#)

Reference

[AdsClient Class \[▸ 183\]](#)

[TryWriteControl Overload \[▸ 352\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.1.2.54.3 **AdsClient.TryWriteControl Method (StateInfo, ReadOnlyMemory.Byte.)**

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWriteControl(
    StateInfo stateInfo,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▸ 1200] New ADS status and device status.
writeBuffer	Type: System.ReadOnlyMemory.Byte. The write buffer.

Return Value

Type: [AdsErrorCode \[▸ 664\]](#)

AdsErrorCode.

Implements

[IAdsStateControl.TryWriteControl\(StateInfo, ReadOnlyMemory.Byte.\)](#) [▶ 1043]





Reference

[AdsClient Class](#) [▶ 183]

[TryWriteControl Overload](#) [▶ 352]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.2.55 AdsClient.TryWriteValue Method**Overload List**

	Name	Description
	TryWriteValue(ISymbol, Object) [▶ 356]	Writes a value to the symbol. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	TryWriteValue.T.(ISymbol, T) [▶ 356]	Writes a value to the symbol. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	TryWriteValue(String, Object) [▶ 354]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	TryWriteValue.T.(String, T) [▶ 355]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Reference

[AdsClient Class](#) [▶ 183]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.2.55.1 AdsClient.TryWriteValue Method (String, Object)

Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public AdsErrorCode TryWriteValue(
    string name,
    Object value
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
------	--

value	Type: System.Object Object holding the value to be written to the ADS symbol
-------	---

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
AdsErrorCode.

Implements

[IAdsSymbolicAccess.TryWriteValue\(String, Object\)](#) [[▶ 1085](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TryWriteValue Overload](#) [[▶ 354](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.55.2 AdsClient.TryWriteValue.T. Method (String, T)

Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public AdsErrorCode TryWriteValue<T>(
    string name,
    T value
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
value	Type: T Object holding the value to be written to the ADS symbol

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
AdsErrorCode.

Implements

[IAdsSymbolicAccess.TryWriteValue.T.\(String, T\)](#) [[▶ 1085](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TryWriteValue Overload \[► 354\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.55.3 **AdsClient.TryWriteValue Method (ISymbol, Object)**

Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWriteValue (
    ISymbol symbol,
    Object? val
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol the value is written to.
val	Type: System.Object The value to write.

Return Value

Type: [AdsErrorCode \[► 664\]](#)

AdsErrorCode.

Implements

[IAdsSymbolicAccess.TryWriteValue\(ISymbol, Object\) \[► 1086\]](#)

Reference

[AdsClient Class \[► 183\]](#)

[TryWriteValue Overload \[► 354\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.55.4 **AdsClient.TryWriteValue.T. Method (ISymbol, T)**

Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWriteValue<T>(
    ISymbol symbol,
    T val
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol the value is written to.
val	Type: T The value to write.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
AdsErrorCode.

Implements

[IAdsSymbolicAccess.TryWriteValue.T.\(ISymbol, T\)](#) [[▶ 1087](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TryWriteValue Overload](#) [[▶ 354](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.56 AdsClient.UnregisterAdsStateChangedAsync Method

unregister ads state changed as an asynchronous operation.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAds> UnregisterAdsStateChangedAsync (
    EventHandler<AdsStateChangedEventArgs> handler,
    CancellationToken cancel
)
```

Parameters

handler	Type: System.EventHandler.AdsStateChangedEventArgs [▶ 729]. The handler function to be unregistered.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds](#) [► 1116].

A task that represents the asynchronous 'UnregisterAdsStateChanged' operation. The [ResultAds](#) [► 1116] parameter contains the state the [ErrorCode](#) [► 1120] of the ADS communication after execution.

Implements

[IAdsStateProvider.UnregisterAdsStateChangedAsync\(EventHandler.AdsStateChangedEventArgs, CancellationToken\)](#) [► 1060]

Exceptions

Exception	Condition
ObjectDisposedException	

Reference

[AdsClient Class](#) [► 183]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.2.57 AdsClient.UnregisterSymbolVersionChangedAsync Method

Unregisters from an [AdsSymbolVersionChanged](#) [► 394] event as an asynchronous operation.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultAds> UnregisterSymbolVersionChangedAsync(
    EventHandler<AdsSymbolVersionChangedEventArgs> handler,
    CancellationToken cancel
)
```

Parameters

handler	Type: System.EventHandler.AdsSymbolVersionChangedEventArgs [► 739]. The handler function to unregister.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds](#) [► 1116].

A task that represents the asynchronous 'UnregisterSymbolVersionChangedAsync' operation. The [ResultAds](#) [► 1116] parameter contains the value [ErrorCode](#) [► 1120] of the ADS communication after execution.

Implements

[IAdsSymbolChangedProvider.UnregisterSymbolVersionChangedAsync\(EventHandler.AdsSymbolVersionChangedEventArgs, CancellationToken\)](#) [► 1063]

Exceptions

Exception	Condition
ObjectDisposedException	




Reference

[AdsClient Class](#) [[▶](#) 183]

[TwinCAT.Ads Namespace](#) [[▶](#) 179]

6.2.1.2.58 AdsClient.Write Method

Overload List

	Name	Description
	Write(UInt32, ReadOnlyMemory.BYTE.) [▶ 361]	Writes data synchronously to an ADS device.
	Write(UInt32, UInt32) [▶ 360]	Triggers a 'Write' call to the ADS device at the specified address.
	Write(UInt32, UInt32, ReadOnlyMemory.BYTE.) [▶ 361]	Writes data synchronously to an ADS device.

Reference

[AdsClient Class](#) [[▶](#) 183]

[TwinCAT.Ads Namespace](#) [[▶](#) 179]

6.2.1.2.58.1 AdsClient.Write Method (UInt32, ReadOnlyMemory`1)

Namespace: [TwinCAT.Ads](#) [[▶](#) 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public void Write(
    uint variableHandle,
    ReadOnlyMemory writeBuffer
)
```

Parameters

variableHandle Type: [System.UInt32](#)

writeBuffer Type: [ReadOnlyMemory](#)

Reference

[AdsClient Class](#) [[▶](#) 183]

[Write Overload](#) [[▶](#) 359]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.2.58.2 AdsClient.Write Method (UInt32, UInt32)

Triggers a 'Write' call to the ADS device at the specified address.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Write(
    uint indexGroup,
    uint indexOffset
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.

Implements

[IAdsReadWrite2.Write\(UInt32, UInt32\)](#) [▶ 1003]

Reference

[AdsClient Class](#) [▶ 183]

[Write Overload](#) [▶ 359]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.2.58.3 AdsClient.Write Method (UInt32, UInt32, ReadOnlyMemory`1)

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public void Write(
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory writeBuffer
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
writeBuffer	Type: ReadOnlyMemory

Reference[AdsClient Class \[► 183\]](#)[Write Overload \[► 359\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.1.2.58.4 AdsClient.Write Method (UInt32, ReadOnlyMemory.Byte.)**

Writes data synchronously to an ADS device.

Namespace: [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public void Write(
    uint variableHandle,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable
writeBuffer	Type: System.ReadOnlyMemory.Byte. The write buffer / value to be written

Implements[IAdsHandle.Write\(UInt32, ReadOnlyMemory.Byte.\) \[► 964\]](#)**Reference**[AdsClient Class \[► 183\]](#)[Write Overload \[► 359\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)[AdsClient.TryWrite\(UInt32, ReadOnlyMemory.Byte.\) \[► 350\]](#)[AdsClient.WriteAsync\(UInt32, ReadOnlyMemory.Byte., CancellationToken\) \[► 375\]](#)**6.2.1.2.58.5 AdsClient.Write Method (UInt32, UInt32, ReadOnlyMemory.Byte.)**

Writes data synchronously to an ADS device.

Namespace: [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public void Write(
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The data to write.

Implements

[IAdsReadWrite2.Write\(UInt32, UInt32, ReadOnlyMemory.Byte.\)](#) [[▶ 1004](#)]





Reference

[AdsClient Class](#) [[▶ 183](#)]

[Write Overload](#) [[▶ 359](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.59 AdsClient.WriteAny Method**Overload List**

	Name	Description
	WriteAny(UInt32, Object) [▶ 362]	Writes an object synchronously to an ADS device.
	WriteAny(UInt32, Object, .Int32.) [▶ 363]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAny(UInt32, UInt32, Object) [▶ 364]	Writes an object synchronously to an ADS device.
	WriteAny(UInt32, UInt32, Object, .Int32.) [▶ 364]	Writes an object synchronously to an ADS device.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.59.1 AdsClient.WriteAny Method (UInt32, Object)

Writes an object synchronously to an ADS device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteAny(
    uint variableHandle,
    Object value
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
value	Type: System.Object Object to write to the ADS device.

Implements

[IAdsAnyAccess.WriteAny\(UInt32, Object\)](#) [► 829]

Reference

[AdsClient Class](#) [► 183]

[WriteAny Overload](#) [► 362]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.2.59.2 AdsClient.WriteAny Method (UInt32, Object, .Int32.)

Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteAny(
    uint variableHandle,
    Object value,
    int[]? args
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
value	Type: System.Object Object to write to the ADS device.
args	Type: .System.Int32 . Additional arguments.

Implements

[IAdsAnyAccess.WriteAny\(UInt32, Object, .Int32.\)](#) [► 829]

Remarks

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference

[AdsClient Class](#) [[▶ 183](#)]

[WriteAny Overload](#) [[▶ 362](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.59.3 AdsClient.WriteAny Method (UInt32, UInt32, Object)

Writes an object synchronously to an ADS device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public void WriteAny(
    uint indexGroup,
    uint indexOffset,
    Object value
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
value	Type: System.Object Object to write to the ADS device.

Implements

[IAdsAnyAccess.WriteAny\(UInt32, UInt32, Object\)](#) [[▶ 830](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[WriteAny Overload](#) [[▶ 362](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.59.4 AdsClient.WriteAny Method (UInt32, UInt32, Object, .Int32.)

Writes an object synchronously to an ADS device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteAny(
    uint indexGroup,
    uint indexOffset,
    Object value,
    int[]? args
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
value	Type: System.Object Object to write to the ADS device.
args	Type: .System.Int32 . Additional arguments.

Implements

[IAdsAnyAccess.WriteAny\(UInt32, UInt32, Object, .Int32.\)](#) [[▶ 830](#)]

Remarks

If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

Remarks

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference


[AdsClient Class](#) [[▶ 183](#)]




[WriteAny Overload](#) [[▶ 362](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.60 AdsClient.WriteAnyAsync Method

Overload List

	Name	Description
	WriteAnyAsync(UInt32, Object, CancellationToken) [▶ 366]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

	Name	Description
	WriteAnyAsync(UInt32, Object, .Int32., CancellationTokent) [▶ 367]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAnyAsync(UInt32, UInt32, Object, CancellationTokent) [▶ 368]	Write the value of an Anytype (Primitive type) asynchronously.
	WriteAnyAsync(UInt32, UInt32, Object, .Int32., CancellationTokent) [▶ 369]	Write the value of an Anytype (Primitive type) asynchronously.

Reference

[AdsClient Class](#) [▶ 183]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.2.60.1 **AdsClient.WriteAnyAsync Method (UInt32, Object, CancellationTokent)**

Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteAnyAsync(
    uint variableHandle,
    Object value,
    CancellationTokent cancel
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
value	Type: System.Object Object to write to the ADS device.
cancel	Type: System.Threading.CancellationTokent The cancellation token.

Return Value

Type: [Task.ResultWrite](#) [▶ 1187].

A task that represents the asynchronous task operation. The result parameter [ResultWrite](#) [▶ 1187] of the write operation contains the [ErrorCode](#) [▶ 1120].

Implements

[IAdsAnyAccess.WriteAnyAsync\(UInt32, Object, CancellationToken\) \[► 832\]](#)

Remarks

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [► 2927] .
string[]	
Array	

Reference

[AdsClient Class \[► 183\]](#)

[WriteAnyAsync Overload \[► 365\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.60.2 AdsClient.WriteAnyAsync Method (UInt32, Object, .Int32., CancellationToken)

Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteAnyAsync(
    uint variableHandle,
    Object value,
    int[]? args,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
value	Type: System.Object Object to write to the ADS device.
args	Type: .System.Int32 . Additional arguments.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite \[► 1187\]](#).

Task<ResultWrite>.

Return Value

Type: [Task.ResultWrite](#) [▶ 1187].

A task that represents the asynchronous task operation. The result parameter [ResultWrite](#) [▶ 1187] of the write operation contains the [ErrorCode](#) [▶ 1120].

Implements

[IAdsAnyAccess.WriteAnyAsync\(UInt32, Object, .Int32., CancellationToken\)](#) [▶ 833]

Remarks

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference

[AdsClient Class](#) [▶ 183]

[WriteAnyAsync Overload](#) [▶ 365]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.1.2.60.3 AdsClient.WriteAnyAsync Method (UInt32, UInt32, Object, CancellationToken)

Write the value of an Anytype (Primitive type) asynchronously.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteAnyAsync(
    uint indexGroup,
    uint indexOffset,
    Object value,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
value	Type: System.Object The value.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite \[▸ 1187\]](#).

A task that represents the asynchronous 'ReadState' operation. The [ResultWrite \[▸ 1187\]](#) parameter contains the value the [ErrorCode \[▸ 1120\]](#) of the ADS communication after execution.

Implements

[IAdsAnyAccess.WriteAnyAsync\(UInt32, UInt32, Object, CancellationToken\) \[▸ 834\]](#)

Reference

[AdsClient Class \[▸ 183\]](#)

[WriteAnyAsync Overload \[▸ 365\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.1.2.60.4 AdsClient.WriteAnyAsync Method (UInt32, UInt32, Object, .Int32., CancellationToken)

Write the value of an Anytype (Primitive type) asynchronously.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteAnyAsync(
    uint indexGroup,
    uint indexOffset,
    Object value,
    int[]? args,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
value	Type: System.Object The value.
args	Type: .System.Int32 . The type arguments (AnyType)
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite \[▸ 1187\]](#).

A task that represents the asynchronous 'ReadState' operation. The [ResultWrite \[▸ 1187\]](#) parameter contains the value the [ErrorCode \[▸ 1120\]](#) of the ADS communication after execution.

Implements

[IAdsAnyAccess.WriteAnyAsync\(UInt32, UInt32, Object, .Int32., CancellationToken\) \[► 834\]](#)



Reference

[AdsClient Class \[► 183\]](#)

[WriteAnyAsync Overload \[► 365\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.61 AdsClient.WriteAnyStringAsync Method**Overload List**

	Name	Description
	WriteAnyStringAsync(String, String, Int32, Encoding, CancellationToken) [► 370]	Writes the string (Potentially unsafe!)
	WriteAnyStringAsync(UInt32, String, Int32, Encoding, CancellationToken) [► 371]	Writes the string (Potentially unsafe!)

Reference

[AdsClient Class \[► 183\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.61.1 AdsClient.WriteAnyStringAsync Method (String, String, Int32, Encoding, CancellationToken)

Writes the string (Potentially unsafe!)

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultWrite> WriteAnyStringAsync(
    string symbolPath,
    string value,
    int length,
    Encoding encoding,
    CancellationToken cancel
)
```

Parameters

symbolPath	Type: System.String The symbol path.
value	Type: System.String The value.
length	Type: System.Int32 The length of the string to write
encoding	Type: System.Text.Encoding The encoding.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite](#) [[▶ 1187](#)].
Task<ResultWrite>.

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Remarks

ATTENTION: Potentially this method is unsafe because following data can be overwritten after the string symbol. Please be sure to specify the string length lower than the string size reserved within the process image! The String is written with the specified encoding.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[WriteAnyStringAsync Overload](#) [[▶ 370](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.61.2 **AdsClient.WriteAnyStringAsync Method (UInt32, String, Int32, Encoding, CancellationToken)**

Writes the string (Potentially unsafe!)

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteAnyStringAsync(
    uint variableHandle,
    string value,
    int length,
    Encoding encoding,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
value	Type: System.String The value.
length	Type: System.Int32 The length of the string to write
encoding	Type: System.Text.Encoding The encoding.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite](#) [[▶ 1187](#)].
Task<ResultWrite>.

Remarks

ATTENTION: Potentially this method is unsafe because following data can be overwritten after the string symbol. Please be sure to specify the string length lower than the string size reserved within the process image! The String is written with the specified encoding.




Reference

[AdsClient Class](#) [[▶ 183](#)]

[WriteAnyStringAsync Overload](#) [[▶ 370](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.62 AdsClient.WriteAsync Method**Overload List**

	Name	Description
	WriteAsync (UInt32 , ReadOnlyMemory.B yte , CancellationToken) [▶ 375]	Writes the value data asynchronously that is represented by the writeBuffer to the symbol specified by the variableHandle.
	WriteAsync (UInt32 , UInt32 , CancellationToken) [▶ 373]	Triggers a write call at the specified IndexGroup/IndexOffset asynchronously.
	WriteAsync (UInt32 , UInt32 , ReadOnlyMemory.B yte , CancellationToken) [▶ 375]	Writes the data / Value asynchronously into the specified writeBuffer.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.62.1 **AdsClient.WriteAsync Method (UInt32, ReadOnlyMemory`1, Void)**

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Task<ResultWrite> WriteAsync(
    uint variableHandle,
    ReadOnlyMemory writeBuffer,
    void cancel
)
```

Parameters

variableHandle Type: [System.UInt32](#)

writeBuffer Type: [ReadOnlyMemory](#)

cancel Type: [System.Void](#)

Return Value

Type: [Task.ResultWrite \[► 1187\]](#).

Reference

[AdsClient Class \[► 183\]](#)

[WriteAsync Overload \[► 372\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.62.2 **AdsClient.WriteAsync Method (UInt32, UInt32, CancellationToken)**

Triggers a write call at the specified IndexGroup/IndexOffset asynchronously.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteAsync(
    uint indexGroup,
    uint indexOffset,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultWrite](#) [[▶ 1187](#)].

A task that represents the asynchronous 'ReadWrite' operation. The [ResultWrite](#) [[▶ 1187](#)] parameter contains the [ErrorCode](#) [[▶ 1120](#)] after execution.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[WriteAsync Overload](#) [[▶ 372](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.62.3 **AdsClient.WriteAsync Method (UInt32, UInt32, ReadOnlyMemory`1, Void)**

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Task<ResultWrite> WriteAsync(  
    uint indexGroup,  
    uint indexOffset,  
    ReadOnlyMemory writeBuffer,  
    void cancel  
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
writeBuffer	Type: ReadOnlyMemory
cancel	Type: System.Void

Return Value

Type: [Task.ResultWrite](#) [[▶ 1187](#)].

Reference

[AdsClient Class](#) [[▶ 183](#)]

[WriteAsync Overload](#) [[▶ 372](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.62.4 **AdsClient.WriteAsync Method (UInt32, ReadOnlyMemory.Byte., CancellationToken)**

Writes the value data asynchronously that is represented by the writeBuffer to the symbol specified by the variableHandle.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteAsync (
    uint variableHandle,
    ReadOnlyMemory<byte> writeBuffer,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
writeBuffer	Type: System.ReadOnlyMemory.Byte. The write buffer/value.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite](#) [[▶ 1187](#)].

A task that represents the asynchronous write operation. The [ResultWrite](#) [[▶ 1187](#)] parameter contains the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsHandle.WriteAsync\(UInt32, ReadOnlyMemory.Byte., CancellationToken\)](#) [[▶ 965](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[WriteAsync Overload](#) [[▶ 372](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.62.5 **AdsClient.WriteAsync Method (UInt32, UInt32, ReadOnlyMemory.Byte., CancellationToken)**

Writes the data / Value asynchronously into the specified writeBuffer.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteAsync (
    uint indexGroup,
    uint indexOffset,
```

```
ReadOnlyMemory<byte> writeBuffer,  
CancellationTokn cancel  
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite](#) [[▶ 1187](#)].

A task that represents the asynchronous 'Write' operation. The [ResultWrite](#) [[▶ 1187](#)] parameter contains the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsReadWrite.WriteAsync\(UInt32, UInt32, ReadOnlyMemory.Byte., CancellationToken\)](#) [[▶ 1000](#)]

Reference



[AdsClient Class](#) [[▶ 183](#)]

[WriteAsync Overload](#) [[▶ 372](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.63 AdsClient.WriteControl Method

Overload List

	Name	Description
	WriteControl(StateInfo) [▶ 376]	Changes the ADS status and the device status of an ADS server.
	WriteControl(StateInfo, ReadOnlyMemory.Byte.) [▶ 377]	Changes the ADS status and the device status of an ADS server.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.63.1 AdsClient.WriteControl Method (StateInfo)

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteControl(  
    StateInfo stateInfo  
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [► 1200] New ADS status and device status.
-----------	---

Implements

[IAdsStateControl.WriteControl\(StateInfo\)](#) [► 1044]

Reference

[AdsClient Class](#) [► 183]

[WriteControl Overload](#) [► 376]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.2.63.2 **AdsClient.WriteControl Method (StateInfo, ReadOnlyMemory`1)**

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public void WriteControl(  
    StateInfo stateInfo,  
    ReadOnlyMemory writeBuffer  
)
```

Parameters

stateInfo Type: [TwinCAT.Ads.StateInfo](#) [► 1200]

writeBuffer Type: [ReadOnlyMemory](#)

Reference

[AdsClient Class](#) [► 183]

[WriteControl Overload](#) [► 376]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.2.63.3 **AdsClient.WriteControl Method (StateInfo, ReadOnlyMemory.Byte.)**

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteControl(
    StateInfo stateInfo,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200] New ADS status and device status.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.

Implements

[IAdsStateControl.WriteControl\(StateInfo, ReadOnlyMemory.Byte.\)](#) [[▶ 1045](#)]

Reference



[AdsClient Class](#) [[▶ 183](#)]

[WriteControl Overload](#) [[▶ 376](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.64 AdsClient.WriteControlAsync Method

Overload List

	Name	Description
	WriteControlAsync(AdsState, UInt16, CancellationToken) [▶ 379]	Changes the ADS status and device status of the ADS server asynchronously.
	WriteControlAsync(AdsState, UInt16, ReadOnlyMemory.Byte, CancellationToken) [▶ 380]	Writes the state asynchronously

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.64.1 **AdsClient.WriteControlAsync Method (AdsState, UInt16, CancellationToken)**

Changes the ADS status and device status of the ADS server asynchronously.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAds> WriteControlAsync(
    AdsState adsState,
    ushort deviceState,
    CancellationToken cancel
)
```

Parameters

adsState	Type: TwinCAT.Ads.AdsState [▶ 729] The ADS state.
deviceState	Type: System.UInt16 The device state.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds](#) [[▶ 1116](#)].

A task that represents the asynchronous 'WriteControl' operation. The [ResultAds](#) [[▶ 1116](#)] parameter contains the state the [ErrorCode](#) [[▶ 1120](#)] of the ADS communication after execution.

Implements

[IAdsStateControl.WriteControlAsync\(AdsState, UInt16, CancellationToken\)](#) [[▶ 1046](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[WriteControlAsync Overload](#) [[▶ 378](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.64.2 **AdsClient.WriteControlAsync Method (AdsState, UInt16, ReadOnlyMemory<T>, Void)**

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Task<ResultAds> WriteControlAsync(
    AdsState adsState,
    ushort deviceState,
    ReadOnlyMemory<T> writeData,
    void cancel
)
```

Parameters

adsState	Type: TwinCAT.Ads.AdsState [► 729]
deviceState	Type: System.UInt16
writeData	Type: ReadOnlyMemory
cancel	Type: System.Void

Return Value

Type: [Task.ResultAds](#) [► 1116].

Reference

[AdsClient Class](#) [► 183]

[WriteControlAsync Overload](#) [► 378]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.2.64.3 **AdsClient.WriteControlAsync Method (AdsState, UInt16, ReadOnlyMemory.Byte., CancellationToken)**

Writes the state asynchronously

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultAds> WriteControlAsync(
    AdsState adsState,
    ushort deviceState,
    ReadOnlyMemory<byte> writeData,
    CancellationToken cancel
)
```

Parameters

adsState	Type: TwinCAT.Ads.AdsState [► 729] State of the ads.
deviceState	Type: System.UInt16 State of the device.
writeData	Type: System.ReadOnlyMemory.Byte. The write buffer.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds](#) [► 1116].

A task that represents the asynchronous 'ReadState' operation. The [ResultAds](#) [► 1116] parameter contains the [ErrorCode](#) [► 1120] of the ADS communication after execution.

Implements

[IAdsStateControl.WriteControlAsync\(AdsState, UInt16, ReadOnlyMemory.Byte., CancellationToken\) \[▶ 1047\]](#)

Reference

[AdsClient Class \[▶ 183\]](#)

[WriteControlAsync Overload \[▶ 378\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.1.2.65 AdsClient.WriteSymbolAsync Method

Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteSymbolAsync (
    string name,
    Object value,
    CancellationToken cancel
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
value	Type: System.Object Object holding the value to be written to the ADS symbol
cancel	Type: System.Threading.CancellationToken The cancel token.

Return Value

Type: [Task.ResultWrite \[▶ 1187\]](#).

A task that represents the asynchronous 'WriteSymbol' operation. The [ResultWrite \[▶ 1187\]](#) parameter contains the [ErrorCode \[▶ 1120\]](#) after execution.


Reference




[AdsClient Class \[▶ 183\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.1.2.66 AdsClient.WriteValue Method

Overload List

	Name	Description
	WriteValue(ISymbol, Object) [▶ 383]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

	Name	Description
	WriteValue.T. (ISymbol, T) [► 384]	Writes a value to the symbol. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValue(String, Object) [► 382]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	WriteValue.T.(String, T) [► 382]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Reference

[AdsClient Class \[► 183\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.66.1 AdsClient.WriteValue Method (String, Object)

Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteValue(
    string name,
    Object value
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
value	Type: System.Object Object holding the value to be written to the ADS symbol

Implements

[IAdsSymbolicAccess.WriteValue\(String, Object\) \[► 1088\]](#)

Reference

[AdsClient Class \[► 183\]](#)

[WriteValue Overload \[► 381\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.66.2 AdsClient.WriteValue.T. Method (String, T)

Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteValue<T>(
    string name,
    T value
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
value	Type: T Object holding the value to be written to the ADS symbol

Type Parameters

T	the value type.
---	-----------------

Implements

[IAdsSymbolicAccess.WriteValue.T.\(String, T\)](#) [[▶ 1089](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[WriteValue Overload](#) [[▶ 381](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.66.3 AdsClient.WriteValue Method (ISymbol, Object)

Writes a value to the symbol. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteValue(
    ISymbol symbol,
    Object val
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol the value is written to.
val	Type: System.Object The value to write.

Implements

[IAdsSymbolicAccess.WriteValue\(ISymbol, Object\)](#) [[▶ 1089](#)]

Reference[AdsClient Class \[► 183\]](#)[WriteValue Overload \[► 381\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.1.2.66.4 AdsClient.WriteValue.T. Method (ISymbol, T)**

Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

Namespace: [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public void WriteValue<T>(
    ISymbol symbol,
    T val
)
```


Parameters



symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol the value is written to.
val	Type: T The value to write.

Type Parameters

T	The value type.
---	-----------------

Implements[IAdsSymbolicAccess.WriteValue.T.\(ISymbol, T\) \[► 1090\]](#)**Reference**[AdsClient Class \[► 183\]](#)[WriteValue Overload \[► 381\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.1.2.67 AdsClient.WriteValueAsync Method****Overload List**

	Name	Description
	WriteValueAsync(ISymbol, Object, CancellationToken) [► 386]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

	Name	Description
	WriteValueAsync.T. (ISymbol, T, CancellationToken) [▸ 387]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValueAsync.T. (String, T, CancellationToken) [▸ 385]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Reference

[AdsClient Class](#) [[▸ 183](#)]

[TwinCAT.Ads Namespace](#) [[▸ 179](#)]

6.2.1.2.67.1 AdsClient.WriteValueAsync.T. Method (String, T, CancellationToken)

Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Namespace: [TwinCAT.Ads](#) [[▸ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteValueAsync<T>(
    string name,
    T value,
    CancellationToken cancel
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
value	Type: T Object holding the value to be written to the ADS symbol
cancel	Type: System.Threading.CancellationToken The cancel token.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [Task.ResultWrite](#) [[▸ 1187](#)].

A task that represents the asynchronous 'WriteSymbol' operation. The [ResultWrite](#) [[▸ 1187](#)] parameter contains the [ErrorCode](#) [[▸ 1120](#)] after execution.

Implements

[IAdsSymbolicAccess.WriteValueAsync.T.\(String, T, CancellationToken\)](#) [[▸ 1091](#)]

Reference

[AdsClient Class \[► 183\]](#)

[WriteValueAsync Overload \[► 384\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.67.2 AdsClient.WriteValueAsync Method (ISymbol, Object, CancellationToken)

Writes a value to the symbol. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteValueAsync(
    ISymbol symbol,
    Object val,
    CancellationToken cancel
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol the value is written to.
val	Type: System.Object The value to write.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite \[► 1187\]](#).

A task that represents the asynchronous 'WriteSymbol' operation. The [ResultWrite \[► 1187\]](#) parameter contains the [ErrorCode \[► 1120\]](#) after execution.

Implements

[IAdsSymbolicAccess.WriteValueAsync\(ISymbol, Object, CancellationToken\) \[► 1092\]](#)

Reference

[AdsClient Class \[► 183\]](#)

[WriteValueAsync Overload \[► 384\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.2.67.3 **AdsClient.WriteValueAsync.T. Method (ISymbol, T, CancellationToken)**

Writes a value to the symbol. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteValueAsync<T>(
    ISymbol symbol,
    T val,
    CancellationToken cancel
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol the value is written to.
val	Type: T The value to write.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [Task.ResultWrite](#) [[▶ 1187](#)].

A task that represents the asynchronous 'WriteSymbol' operation. The [ResultWrite](#) [[▶ 1187](#)] parameter contains the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsSymbolicAccess.WriteValueAsync.T.\(ISymbol, T, CancellationToken\)](#) [[▶ 1093](#)]

Reference

[AdsClient Class](#) [[▶ 183](#)]

[WriteValueAsync Overload](#) [[▶ 384](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.2.68 **AdsClient.RegisterSymbolVersionChanged Method**

Registers for an [AdsSymbolVersionChanged](#) [[▶ 394](#)] event as an asynchronous operation.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode RegisterSymbolVersionChanged(
    EventHandler<AdsSymbolVersionChangedEventArgs> handler
)
```

Parameters

handler	Type: System.EventHandler.AdsSymbolVersionChangedEventArgs [▶ 739] . The handler function to register.
---------	---

Return Value

Type: [AdsErrorCode \[▶ 664\]](#)

A task that represents the asynchronous 'RegisterSymbolVersionChanged' operation. The [ResultAds \[▶ 1116\]](#) parameter contains the value [ErrorCode \[▶ 1120\]](#) of the ADS communication after execution.

Implements

[IAdsSymbolChangedProvider.RegisterSymbolVersionChanged\(EventHandler.AdsSymbolVersionChangedEventArgs\) \[▶ 1064\]](#)

Exceptions

Exception	Condition
ObjectDisposedException	

Reference

[AdsClient Class \[▶ 183\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.1.2.69 AdsClient.UnregisterSymbolVersionChanged Method

Unregisters the symbol version changed.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode UnregisterSymbolVersionChanged(
    EventHandler<AdsSymbolVersionChangedEventArgs> handler
)
```

Parameters

handler	Type: System.EventHandler.AdsSymbolVersionChangedEventArgs [▶ 739] . The handler function to unregister.
---------	---

Return Value

Type: [AdsErrorCode \[▶ 664\]](#)

AdsErrorCode.

Implements

[IAdsSymbolChangedProvider.UnregisterSymbolVersionChanged\(EventHandler.AdsSymbolVersionChangedEventArgs.\)](#) [[▶ 1064](#)]

Exceptions

Exception	Condition
ObjectDisposedException	

Reference













[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.1.3 AdsClient Events

The [AdsClient](#) [[▶ 183](#)] type exposes the following members.

Events

	Name	Description
 	AdsNotification [▶ 390]	Occurs when the ADS device sends a notification to the client.
	AdsNotificationError [▶ 391]	Occurs when a exception has occurred during notification management.
 	AdsNotificationEx [▶ 391]	Occurs when the ADS devices sends a notification to the client.
	AdsNotificationsInvalidated [▶ 396]	Occurs when Notification Unregistrations / Invalidates are received from the AdsServer
	AdsStateChanged [▶ 392]	Occurs when the ADS state changes.
 	AdsSumNotification [▶ 393]	Occurs when Notifications are send (bundled notifications)
	AdsSymbolVersionChanged [▶ 394]	Occurs when the symbol version has been changed changes.
	ConnectionStateChanged [▶ 395]	Occurs when the connection state has been changed.
	RouterStateChanged [▶ 395]	(Local) Router state changed event.

Reference

[AdsClient Class](#) [[▶ 183](#)]

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.3.1 AdsClient.AdsNotification Event

Occurs when the ADS device sends a notification to the client.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<AdsNotificationEventArgs> AdsNotification
```

Value

Type: [System.EventHandler.AdsNotificationEventArgs \[► 695\]](#).

Implements

[IAdsNotifications.AdsNotification \[► 991\]](#)

Remarks

The Event Argument contains the raw data value of the notification, not marshaled to .NET types.

Examples

Example of receiving AdsNotification events.

Trigger on changed values by ADS Notifications

```
private async Task RegisterNotificationsAsync()
{
    CancellationToken cancel = CancellationToken.None;

    using (AdsClient client = new AdsClient())
    {
        // Add the Notification event handler
        client.AdsNotification += Client_AdsNotification2;

        // Connect to target
        client.Connect(AmsNetId.Local, 851);
        uint notificationHandle = 0;

        // Notification to a DINT Type (UINT32)
        // Check for change every 200 ms

        //byte[] notificationBuffer = new byte[sizeof(UInt32)];
        int size = sizeof(UInt32);

        ResultHandle result = await client.AddDeviceNotificationAsync("MAIN.nCounter", size, new NotificationSettings(AdsTransMode.OnChange, 200, 0), null, cancel);

        if (result.Succeeded)
        {
            notificationHandle = result.Handle;
            await Task.Delay(5000); // Wait asynchronously without blocking the UI Thread.
            // Unregister the Event / Handle
            ResultAds result2 = await client.DeleteDeviceNotificationAsync(notificationHandle, cancel);
        }
        client.AdsNotification -= Client_AdsNotification2;
    }
}

private void Client_AdsNotification2(object sender, AdsNotificationEventArgs e)
{
    // Or here we know about UDINT type --> can be marshalled as UINT32
    uint nCounter = BinaryPrimitives.ReadUInt32LittleEndian(e.Data.Span);
}
```

```
// If Synchronization is needed (e.g. in Windows.Forms or WPF applications)
// we could synchronize via SynchronizationContext into the UI Thread

/*SynchronizationContext syncContext = SynchronizationContext.Current;
   _context.Post(status => someLabel.Text = nCounter.ToString(), null); // Non-blocking post */
}
```

Reference

[AdsClient Class \[► 183\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.3.2 AdsClient.AdsNotificationError Event

Occurs when a exception has occurred during notification management.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<AdsNotificationErrorEventArgs> AdsNotificationError
```

Value

Type: [System.EventHandler.AdsNotificationErrorEventArgs \[► 694\]](#).

Implements

[IAdsNotifications.AdsNotificationError \[► 993\]](#)

Remarks

The occurrence of this event can have two different reasons:

1. Indicates an internal error occurred during Notification management.
2. The registered notification becomes invalid on the server, eg. after a PLC Download / Online Change. If the ADS Server detects that the (still registered) Notification Sender is getting invalid, it sends an error notification so that the client will be informed about detached notifications. The event arguments contains the [AdsInvalidNotificationException \[► 690\]](#) which describes the invalid notification handle by its [Handle \[► 692\]](#) property.

Reference

[AdsClient Class \[► 183\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

[TwinCAT.Ads.AdsInvalidNotificationException \[► 690\]](#)

[AdsClient.AdsNotification \[► 390\]](#)

[AdsClient.AdsNotificationEx \[► 391\]](#)

6.2.1.3.3 AdsClient.AdsNotificationEx Event

Occurs when the ADS devices sends a notification to the client.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<AdsNotificationExEventArgs> AdsNotificationEx
```

Value

Type: [System.EventHandler.AdsNotificationExEventArgs](#) [► 699].

Implements

[IAdsNotifications.AdsNotificationEx](#) [► 993]

Remarks

The Notification event arguments marshals the data value automatically to the specified .NET Type with ANY_TYPE marshallers.

Examples

Example of receiving AdsNotificationEx events.

Trigger on changed values by ADS Notifications

```
CancellationToken cancel = CancellationToken.None;

using (AdsClient client = new AdsClient())
{
    client.AdsNotificationEx += Client_AdsNotificationEx;
    client.Connect(AmsNetId.Local, 851);

    // Add UDINT
    ResultHandle resultHandle = await client.AddDeviceNotificationExAsync("MAIN.udint", new Notifica
tionSettings(AdsTransMode.OnChange, 200, 200), null, typeof(uint), null, cancel);
    await Task.Delay(5000, cancel); // Wait ...
    ResultAds resultHandleDelete = await client.DeleteDeviceNotificationAsync(resultHandle.Handle, ca
ncel); // Unregister Event
}
```

Reference

[AdsClient Class](#) [► 183]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.3.4 AdsClient.AdsStateChanged Event

Occurs when the ADS state changes.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<AdsStateChangedEventArgs> AdsStateChanged
```

Value

Type: [System.EventHandler.AdsStateChangedEventArgs](#) [► 729].

Implements

[IAdsStateProvider.AdsStateChanged](#) [► 1061]

Remarks

This works only for ports that support Notifications (e.g. Port 851 but not Port 10000).

Reference

[AdsClient Class](#) [► 183]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.3.5 AdsClient.AdsSumNotification Event

Occurs when Notifications are send (bundled notifications)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<AdsSumNotificationEventArgs> AdsSumNotification
```

Value

Type: [System.EventHandler.AdsSumNotificationEventArgs](#).

Implements

[IAdsNotifications.AdsSumNotification](#) [► 994]

Remarks

As an optimization, this event receives all ADS Notifications that occurred at one point in time together. As consequence, the overhead of handler code is reduced, what can be important if notifications are triggered in a high frequency and the event has to be synchronized to the UI thread context. Because multiple notifications are bound together, less thread synchronization is necessary. The [AdsNotification](#) [► 390] and [AdsNotificationEx](#) [► 391] events shouldn't be used when SumNotifications are registered, because they have an performance side effect to this AdsSumNotification event. The full performance is reached only, when all notifications are handled on this event.

Examples

Example of receiving AdsSumNotification events.

Trigger on changed values by ADS Notifications

```
private async Task RegisterSumNotificationsAsync()  
{  
    CancellationToken cancel = CancellationToken.None;  
  
    using (AdsClient client = new AdsClient())  
    {  
        // Add the Notification event handler  
        client.AdsSumNotification += Client_SumNotification;  
  
        // Connect to target  
        client.Connect(AmsNetId.Local, 851);  
        uint notificationHandle = 0;
```

```

// Notification to a DINT Type (UINT32)
// Check for change every 200 ms

//byte[] notificationBuffer = new byte[sizeof(UInt32)];

ResultHandle result = await client.AddDeviceNotificationAsync("MAIN.nCounter", sizeof(UInt32), new NotificationSettings(AdsTransMode.OnChange, 200, 0), null, cancel);

if (result.Succeeded)
{
    notificationHandle = result.Handle;
    await Task.Delay(5000); // Wait asynchronously without blocking the UI Thread.
    // Unregister the Event / Handle
    ResultAds result2 = await client.DeleteDeviceNotificationAsync(notificationHandle, cancel);
}
client.AdsNotification -= Client_AdsNotification2;
}
}

private void Client_SumNotification(object sender, AdsSumNotificationEventArgs e)
{
    // Timestamp of the Notification List
    DateTimeOffset dateTime = e.TimeStamp;

    // List of Raw ADS Notifications
    IList<Notification> notifications = e.Notifications;

    foreach(Notification notification in notifications)
    {
        // Notifications can be handled more efficiently, because they occur together
        // handler and can be transformed/
        // synchronized in one step compared to AdsClient.AdsNotification events.
    }
}

```

Reference

[AdsClient Class \[► 183\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

[AdsClient.AdsNotification \[► 390\]](#)

6.2.1.3.6 AdsClient.AdsSymbolVersionChanged Event

Occurs when the symbol version has been changed changes.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<AdsSymbolVersionChangedEventArgs> AdsSymbolVersionChanged
```

Value

Type: [System.EventHandler.AdsSymbolVersionChangedEventArgs \[► 739\]](#).

Implements

[IAdsSymbolChangedProvider.AdsSymbolVersionChanged \[► 1065\]](#)

Remarks

This is the case when the connected ADS server restarts. This invalidates all actual opened symbol handles. The SymbolVersion counter doesn't trigger, when an online change is made on the PLC (ports 801, ..., 851 ...)

Reference[AdsClient Class \[► 183\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.1.3.7 AdsClient.ConnectionStateChanged Event**

Occurs when the connection state has been changed.

Namespace: [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public event EventHandler<ConnectionStateChangedEventArgs> ConnectionStateChanged
```

Value

Type: [System.EventHandler.ConnectionStateChangedEventArgs \[► 74\]](#).

Implements[IConnectionStateProvider.ConnectionStateChanged \[► 91\]](#)**Reference**[AdsClient Class \[► 183\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.1.3.8 AdsClient.RouterStateChanged Event**

(Local) Router state changed event.

Namespace: [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public event EventHandler<AmsRouterNotificationEventArgs> RouterStateChanged
```

Value

Type: [System.EventHandler.AmsRouterNotificationEventArgs \[► 798\]](#).

Implements[IRouterNotificationProvider.RouterStateChanged \[► 1100\]](#)**Exceptions**

Exception	Condition
ObjectDisposedException	

Remarks

This event indicates, that a changed event is received from the Local AmsRouter independant of the connected target address. A remote system RouterStateChanged event cannot be received at another system - it cannot traverse TwinCAT systems.

Reference

[AdsClient Class \[► 183\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.3.9 AdsClient.AdsNotificationsInvalidated Event

Occurs when Notification Unregistrations / Invalidates are received from the AdsServer

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public event EventHandler<AdsNotificationsInvalidatedEventArgs> AdsNotificationsInvalidated
```

Value

Type: [System.EventHandler.AdsNotificationsInvalidatedEventArgs \[► 1228\]](#).

Remarks

Some ADS servers are sending 0-size notifications, when the Notification handle is not valid anymore. If received, this event will be triggered, to notify any consumers to invalidate the notification handles. One example for these sort of invalidation is, if ADS Notifications are already registered at the PLC ADS Server, and the PLC Control downloads a new program. All registered notification handles are invalidated!

Reference

[AdsClient Class \[► 183\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.4 AdsClient Constructor**Overload List**

	Name	Description
	AdsClient. [► 397]	Initializes a new instance of the AdsClient [► 183] class.
	AdsClient(AdsClient Settings) [► 397]	Initializes a new instance of the AdsClient [► 183] class with the specified settings.
	AdsClient(ILogger) [► 397]	Initializes a new instance of the AdsClient [► 183] class.
	AdsClient(ISession, ILogger) [► 398]	Initializes a new instance of the AdsClient [► 183] class bound to a session.
	AdsClient(ISession, AdsClientSettings, ILogger) [► 398]	Initializes a new instance of the AdsClient [► 183] class.

Reference

[AdsClient Class](#) [► 183]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.4.1 AdsClient Constructor

Initializes a new instance of the [AdsClient](#) [► 183] class.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsClient()
```

Reference

[AdsClient Class](#) [► 183]

[AdsClient Overload](#) [► 396]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.4.2 AdsClient Constructor (AdsClientSettings)

Initializes a new instance of the [AdsClient](#) [► 183] class with the specified settings.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsClient(  
    AdsClientSettings settings  
)
```

Parameters

settings	Type: TwinCAT.Ads.AdsClientSettings [► 399] The settings.
----------	--

Reference

[AdsClient Class](#) [► 183]

[AdsClient Overload](#) [► 396]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.1.4.3 AdsClient Constructor (ILogger)

Initializes a new instance of the [AdsClient](#) [► 183] class.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsClient(
    ILogger? logger
)
```

Parameters

logger	Type: Microsoft.Extensions.Logging.ILogger The logger.
--------	---

Reference

[AdsClient Class \[► 183\]](#)

[AdsClient Overload \[► 396\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.4.4 AdsClient Constructor (ISession, ILogger)

Initializes a new instance of the [AdsClient \[► 183\]](#) class bound to a session.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsClient(
    ISession session,
    ILogger? logger
)
```

Parameters

session	Type: TwinCAT.ISession [► 92] The session.
logger	Type: Microsoft.Extensions.Logging.ILogger The logger.

Reference

[AdsClient Class \[► 183\]](#)

[AdsClient Overload \[► 396\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.1.4.5 AdsClient Constructor (ISession, AdsClientSettings, ILogger)

Initializes a new instance of the [AdsClient \[► 183\]](#) class.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsClient(
    ISession? session,
    AdsClientSettings settings,
    ILogger? logger
)
```

Parameters

session	Type: TwinCAT.ISession [▶ 92] The session.
settings	Type: TwinCAT.Ads.AdsClientSettings [▶ 399] The settings.
logger	Type: Microsoft.Extensions.Logging.ILogger The logger.

Reference

[AdsClient Class](#) [▶ 183]

[AdsClient Overload](#) [▶ 396]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.2 AdsClientSettings Class

Settings object for the [AdsClient](#) [▶ 183] class.

Inheritance Hierarchy

[System.Object](#)

TwinCAT.Ads.AdsClientSettings

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229


Syntax

C#



```
public class AdsClientSettings
```





The AdsClientSettings type exposes the following members.

Constructors







	Name	Description
	AdsClientSettings [▶ 400]	Creates a Default settings AdsClientSettings object with custom timeout.

Properties

	Name	Description
	CompatibilityDefault t [▶ 401]	Compatibility settings object
	Default [▶ 402]	Gets the default settings (Default interceptors, Timeout 5000 ms)

	Name	Description
		
	FastWriteThrough [▶ 402]	Gets a Settings object that configures the AdsClient for FastWriteThrough
		
	Timeout [▶ 403]	The communication Timeout that is set initially on the AdsClient [▶ 183]

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Remarks

This [AdsClientSettings](#) object is used to initialize the [AdsClient](#) [▶ 183] with application appropriate settings. Several predefined application dependant settings are available as static properties:

- [Default](#) [▶ 402]
- [FastWriteThrough](#) [▶ 402]
- [CompatibilityDefault](#) [▶ 401]

Reference

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.2.1 AdsClientSettings Constructor

Creates a Default settings [AdsClientSettings](#) [▶ 399] object with custom timeout.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: [TwinCAT.Ads](#) (in [TwinCAT.Ads.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsClientSettings(
    int timeout
)
```

Parameters

timeout	Type: System.Int32 The timeout of the AdsClient [▶ 183] in milliseconds.
---------	---

Reference





[AdsClientSettings Class \[▶ 399\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.2.2 AdsClientSettings Properties

The [AdsClientSettings \[▶ 399\]](#) type exposes the following members.

Properties

	Name	Description
 S	CompatibilityDefault [▶ 401]	Compatibility settings object
 S	Default [▶ 402]	Gets the default settings (Default interceptors, Timeout 5000 ms)
 S	FastWriteThrough [▶ 402]	Gets a Settings object that configures the AdsClient for FastWriteThrough
	Timeout [▶ 403]	The communication Timeout that is set initially on the AdsClient [▶ 183]

Reference

[AdsClientSettings Class \[▶ 399\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.2.2.1 AdsClientSettings.CompatibilityDefault Property

Compatibility settings object

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AdsClientSettings CompatibilityDefault { get; }
```

Property Value

Type: [AdsClientSettings \[▶ 399\]](#)

The settings object.

Remarks

The compatibility settings initialize the AdsClient the same way as it is done in earlier versions of the TwinCAT.Ads.dll (earlier than Version 4.2)

- [All \[▶ 1212\]](#)
- No FailFastHandlerInterceptor active.
- Default communicationtimeout 5000ms.
- Synchronized Notifications.

Reference

[AdsClientSettings Class \[► 399\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.2.2.2 AdsClientSettings.Default Property

Gets the default settings (Default interceptors, Timeout 5000 ms)

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AdsClientSettings Default { get; }
```

Property Value

Type: [AdsClientSettings \[► 399\]](#)

The default.

Remarks

Creates an settings object, with specification for [All \[► 1212\]](#) and [FailFastHandlerInterceptor](#).

- [All \[► 1212\]](#)
- [FailFastHandlerInterceptor](#) is active.
- Default communication timeout 5000ms.
- Not synchronized Notifications.

Reference

[AdsClientSettings Class \[► 399\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.2.2.3 AdsClientSettings.FastWriteThrough Property

Gets a Settings object that configures the AdsClient for FastWriteThrough

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AdsClientSettings FastWriteThrough { get; }
```

Property Value

Type: [AdsClientSettings \[► 399\]](#)

Client settings for a fast write through (with 200 ms Timeout).

Remarks

The settings typically can be used for polling clients, where the "FailFast" feature will be bypassed. That means, that communication fails doesn't trigger the FailFast interceptor and every Request will go out via ADS. This has the Drawback that communication Timeouts are longer and subsequent timeouts block the ADS mailbox (with the danger of overflows). So use this setting with care for specific purposes and should not be used for standard communication.

- No FailFastHandlerInterceptor active.
- Default communicationtimeout 200ms.
- Not synchronized Notifications.

Reference

[AdsClientSettings Class](#) [► 399]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.2.4 AdsClientSettings.Timeout Property

The communication Timeout that is set initially on the [AdsClient](#) [► 183]

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Timeout { get; set; }
```

Property Value

Type: [Int32](#)
The timeout.

Remarks

Setting the Settings Timeout doesn't change the IConnection timeout after the connection is established.

Reference




[AdsClientSettings Class](#) [► 399]




[TwinCAT.Ads Namespace](#) [► 179]

6.2.2.3 AdsClientSettings Methods

The [AdsClientSettings](#) [► 399] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)

	Name	Description
	GetType	Gets the <u>Type</u> of the current instance. (Inherited from <u>Object</u> .)
	MemberwiseClone	Creates a shallow copy of the current <u>Object</u> . (Inherited from <u>Object</u> .)
	ToString	Returns a string that represents the current object. (Inherited from <u>Object</u> .)

Reference

[AdsClientSettings Class](#) [[▶ 399](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.3 AdsCommandId Enumeration

AdsCommandId Enumeration

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum AdsCommandId
```

Members

	Member name	Value	Description
	None	0	None / Uninitialized
	Invalid	0	Invalid
	ReadDeviceInfo	1	ReadDeviceInfo command
	Read	2	Read Command
	Write	3	Write Command
	ReadState	4	ReadState Command
	WriteControl	5	WriteControl Command
	AddNotification	6	AddNotification Command
	DeleteNotification	7	DeleteNotification Command
	Notification	8	Notification event.
	ReadWrite	9	ReadWrite Command

Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.4 AdsCommunicationStatistics Class

ADS Communication statistics

Inheritance Hierarchy

[System.Object](#)

 TwinCAT.Ads.AdsCommunicationStatistics

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14













Syntax

C#





```
public class AdsCommunicationStatistics
```



The AdsCommunicationStatistics type exposes the following members.

Properties

	Name	Description
	AccessWaitTime [▶ 407]	Gets the wait time for the next access (Resurrection time) if in Lost [▶ 72].
	ConnectionActiveSi nce [▶ 407]	Gets the UTC time of the last conenction activation.
	ConnectionEstablish edAt [▶ 407]	Gets the UTC time when the current connection was established.
	ConnectionLostCou nt [▶ 408]	Gets the connection lost count.
	ConnectionLostTim e [▶ 408]	Gets the UTC connection lost time.
	ConnectionResurrec tions [▶ 409]	Gets the number of resurrections on the AdsConnection [▶ 412]
	ErrorsSinceLastSucc eeded [▶ 409]	Gets the error count since last access (UTC)
	LastSucceededAcce ss [▶ 410]	Gets the UTC time of the last succeeded access.
	Resurrections [▶ 410]	Gets the number of Resurrections of this Session.
	SessionEstablishedA t [▶ 410]	Gets the UTC time when the session was established.
	TotalCycles [▶ 411]	Gets the total cycles.
	TotalErrors [▶ 411]	Gets the total error count.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)

	Name	Description
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Remarks

These statistics can be used for communication diagnosis. They contain Error/Succeed counts as well as Resurrection infos.













Reference

[TwinCAT.Ads Namespace](#) [► 179]

6.2.4.1 AdsCommunicationStatistics Properties

The [AdsCommunicationStatistics](#) [► 404] type exposes the following members.

Properties

	Name	Description
	AccessWaitTime [► 407]	Gets the wait time for the next access (Resurrection time) if in Lost [► 72].
	ConnectionActiveSi nce [► 407]	Gets the UTC time of the last conenction activation.
	ConnectionEstablish edAt [► 407]	Gets the UTC time when the current connection was established.
	ConnectionLostCou nt [► 408]	Gets the connection lost count.
	ConnectionLostTim e [► 408]	Gets the UTC connection lost time.
	ConnectionResurrec tions [► 409]	Gets the number of resurrections on the AdsConnection [► 412]
	ErrorsSinceLastSucc eeded [► 409]	Gets the error count since last access (UTC)
	LastSucceededAcce ss [► 410]	Gets the UTC time of the last succeeded access.
	Resurrections [► 410]	Gets the number of Resurrections of this Session.
	SessionEstablishedA t [► 410]	Gets the UTC time when the session was established.
	TotalCycles [► 411]	Gets the total cycles.
	TotalErrors [► 411]	Gets the total error count.

Reference

[AdsCommunicationStatistics Class](#) [► 404]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.4.1.1 AdsCommunicationStatistics.AccessWaitTime Property

Gets the wait time for the next access (Resurrection time) if in [Lost](#) [► 72].

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public TimeSpan AccessWaitTime { get; }
```

Property Value

Type: [TimeSpan](#)

The wait time if in [Lost](#) [► 72] otherwise **TimeSpan.Zero**.

Reference

[AdsCommunicationStatistics Class](#) [► 404]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.4.1.2 AdsCommunicationStatistics.ConnectionActiveSince Property

Gets the UTC time of the last connection activation.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Nullable<DateTimeOffset> ConnectionActiveSince { get; }
```

Property Value

Type: [Nullable.DateTimeOffset](#).

Connection active time.

Reference

[AdsCommunicationStatistics Class](#) [► 404]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.4.1.3 AdsCommunicationStatistics.ConnectionEstablishedAt Property

Gets the UTC time when the current connection was established.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Nullable<DateTimeOffset> ConnectionEstablishedAt { get; }
```

Property Value

Type: [Nullable.DateTimeOffset](#).

The connection established at.

Reference

[AdsCommunicationStatistics Class](#) [[▶ 404](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.4.1.4 AdsCommunicationStatistics.ConnectionLostCount Property

Gets the connection lost count.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public int ConnectionLostCount { get; }
```

Property Value

Type: [Int32](#)

The connection lost count.

Reference

[AdsCommunicationStatistics Class](#) [[▶ 404](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.4.1.5 AdsCommunicationStatistics.ConnectionLostTime Property

Gets the UTC connection lost time.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Nullable<DateTimeOffset> ConnectionLostTime { get; }
```


Property Value

Type: [Nullable.DateTimeOffset](#).
The connection lost time.

Reference

[AdsCommunicationStatistics Class](#) [► 404]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.4.1.6 **AdsCommunicationStatistics.ConnectionResurrections** Property

Gets the number of resurrections on the [AdsConnection](#) [► 412]

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:
5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public int ConnectionResurrections { get; }
```

Property Value

Type: [Int32](#)
The resurrections.

Reference

[AdsCommunicationStatistics Class](#) [► 404]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.4.1.7 **AdsCommunicationStatistics.ErrorsSinceLastSucceeded** Property

Gets the error count since last access (UTC)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:
5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public int ErrorsSinceLastSucceeded { get; }
```

Property Value

Type: [Int32](#)
The error count since last access.

Reference

[AdsCommunicationStatistics Class](#) [► 404]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.4.1.8 **AdsCommunicationStatistics.LastSucceededAccess** Property

Gets the UTC time of the last succeeded access.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:
5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Nullable<DateTimeOffset> LastSucceededAccess { get; }
```

Property Value

Type: [Nullable.DateTimeOffset](#).
The last succeeded access.

Reference

[AdsCommunicationStatistics Class](#) [[▶ 404](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.4.1.9 **AdsCommunicationStatistics.Resurrections** Property

Gets the number of Resurrections of this Session.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:
5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public int Resurrections { get; }
```

Property Value

Type: [Int32](#)
The resurrections.

Reference

[AdsCommunicationStatistics Class](#) [[▶ 404](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.4.1.10 **AdsCommunicationStatistics.SessionEstablishedAt** Property

Gets the UTC time when the session was established.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:
5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public DateTimeOffset SessionEstablishedAt { get; }
```

Property Value

Type: [DateTimeOffset](#)

The session established at.

Reference

[AdsCommunicationStatistics Class](#) [[▶ 404](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.4.1.11 **AdsCommunicationStatistics.TotalCycles Property**

Gets the total cycles.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:
5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public int TotalCycles { get; }
```

Property Value

Type: [Int32](#)

The total cycles.

Reference

[AdsCommunicationStatistics Class](#) [[▶ 404](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.4.1.12 **AdsCommunicationStatistics.TotalErrors Property**

Gets the total error count.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:
5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public int TotalErrors { get; }
```

Property Value

Type: [Int32](#)

The total error count.

Reference







[AdsCommunicationStatistics Class](#) [[▶ 404](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.4.2 AdsCommunicationStatistics Methods

The [AdsCommunicationStatistics](#) [► 404] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[AdsCommunicationStatistics Class](#) [► 404]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5 AdsConnection Class

ADS Connection class

Inheritance Hierarchy

[System.Object](#)

 TwinCAT.Ads.AdsConnection

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229





Syntax

















C#

```
public sealed class AdsConnection : IAdsConnection,
    IConnection, IConnectionStateProvider, IAdsNotifications, IAdsSymbolicAccess, IAdsAnyAccess,
    IAdsHandle, IAdsReadWrite2, IAdsReadWrite, IAdsStateProvider, IAdsStateControl,
    IAdsSymbolChangedProvider, IAdsRpcInvoke, IAdsReadWriteTimeoutAccess, IAdsStateControlTimeout, I
    Disposable
```


The [AdsConnection](#) type exposes the following members.




Properties


















	Name	Description
	AccessWaitTime [► 435]	Gets the access wait time.
	ActiveSince [► 435]	Gets the UTC time when the last active/resurrected Connection was established
	Address [► 436]	Gets the AmsAddress [► 752] of the ADS server.
	ClientAddress [► 436]	Get the AmsAddress [► 752] of the ADS client.







	Name	Description
	ConnectionEstablishedAt [▶ 437]	Gets the UTC time when the Connection was originally established.
	ConnectionLostTime [▶ 438]	Gets the connection lost time.
 	ConnectionState [▶ 438]	Gets the current Connection state of the AdsConnection
	DefaultValueEncoding [▶ 439]	Gets the default value encoding.
	Disposed [▶ 440]	Gets a value indicating whether this AdsConnection is disposed.
	Id [▶ 440]	Gets the AdsConnection identifier.
	IsActive [▶ 441]	Gets a value indicating whether communication is in active state
	IsConnected [▶ 441]	Gets a value indicating whether the local ADS port was opened successfully. It does not indicate if the target port is available. Use the method ReadState to determine if the target port is available.
	IsLocal [▶ 441]	Gets a value indicating whether the ADS client is connected to a ADS Server on the local computer.
	IsLost [▶ 442]	Gets a value indicating whether the communication is in lost / open state
	IsReconnecting [▶ 442]	Gets a value indicating whether communication is ready for reconnecting
	Name [▶ 443]	Gets the name of this AdsConnection.
	Session [▶ 444]	Gets the Session object of the AdsConnection object.
	State [▶ 444]	Gets the current ConnectionState [▶ 438]
	Timeout [▶ 445]	Gets the timeout (in milliseconds)
	TotalConnectionLosses [▶ 445]	Gets the connection lost count.
	TotalResurrectingTries [▶ 446]	Gets the number of tries to resurrect the AdsConnection.
	TotalResurrections [▶ 446]	Gets the number of succeeded connection resurrections.

Methods














	Name	Description
	AddDeviceNotification(String, Int32, NotificationSettings, Object) [▶ 467]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 652] event.

















	Name	Description
	AddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object) [▶ 468]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event.
	AddDeviceNotificationAsync(String, Int32, NotificationSettings, Object, CancellationToken) [▶ 470]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 991] event.
	AddDeviceNotificationAsync(UInt32, UInt32, Int32, NotificationSettings, Object, CancellationToken) [▶ 471]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 652] event.
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type) [▶ 473]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 654] event.
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32.) [▶ 474]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 654] event.
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type) [▶ 475]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 654] event.
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32.) [▶ 476]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive ('AnyType') types are allowed for the parameter type.
	AddDeviceNotificationExAsync(String, NotificationSettings,	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 654] event.











	Name	Description
	Object, Type, .Int32, CancellationToken [▶ 478]	
	AddDeviceNotificationExAsync(UInt32, UInt32, NotificationSettings, Object, Type, .Int32, CancellationToken) [▶ 479]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 654] event.
	CleanupSymbolTable [▶ 480]	Clears the internal symbol cache.
	Close [▶ 481]	Closes the AdsConnection
	Connect [▶ 481]	(Re)Connects the IConnection [▶ 79] when disconnected.
	CreateVariableHandle [▶ 493]	Generates a unique handle for an ADS variable.
	CreateVariableHandleAsync [▶ 494]	Determines the Symbol handle by its instance path asynchronously.
	DeleteDeviceNotification [▶ 495]	Deletes an existing notification.
	DeleteDeviceNotificationAsync [▶ 495]	Deletes a registered notification asynchronously.
	DeleteVariableHandle [▶ 496]	Releases the handle of a ADS variable again.
	DeleteVariableHandleAsync [▶ 497]	Releases the handle of a ADS variable again (asynchronously)
	Disconnect [▶ 498]	Disconnects this IConnection [▶ 79].
	Dispose [▶ 498]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
 	InvokeRpcMethod(String, String, .Object.) [▶ 499]	Invokes the specified RPC Method
















	Name	Description
 	InvokeRpcMethod(String, Object, Object.) [▶ 501]	Invokes the specified RPC Method
 	InvokeRpcMethod(String, Object, AnyTypeSpecifier) [▶ 504]	Invokes the specified RPC Method
 	InvokeRpcMethod(String, Object, AnyTypeSpecifier, AnyTypeSpecifier, Object..) [▶ 502]	Invokes the specified RPC Method
 	InvokeRpcMethodAsync(String, Object, CancellationToken) [▶ 506]	Invokes the specified RPC Method asynchronously
 	InvokeRpcMethodAsync(IRpcCallableInstance, IRpcMethod, Object, AnyTypeSpecifier, AnyTypeSpecifier, CancellationToken) [▶ 510]	Invokes the specified RPC Method asynchronously
 	InvokeRpcMethodAsync(String, Object, AnyTypeSpecifier, AnyTypeSpecifier, CancellationToken) [▶ 508]	Invokes the specified RPC Method asynchronously
	Read(UInt32, Memory.Byte.) [▶ 514]	Reads the value from the symbol that is represented by the handle.
	Read(UInt32, UInt32, Memory.Byte.) [▶ 515]	Reads data synchronously from an ADS device and writes it to the given readBuffer
	Read(UInt32, UInt32, Memory.Byte., Int32.) [▶ 516]	Reads data synchronously from an ADS device and writes it to the given stream.














	Name	Description
	ReadAny(UInt32, Type) [▶ 519]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, Type, .Int32.) [▶ 520]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type) [▶ 522]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type, .Int32.) [▶ 523]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type, .Int32., Int32) [▶ 524]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32) [▶ 517]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, .Int32.) [▶ 518]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, UInt32) [▶ 519]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, UInt32, .Int32.) [▶ 521]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, Type, CancellationToken) [▶ 528]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, Type, .Int32., CancellationToken) [▶ 529]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, UInt32, Type, CancellationToken) [▶ 532]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, UInt32, Type, .Int32., CancellationToken) [▶ 532]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T.(UInt32, CancellationToken) [▶ 526]	Reads data synchronously from an ADS device.















	Name	Description
	ReadAnyAsync.T. (UInt32, Int32, CancellationToken) [▶ 527]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T. (UInt32, UInt32, CancellationToken) [▶ 529]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T. (UInt32, UInt32, Int32, CancellationToken) [▶ 531]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyString(UInt 32, Int32, Encoding) [▶ 534]	Reads a string from the specified symbol/variable.
	ReadAnyString(UInt 32, UInt32, Int32, Encoding) [▶ 535]	Reads as string from a specified address.
	ReadAnyStringAsyn c(UInt32, Int32, Encoding, CancellationToken) [▶ 536]	Reads a string asynchronously from the specified symbol/variable
	ReadAnyStringAsyn c(UInt32, UInt32, Int32, Encoding, CancellationToken) [▶ 537]	read any string as an asynchronous operation.
	ReadAsync(UInt32, Memory.Byte., CancellationToken) [▶ 540]	Reads the value data of the symbol asynchronously into the readBuffer.
	ReadAsync(UInt32, UInt32, Memory.Byte., CancellationToken) [▶ 541]	Reads the data asynchronously from specified IndexGroup/IndexOffset
	ReadDataType [▶ 541]	Call this method to obtain information about the specified data type.
	ReadDataTypeAsyn c [▶ 542]	read data type as an asynchronous operation.
	ReadDeviceInfo [▶ 543]	Reads the identification and version number of an ADS server.
	ReadDeviceInfoAsyn c [▶ 544]	Reads the identification and version number of an ADS server.













	Name	Description
	ReadState. [▶ 545]	Reads the ADS status and the device status from an ADS server.
	ReadState(Int32) ▶ 546	Reads the ADS status and the device status from an ADS server.
	ReadStateAsync ▶ 546	Read the ADS State asynchronously
	ReadSymbol. [▶ 547]	Call this method to obtain information about the individual symbols (variables) in ADS devices.
	ReadSymbolAsync ▶ 548	Call this method to obtain information about the individual symbols (variables) in ADS devices.
	ReadValue(ISymbol) ▶ 549	Reads the value of a symbol and returns it as an object. Strings and all primitive datatypes (UInt32, Int32, Bool etc.) are supported. Arrays and structures cannot be read.
	ReadValue(String, Type) [▶ 551]	Reads the value of a symbol and returns the value as object. The parameter type must have the same layout as the ADS symbol.
	ReadValue.T. (ISymbol) [▶ 550]	Reads the value of a symbol and returns it as an object.
	ReadValue.T.(String) ▶ 549	Reads the value of a symbol and returns the value. The parameter type must have the same layout as the ADS symbol.
	ReadValueAsync(ISymbol, CancellationToken) ▶ 553	Reads the value of a symbol and returns it as an object. Strings and all primitive datatypes (UInt32, Int32, Bool etc.) are supported. Arrays and structures cannot be read.
	ReadValueAsync(String, Type, CancellationToken) ▶ 555	Reads the value of a symbol asynchronously.
	ReadValueAsync.T. (ISymbol, CancellationToken) ▶ 554	Reads the value of a symbol asynchronously and returns it as an object.
	ReadValueAsync.T. (String, CancellationToken) ▶ 552	Reads the value of a symbol asynchronously.
	ReadWrite(UInt32, Memory.Byte, ReadOnlyMemory.Byte.) [▶ 558]	Writes data synchronously to an ADS device and then Reads data from that target.
	ReadWrite(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte.) [▶ 559]	Writes data synchronously to an ADS device and then Reads data from this device into the readBuffer
	ReadWrite(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte, Int32) [▶ 560]	Writes data synchronously to an ADS device and then Reads data from this device.












	Name	Description
	ReadWriteAsync(UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken) [▶ 563]	ReadWrites value data asynchronously to/from the symbol represented by the variableHandle.
	ReadWriteAsync(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken) [▶ 564]	Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer
	RegisterAdsStateChangedAsync [▶ 565]	Registers for AdsStateChanged [▶ 655] events as an asynchronous operation.
	RegisterSymbolVersionChanged [▶ 650]	Registers the symbol version changed.
	RegisterSymbolVersionChangedAsync [▶ 565]	Registers the symbol version changed asynchronously.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryAddDeviceNotification(String, Int32, NotificationSettings, Object, UInt32.) [▶ 567]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 652] event.
	TryAddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.) [▶ 568]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 652] event.
	TryAddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32., UInt32.) [▶ 570]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event.
	TryAddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., UInt32.) [▶ 571]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 654] event.








	Name	Description
	TryCreateVariableHandle [▶ 572]	Read (determine) the Symbol handle by its name/path
	TryDeleteDeviceNotification [▶ 572]	Deletes a registered notification.
	TryDeleteVariableHandle [▶ 573]	Releases the specified symbol/variable handle synchronously.
 	TryInvokeRpcMethod(String, String, .Object, Object.) [▶ 575]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(String, String, .Object, .Object, Object.) [▶ 577]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(IRpcCallableInstance, IRpcMethod, .Object, .AnyTypeSpecifier, .AnyTypeSpecifier, .Object, Object.) [▶ 580]	Invokes the rpc method.
 	TryInvokeRpcMethod(String, String, .Object, .AnyTypeSpecifier, .AnyTypeSpecifier, .Object, Object.) [▶ 578]	Invokes the rpc method.
	TryRead(UInt32, Memory.Byte, Int32.) [▶ 585]	Reads the value synchronously data of the symbol, that is represented by the variable handle into the readBuffer.
	TryRead(UInt32, UInt32, Memory.Byte, Int32.) [▶ 586]	Reads data synchronously from an ADS device and writes it to the given stream.
	TryRead(UInt32, UInt32, Memory.Byte, Int32, Int32.) [▶ 586]	Reads data synchronously from an ADS device and writes it to the given stream.
	TryReadDataType [▶ 587]	Call this method to obtain information about the specified data type.

	Name	Description
	TryReadState(StateInfo.) [▶ 589]	Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful.
	TryReadState(Int32, StateInfo.) [▶ 589]	Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful.
	TryReadSymbol [▶ 590]	Call this method to obtain information about the individual symbols (variables) in ADS devices.
	TryReadValue(ISymbol, Object.) [▶ 592]	Reads the value of a symbol and returns the value as object. The parameter type must have the same layout as the ADS symbol.
	TryReadValue(String, Type, Object.) [▶ 594]	Reads the value of a symbol and returns the value as object.
	TryReadValue.T.(ISymbol, T.) [▶ 593]	Reads the value of a symbol and returns it as an object.
	TryReadValue.T.(String, T.) [▶ 591]	Reads the value of a symbol and returns the value as object.
	TryReadWrite(UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.) [▶ 597]	ReadWrites value data synchronously to/from the symbol represented by the variableHandle.
	TryReadWrite(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.) [▶ 598]	Writes data synchronously to an ADS device and reads data from that device.
	TryReadWrite(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32, Int32.) [▶ 599]	Writes data synchronously to an ADS device and then Reads data from this device.
	TryWrite(UInt32, ReadOnlyMemory.Byte.) [▶ 603]	Writes the value data synchronously that is represented in the writeBuffer to the symbol with the specified variableHandle.
	TryWrite(UInt32, UInt32, ReadOnlyMemory.Byte.) [▶ 604]	Writes data synchronously to an ADS device.
	TryWrite(UInt32, UInt32, ReadOnlyMemory.Byte., Int32) [▶ 604]	Writes data synchronously to an ADS device.












	Name	Description
	TryWriteControl(StationInfo) [▶ 606]	Changes the ADS status and the device status of an ADS server.
	TryWriteControl(StationInfo, Int32) [▶ 606]	Changes the ADS status and the device status of an ADS server.
	TryWriteControl(StationInfo, ReadOnlyMemory.Byte) [▶ 608]	Changes the ADS status and the device status of an ADS server.
	TryWriteControl(StationInfo, ReadOnlyMemory.Byte, Int32) [▶ 609]	Changes the ADS status and the device status of an ADS server.
	TryWriteValue(ISymbol, Object) [▶ 612]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	TryWriteValue(String, Object) [▶ 611]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	TryWriteValue.T.(ISymbol, T) [▶ 613]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	TryWriteValue.T.(String, T) [▶ 611]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	UnregisterAdsStateChangedAsync [▶ 614]	Registers for AdsStateChanged [▶ 655] events as an asynchronous operation.
	UnregisterSymbolVersionChanged [▶ 651]	Unregisters the symbol version changed.
	UnregisterSymbolVersionChangedAsync [▶ 614]	Unregisters the symbol version changed asynchronous.
	Write(UInt32, ReadOnlyMemory.Byte) [▶ 619]	Writes data synchronously to an ADS device.
	Write(UInt32, UInt32) [▶ 616]	Trigger Client Method/Command.
	Write(UInt32, UInt32, Int32) [▶ 617]	Trigger Client Method/Command.


	Name	Description
	<u>Write(UInt32, UInt32, ReadOnlyMemory.BYTE)</u> [▶ 619]	Writes data synchronously to an ADS device.
	<u>Write(UInt32, UInt32, ReadOnlyMemory.BYTE, Int32)</u> [▶ 620]	Writes data synchronously to an ADS device.
	<u>WriteAny(UInt32, Object)</u> [▶ 621]	Writes an object synchronously to an ADS device.
	<u>WriteAny(UInt32, Object, .Int32.)</u> [▶ 622]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	<u>WriteAny(UInt32, UInt32, Object)</u> [▶ 623]	Writes an object synchronously to an ADS device.
	<u>WriteAny(UInt32, UInt32, Object, .Int32.)</u> [▶ 623]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	<u>WriteAny(UInt32, UInt32, Object, .Int32., Int32)</u> [▶ 624]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	<u>WriteAnyAsync(UInt32, Object, CancellationToken)</u> [▶ 626]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	<u>WriteAnyAsync(UInt32, Object, .Int32., CancellationToken)</u> [▶ 627]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	<u>WriteAnyAsync(UInt32, UInt32, Object, CancellationToken)</u> [▶ 628]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	<u>WriteAnyAsync(UInt32, UInt32, Object, .Int32., CancellationToken)</u> [▶ 628]	write any as an asynchronous operation.
	<u>WriteAnyStringAsync(String, String, Int32, Encoding, CancellationToken)</u> [▶ 630]	write any string as an asynchronous operation.

	Name	Description
	<u>WriteAnyStringAsync</u> (UInt32, String, Int32, Encoding, CancellationToken) [▶ 631]	write any string as an asynchronous operation.
	<u>WriteAsync</u> (UInt32, ReadOnlyMemory.Byte, CancellationToken) [▶ 634]	Writes the value data asynchronously that is represented by the writeBuffer to the symbol specified by the variableHandle.
	<u>WriteAsync</u> (UInt32, UInt32, CancellationToken) [▶ 633]	Triggers a write call at the specified IndexGroup/IndexOffset asynchronously.
	<u>WriteAsync</u> (UInt32, UInt32, ReadOnlyMemory.Byte, CancellationToken) [▶ 635]	Writes the data / Value asynchronously into the specified writeBuffer.
	<u>WriteControl</u> (StateInfo) [▶ 637]	Changes the ADS status and the device status of an ADS server.
	<u>WriteControl</u> (StateInfo, Int32) [▶ 637]	Changes the ADS status and the device status of an ADS server.
	<u>WriteControl</u> (StateInfo, ReadOnlyMemory.Byte) [▶ 639]	Changes the ADS status and the device status of an ADS server.
	<u>WriteControl</u> (StateInfo, ReadOnlyMemory.Byte, Int32) [▶ 640]	Changes the ADS status and the device status of an ADS server.
	<u>WriteControlAsync</u> (AdsState, UInt16, CancellationToken) [▶ 641]	Changes the ADS status and device status of the ADS server asynchronously.
	<u>WriteControlAsync</u> (AdsState, UInt16, ReadOnlyMemory.Byte, CancellationToken) [▶ 642]	Writes the AdsState [▶ 729] and device state to the ADS device.
	<u>WriteSymbolAsync</u> [▶ 643]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

	Name	Description
	WriteValue(ISymbol, Object) [▶ 646]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValue(String, Object) [▶ 644]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	WriteValue.T. (ISymbol, T) [▶ 646]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValue.T.(String, T) [▶ 645]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	WriteValueAsync(ISymbol, Object, CancellationToken) [▶ 648]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValueAsync.T. (ISymbol, T, CancellationToken) [▶ 649]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValueAsync.T. (String, T, CancellationToken) [▶ 648]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Events












	Name	Description
 	AdsNotification [▶ 652]	Occurs when the ADS device sends a notification to the client.
	AdsNotificationError [▶ 654]	Occurs when a exception has occurred during notification management.
 	AdsNotificationEx [▶ 654]	Occurs when the ADS devices sends a notification to the client.
	AdsStateChanged [▶ 655]	Occurs when ADS State has been changed.
 	AdsSumNotification [▶ 656]	Occurs when Notifications are send (bundled notifications)
	AdsSymbolVersionC hanged [▶ 657]	Occurs when the symbol version has been changed.
 	ConnectionStateCha nged [▶ 658]	Occurs when connection status of the AdsConnection has been changed.
















	Name	Description
	RouterStateChanged [▶ 659]	(Local) Router state changed event.










Extension Methods













	Name	Description
	PollAdsState(IObservable.Unit.) [▶ 3294]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
	PollAdsState(IObservable.Unit.) [▶ 1256]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState(TimeSpan) [▶ 3295]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
	PollAdsState(TimeSpan) [▶ 1257]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState2(IObservable.Unit.) [▶ 1271]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState2(TimeSpan) [▶ 1272]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState2Async(IObservable.Unit., Cancellation.Token) [▶ 1274]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState2Async(TimeSpan, Cancellation.Token) [▶ 1275]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsStateAsync(IObservable.Unit., Cancellation.Token) [▶ 3296]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
	PollAdsStateAsync(IObservable.Unit., Cancellation.Token) [▶ 1259]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)

	Name	Description
	PollAdsStateAsync(TimeSpan, CancellationToken) [▶ 3298]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
	PollAdsStateAsync(TimeSpan, CancellationToken) [▶ 1260]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollDeviceState(IObservable.Unit.) [▶ 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollDeviceState(TimeSpan) [▶ 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollDeviceStateAsync(IObservable.Unit., CancellationToken) [▶ 1278]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollDeviceStateAsync(TimeSpan, CancellationToken) [▶ 1279]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollSystemServiceState [▶ 3282]	Polls the state of the system service. (Defined by SystemServiceExtension [▶ 3281].)
	PollSystemServiceStateAsync [▶ 3283]	Polls the system service state asynchronously (Defined by SystemServiceExtension [▶ 3281].)
	PollValues(ISymbol, Type, IObservable.Unit.) [▶ 1310]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, TimeSpan) [▶ 1311]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, IObservable.Unit.) [▶ 1300]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, TimeSpan) [▶ 1301]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32., TimeSpan) [▶ 1316]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	PollValues(ISymbol, Type, IObservable.Unit, Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, TimeSpan, Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32, TimeSpan) [▶ 1304]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, IObservable.Unit, Func.Exception, Object.) [▶ 1305]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, TimeSpan, Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32, IObservable.Unit, Func.Exception, Object.) [▶ 1321]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32, TimeSpan, Func.Exception, Object.) [▶ 1322]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32, IObservable.Unit, Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32, TimeSpan, Func.Exception, Object.) [▶ 1307]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, IObservable.Unit) [▶ 1308]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, TimeSpan) [▶ 1309]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	<code>PollValues.T.(String, IObservable.Unit.)</code> [▶ 1294]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	<code>PollValues.T.(String, TimeSpan)</code> [▶ 1295]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	<code>PollValues.T.(ISymbol, IObservable.Unit, Func.Exception, T.)</code> [▶ 1314]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	<code>PollValues.T.(ISymbol, TimeSpan, Func.Exception, T.)</code> [▶ 1315]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	<code>PollValues.T.(ISymbol, .Int32, IObservable.Unit.)</code> [▶ 1312]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	<code>PollValues.T.(ISymbol, .Int32, TimeSpan)</code> [▶ 1313]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	<code>PollValues.T.(String, IObservable.Unit, Func.Exception, T.)</code> [▶ 1298]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	<code>PollValues.T.(String, TimeSpan, Func.Exception, T.)</code> [▶ 1299]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	<code>PollValues.T.(String, .Int32, IObservable.Unit.)</code> [▶ 1296]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	<code>PollValues.T.(String, .Int32, TimeSpan)</code> [▶ 1297]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	<code>PollValues.T.(ISymbol, .Int32, IObservable.Unit, Func.Exception, T.)</code> [▶ 1318]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	<code>PollValues.T.(ISymbol, .Int32, TimeSpan, Func.Exception, T.)</code> [▶ 1319]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	<code>PollValues.T.</code> (String, .Int32., IObservable.Unit., Func.Exception, T.) [▶ 1302]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	<code>PollValues.T.</code> (String, .Int32., TimeSpan, Func.Exception, T.) [▶ 1303]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	<code>PollValues2.T.</code> (ISymbol, .Int32., IObservable.Unit.) [▶ 1330]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	<code>PollValues2.T.</code> (String, .Int32., IObservable.Unit.) [▶ 1329]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	<code>ReadSysServState</code> [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (Defined by SystemServiceExtension [▶ 3281].)
	<code>ReadSysServStateAs ync</code> [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (asynchronous) (Defined by SystemServiceExtension [▶ 3281].)
	<code>ReadWithFallback(U Int32, UInt32, Memory.Byte., UInt32, Boolean.)</code> [▶ 1219]	Overloaded. Ads Read with fallback. (Defined by AdsClientExtensions [▶ 1215].)
	<code>ReadWithFallback(U Int32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., Boolean.)</code> [▶ 1220]	Overloaded. Ads Read with Fallback. (Defined by AdsClientExtensions [▶ 1215].)
	<code>ReadWithFallbackAs ync(UInt32, UInt32, UInt32, Memory.Byte., CancellationToken)</code> [▶ 1221]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	<code>ReadWithFallbackAs ync(UInt32, UInt32, UInt32, Memory.Byte., Func.ResultRead,</code>	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)

	Name	Description
	Boolean., CancellationToken [▶ 1222]	
	RepeatedRead(UInt32, UInt32, Memory.Byte., Int32, TimeSpan) [▶ 1224]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedRead(UInt32, UInt32, Memory.Byte., Int32, TimeSpan, Func.ResultRead, Boolean.) [▶ 1225]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedReadAsync(UInt32, UInt32, Memory.Byte., Int32, TimeSpan, CancellationToken) [▶ 1226]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedReadAsync(UInt32, UInt32, Memory.Byte., Int32, TimeSpan, Func.ResultRead, Boolean., CancellationToken) [▶ 1227]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	RestartTwinCATAsync [▶ 3285]	Sends a SystemService Restart to the target system and waits until the system has finished the state changes (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [▶ 3281].)
	SetAdsState [▶ 3299]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [▶ 3292].)
	SetAdsStateAsync [▶ 3300]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [▶ 3292].)
	WaitUntilRestarted [▶ 3286]	Waits until the Restart is detected on the client (SystemService, Port 10000) (Defined by SystemServiceExtension [▶ 3281].)
	WaitUntilRestartedAsync [▶ 3287]	Waits until the SystemService Restart is detected on the client (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [▶ 3281].)
 	WhenAdsStateChanges [▶ 1261]	Gets an observable sequence of AdsState [▶ 729]s. (Defined by AdsClientExtensions [▶ 1249].)
	WhenNotification(ISymbol) [▶ 1263]	Overloaded. Gets an observable sequence of Notification [▶ 1100]s. (Defined by AdsClientExtensions [▶ 1249].)

	Name	Description
 	WhenNotification(ISymbolCollection) [▶ 1264]	Overloaded. Gets an observable sequence of Notification [▶ 1100] objects. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenNotification(IList.ISymbol, NotificationSettings) [▶ 1268]	Overloaded. Gets an observable sequence of Notification [▶ 1100] objects. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenNotification(ISymbol, NotificationSettings) [▶ 1265]	Overloaded. Gets an observable sequence of SymbolValueNotification [▶ 1331]s. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenNotification(String, Type, NotificationSettings) [▶ 1324]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [▶ 1281].)
 	WhenNotification.T.(String, NotificationSettings) [▶ 1323]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [▶ 1281].)
 	WhenSymbolVersionChanges . [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenSymbolVersionChanges(IScheduler) [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenValueChanged [▶ 1345]	Observable sequence of Values driven by ADS Notifications on the specified symbol. (Defined by ValueSymbolExtensions [▶ 1333].)
 	WriteValues.T.(String, IObservable.T.) [▶ 1326]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)
 	WriteValues.T.(String, IObservable.T., Action.Exception.) [▶ 1327]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)

Remarks

The ADS Connection class represents an ADS Point-to-Point Connection between client and server. It is established by using the Connect method of the [AdsSession](#) [[▶ 701](#)] object. An ADS Connection can have different [ConnectionStates](#) [[▶ 438](#)], which represent the state of the logical ADS connection.

Reference


[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[TwinCAT.Ads.AdsSession](#) [[▶ 701](#)]

[TwinCAT.Ads.IAdsConnection](#) [[▶ 876](#)]

[System.IDisposable](#)









Also see about this







-  [AdsClientExtensions.WhenNotification Method \(IAdsConnection, ISymbolCollection, NotificationSettings\)](#) [[▶ 1266](#)]

6.2.5.1 AdsConnection Properties

The [AdsConnection](#) [[▶ 412](#)] type exposes the following members.

Properties

	Name	Description
	AccessWaitTime [▶ 435]	Gets the access wait time.
	ActiveSince [▶ 435]	Gets the UTC time when the last active/resurrected Connection was established
	Address [▶ 436]	Gets the AmsAddress [▶ 752] of the ADS server.
	ClientAddress [▶ 436]	Get the AmsAddress [▶ 752] of the ADS client.
	ConnectionEstablishedAt [▶ 437]	Gets the UTC time when the Connection was originally established.
	ConnectionLostTime [▶ 438]	Gets the connection lost time.
 	ConnectionState [▶ 438]	Gets the current Connection state of the AdsConnection [▶ 412]
	DefaultValueEncoding [▶ 439]	Gets the default value encoding.
	Disposed [▶ 440]	Gets a value indicating whether this AdsConnection [▶ 412] is disposed.
	Id [▶ 440]	Gets the AdsConnection [▶ 412] identifier.
	IsActive [▶ 441]	Gets a value indicating whether communication is in active state
	IsConnected [▶ 441]	Gets a value indicating whether the local ADS port was opened successfully. It does not indicate if the target port is available. Use the method ReadState to determine if the target port is available.
	IsLocal [▶ 441]	Gets a value indicating whether the ADS client is connected to a ADS Server on the local computer.
	IsLost [▶ 442]	Gets a value indicating whether the communication is in lost / open state
	IsReconnecting [▶ 442]	Gets a value indicating whether communication is ready for reconnecting
	Name [▶ 443]	Gets the name of this AdsConnection [▶ 412].

	Name	Description
	Session [▶ 444]	Gets the Session object of the AdsConnection [▶ 412] object.
	State [▶ 444]	Gets the current ConnectionState [▶ 438]
	Timeout [▶ 445]	Gets the timeout (in milliseconds)
	TotalConnectionLosses [▶ 445]	Gets the connection lost count.
	TotalResurrectingTries [▶ 446]	Gets the number of tries to resurrect the AdsConnection [▶ 412].
	TotalResurrections [▶ 446]	Gets the number of succeeded connection resurrections.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.1.1 **AdsConnection.AccessWaitTime** Property

Gets the access wait time.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public TimeSpan AccessWaitTime { get; }
```

Property Value

Type: [TimeSpan](#)

The access wait time.

Remarks

Gets the Wait Time until the next communication try will be done. This time is calculated as follows:
ResurrectionTime - (DateTime.Now - ConnectionLostTime)

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[AdsConnection.ConnectionLostTime](#) [[▶ 438](#)]

[SessionSettings.ResurrectionTime](#) [[▶ 1197](#)]

6.2.5.1.2 **AdsConnection.ActiveSince** Property

Gets the UTC time when the last active/resurrected Connection was established

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTimeOffset? ActiveSince { get; }
```

Property Value

Type: [Nullable.DateTimeOffset](#).

The active since.

Reference

[AdsConnection Class](#) [▶ 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.1.3 AdsConnection.Address Property

Gets the [AmsAddress](#) [▶ 752] of the ADS server.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsAddress Address { get; }
```

Property Value

Type: [AmsAddress](#) [▶ 752]

The server address.

Implements

[IAdsConnection.Address](#) [▶ 895]

Reference

[AdsConnection Class](#) [▶ 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.1.4 AdsConnection.ClientAddress Property

Get the [AmsAddress](#) [▶ 752] of the ADS client.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsAddress ClientAddress { get; }
```

Property Value

Type: [AmsAddress](#) [► 752]
The client address.

Implements

[IAdsConnection.ClientAddress](#) [► 895]

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.1.5 **AdsConnection.ConnectionEstablishedAt** Property

Gets the UTC time when the Connection was originally established.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTimeOffset? ConnectionEstablishedAt { get; }
```

Property Value

Type: [Nullable.DateTimeOffset](#).
The connection established at.

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.1.6 **AdsConnection.ConnectionLostCount** Property

Gets the connection lost count.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public int ConnectionLostCount { get; }
```

Property Value

Type: [Int32](#)
The connection lost count.

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.5.1.7 **AdsConnection.ConnectionLostTime** Property

Gets the connection lost time.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTimeOffset? ConnectionLostTime { get; }
```

Property Value

Type: [Nullable.DateTimeOffset](#).

The connection lost time.

Reference

[AdsConnection Class \[▸ 412\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.5.1.8 **AdsConnection.ConnectionState** Property

Gets the current Connection state of the [AdsConnection \[▸ 412\]](#)

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ConnectionState ConnectionState { get; }
```

Property Value

Type: [ConnectionState \[▸ 72\]](#)

The state of the connection.

Implements

[IConnectionStateProvider.ConnectionState \[▸ 90\]](#)

Remarks

The Connection state changes only if the [IConnection \[▸ 79\]](#) is established / shut down or active communication is triggered by the User of the [IConnection \[▸ 79\]](#) object.

Examples

The following sample shows how to keep the ConnectionState updated by triggering ADS Communication.

Trigger ConnectionState changes in WPF Applications

```
private DispatcherTimer _timer = null;  
private AdsSession _session = null;  
//private AdsConnection _connection = null;
```

```
private void Window_Loaded(object sender, RoutedEventArgs e)
{
    _session = new AdsSession(AmsNetId.Local, 10000);
    IConnection connection = _session.Connect();
    tbConnectionState.Text = connection.ConnectionState.ToString();
    _session.ConnectionStateChanged += _session_ConnectionStateChanged;

    _timer = new DispatcherTimer();
    _timer.Interval = TimeSpan.FromMilliseconds(200);
    _timer.Tick += TimerOnTick;

    _timer.Start();
}

private void Window_Unloaded(object sender, RoutedEventArgs e)
{
    _timer.Stop();
    _session.Dispose();
}

private void _session_ConnectionStateChanged(object sender, TwinCAT.ConnectionStateChangedEventArgs e)
{
    // ConnectionStateChanged will be triggered by communication Invokes
    tbConnectionState.Text = e.NewState.ToString();
}

private void TimerOnTick(object sender, EventArgs eventArgs)
{
    // The Timer Event will occur here in the UI thread because its an DispatcherTimer event!
    // An active ADS request will trigger Connection State periodically!
    StateInfo stateInfo;
    if (_session.Connection.TryReadState(out stateInfo) == AdsErrorCode.NoError)
    {
        tbAdsState.Text = stateInfo.AdsState.ToString();
    }
    else
    {
        tbAdsState.Text = "Invalid";
    }
}
```

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

[AdsConnection.ConnectionStateChanged](#) [► 658]

6.2.5.1.9 AdsConnection.DefaultValueEncoding Property

Gets the default value encoding.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Encoding DefaultValueEncoding { get; }
```

Property Value

Type: [Encoding](#)

The default value encoding.

Implements

[IConnection.DefaultValueEncoding](#) [► 81]

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.1.10 **AdsConnection.Disposed Property**

Gets a value indicating whether this [AdsConnection](#) [► 412] is disposed.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Disposed { get; }
```

Property Value

Type: [Boolean](#)

true if disposed; otherwise, false.

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.1.11 **AdsConnection.Id Property**

Gets the [AdsConnection](#) [► 412] identifier.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Id { get; }
```

Property Value

Type: [Int32](#)

The identifier.

Implements

[IConnection.Id](#) [► 81]

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.1.12 **AdsConnection.IsActive** Property

Gets a value indicating whether communication is in active state

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsActive { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is active; otherwise, false.

Reference

[AdsConnection Class](#) [▶ 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.1.13 **AdsConnection.IsConnected** Property

Gets a value indicating whether the local ADS port was opened successfully. It does not indicate if the target port is available. Use the method `ReadState` to determine if the target port is available.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsConnected { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is connected; otherwise, false.

Implements

[IConnection.IsConnected](#) [▶ 82]

Reference

[AdsConnection Class](#) [▶ 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.1.14 **AdsConnection.IsLocal** Property

Gets a value indicating whether the ADS client is connected to a ADS Server on the local computer.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsLocal { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is local; otherwise, false.

Implements

[IAdsConnection.IsLocal](#) [[▶ 896](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.1.15 **AdsConnection.IsLost Property**

Gets a value indicating whether the communication is in lost / open state

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsLost { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is lost; otherwise, false.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.1.16 **AdsConnection.IsReconnecting Property**

Gets a value indicating whether communication is ready for reconnecting

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsReconnecting { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is reconnecting; otherwise, false.

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.1.17 **AdsConnection.Name Property**

Gets the name of this [AdsConnection](#) [► 412].

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Name { get; }
```

Property Value

Type: [String](#)
The name.

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.1.18 **AdsConnection.ResurrectingTries Property**

Gets the number of tries to resurrect the [AdsConnection](#) [► 412].

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:
5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public int ResurrectingTries { get; }
```

Property Value

Type: [Int32](#)
The number of tried resurrections of the [IConnection](#) [► 79].

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.1.19 **AdsConnection.Resurrections Property**

Gets the number of succeeded connection resurrections.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:
5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public int Resurrections { get; }
```

Property Value

Type: [Int32](#)

The resurrection count.

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.1.20 **AdsConnection.Session Property**

Gets the Session object of the [AdsConnection](#) [► 412] object.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISession Session { get; }
```

Property Value

Type: [ISession](#) [► 92]

The client.

Implements

[IConnection.Session](#) [► 82]

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.1.21 **AdsConnection.State Property**

Gets the current [ConnectionState](#) [► 438]

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ConnectionState State { get; }
```

Property Value

Type: [ConnectionState](#) [► 72]
The state.

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.1.22 **AdsConnection.Timeout Property**

Gets the timeout (in milliseconds)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Timeout { get; set; }
```

Property Value

Type: [Int32](#)
The timeout.

Implements

[IConnection.Timeout](#) [► 83]

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.1.23 **AdsConnection.TotalConnectionLosses Property**

Gets the connection lost count.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int TotalConnectionLosses { get; }
```

Property Value

Type: [Int32](#)
The connection lost count.

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.1.24 **AdsConnection.TotalResurrectingTries Property**

Gets the number of tries to resurrect the [AdsConnection](#) [▶ 412].

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int TotalResurrectingTries { get; }
```

Property Value

Type: [Int32](#)

The number of tried resurrections of the [IConnection](#) [▶ 79].

Reference

[AdsConnection Class](#) [▶ 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.1.25 **AdsConnection.TotalResurrections Property**

Gets the number of succeeded connection resurrections.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int TotalResurrections { get; }
```

Property Value

Type: [Int32](#)

The resurrection count.

Reference

[AdsConnection Class](#) [▶ 412]















[TwinCAT.Ads Namespace](#) [▶ 179]
















6.2.5.2 **AdsConnection Methods**
















The [AdsConnection](#) [▶ 412] type exposes the following members.












Methods


















	Name	Description
	AddDeviceNotification(String, Int32, NotificationSettings, Object) [▶ 467]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 652] event.
	AddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object) [▶ 468]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event.
	AddDeviceNotificationAsync(String, Int32, NotificationSettings, Object, CancellationToken) [▶ 470]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 991] event.
	AddDeviceNotificationAsync(UInt32, UInt32, Int32, NotificationSettings, Object, CancellationToken) [▶ 471]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 652] event.
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type) [▶ 473]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 654] event.
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32.) [▶ 474]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 654] event.
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type) [▶ 475]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 654] event.
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32.) [▶ 476]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive ('AnyType') types are allowed for the parameter type.










	Name	Description
	AddDeviceNotificationExAsync(String, NotificationSettings, Object, Type, .Int32., CancellationToken) [▶ 478]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 654] event.
	AddDeviceNotificationExAsync(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., CancellationToken) [▶ 479]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 654] event.
	CleanupSymbolTable [▶ 480]	Clears the internal symbol cache.
	Close [▶ 481]	Closes the AdsConnection [▶ 412]
	Connect [▶ 481]	(Re)Connects the IConnection [▶ 79] when disconnected.
	CreateVariableHandle [▶ 493]	Generates a unique handle for an ADS variable.
	CreateVariableHandleAsync [▶ 494]	Determines the Symbol handle by its instance path asynchronously.
	DeleteDeviceNotification [▶ 495]	Deletes an existing notification.
	DeleteDeviceNotificationAsync [▶ 495]	Deletes a registered notification asynchronously.
	DeleteVariableHandle [▶ 496]	Releases the handle of a ADS variable again.
	DeleteVariableHandleAsync [▶ 497]	Releases the handle of a ADS variable again (asynchronously)
	Disconnect [▶ 498]	Disconnects this IConnection [▶ 79].
	Dispose [▶ 498]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)














	Name	Description
 	InvokeRpcMethod(String, .Object.) [▶ 499]	Invokes the specified RPC Method
 	InvokeRpcMethod(String, .Object., .Object.) [▶ 501]	Invokes the specified RPC Method
 	InvokeRpcMethod(String, .Object., AnyTypeSpecifier) [▶ 504]	Invokes the specified RPC Method
 	InvokeRpcMethod(String, .Object., .AnyTypeSpecifier, .AnyTypeSpecifier, .Object.) [▶ 502]	Invokes the specified RPC Method
 	InvokeRpcMethodAsync(String, .Object., CancellationToken) [▶ 506]	Invokes the specified RPC Method asynchronously
 	InvokeRpcMethodAsync(IRpcCallableInstance, IRpcMethod, .Object, .AnyTypeSpecifier, .AnyTypeSpecifier, CancellationToken) [▶ 510]	Invokes the specified RPC Method asynchronously
 	InvokeRpcMethodAsync(String, .Object., .AnyTypeSpecifier, AnyTypeSpecifier, CancellationToken) [▶ 508]	Invokes the specified RPC Method asynchronously
	Read(UInt32, Memory.Byte.) [▶ 514]	Reads the value from the symbol that is represented by the handle.















	Name	Description
	Read(UInt32, UInt32, Memory.Byte.) [▶ 515]	Reads data synchronously from an ADS device and writes it to the given readBuffer
	Read(UInt32, UInt32, Memory.Byte., Int32) [▶ 516]	Reads data synchronously from an ADS device and writes it to the given stream.
	ReadAny(UInt32, Type) [▶ 519]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, Type, .Int32.) [▶ 520]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type) [▶ 522]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type, .Int32.) [▶ 523]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type, .Int32., Int32) [▶ 524]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32) [▶ 517]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, .Int32.) [▶ 518]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, UInt32) [▶ 519]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, UInt32, .Int32.) [▶ 521]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, Type, CancellationToken) [▶ 528]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, Type, .Int32., CancellationToken) [▶ 529]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, UInt32, Type, CancellationToken) [▶ 532]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, UInt32,	Reads data asynchronously from an ADS device and writes it to an object.















	Name	Description
	Type, .Int32., CancellationToken [▶ 532]	
	ReadAnyAsync.T. (UInt32, CancellationToken) [▶ 526]	Reads data synchronously from an ADS device.
	ReadAnyAsync.T. (UInt32, .Int32., CancellationToken) [▶ 527]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T. (UInt32, UInt32, CancellationToken) [▶ 529]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T. (UInt32, UInt32, .Int32., CancellationToken) [▶ 531]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyString(UInt32, Int32, Encoding) [▶ 534]	Reads a string from the specified symbol/variable.
	ReadAnyString(UInt32, UInt32, Int32, Encoding) [▶ 535]	Reads as string from a specified address.
	ReadAnyStringAsync(UInt32, Int32, Encoding, CancellationToken) [▶ 536]	Reads a string asynchronously from the specified symbol/variable
	ReadAnyStringAsync(UInt32, UInt32, Int32, Encoding, CancellationToken) [▶ 537]	read any string as an asynchronous operation.
	ReadAsync(UInt32, Memory.Byte., CancellationToken) [▶ 540]	Reads the value data of the symbol asynchronously into the readBuffer.
	ReadAsync(UInt32, UInt32, Memory.Byte., CancellationToken) [▶ 541]	Reads the data asynchronously from specified IndexGroup/IndexOffset
	ReadDataType [▶ 541]	Call this method to obtain information about the specified data type.















	Name	Description
	ReadDataTypeAsync [▶ 542]	read data type as an asynchronous operation.
	ReadDeviceInfo [▶ 543]	Reads the identification and version number of an ADS server.
	ReadDeviceInfoAsync [▶ 544]	Reads the identification and version number of an ADS server.
	ReadState. [▶ 545]	Reads the ADS status and the device status from an ADS server.
	ReadState(Int32) [▶ 546]	Reads the ADS status and the device status from an ADS server.
	ReadStateAsync [▶ 546]	Read the ADS State asynchronously
	ReadSymbol [▶ 547]	Call this method to obtain information about the individual symbols (variables) in ADS devices.
	ReadSymbolAsync [▶ 548]	Call this method to obtain information about the individual symbols (variables) in ADS devices.
	ReadValue(ISymbol) [▶ 549]	Reads the value of a symbol and returns it as an object. Strings and all primitive datatypes (UInt32, Int32, Bool etc.) are supported. Arrays and structures cannot be read.
	ReadValue(String, Type) [▶ 551]	Reads the value of a symbol and returns the value as object. The parameter type must have the same layout as the ADS symbol.
	ReadValue.T. (ISymbol) [▶ 550]	Reads the value of a symbol and returns it as an object.
	ReadValue.T.(String) [▶ 549]	Reads the value of a symbol and returns the value. The parameter type must have the same layout as the ADS symbol.
	ReadValueAsync(ISymbol, Cancellation Token) [▶ 553]	Reads the value of a symbol and returns it as an object. Strings and all primitive datatypes (UInt32, Int32, Bool etc.) are supported. Arrays and structures cannot be read.
	ReadValueAsync(String, Type, Cancellation Token) [▶ 555]	Reads the value of a symbol asynchronously.
	ReadValueAsync.T. (ISymbol, Cancellation Token) [▶ 554]	Reads the value of a symbol asynchronously and returns it as an object.
	ReadValueAsync.T. (String, Cancellation Token) [▶ 552]	Reads the value of a symbol asynchronously.
	ReadWrite(UInt32, Memory.Byte, ReadOnlyMemory.Byte.) [▶ 558]	Writes data synchronously to an ADS device and then Reads data from that target.











	Name	Description
	ReadWrite(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte.) [▶ 559]	Writes data synchronously to an ADS device and then Reads data from this device into the readBuffer
	ReadWrite(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte, Int32) [▶ 560]	Writes data synchronously to an ADS device and then Reads data from this device.
	ReadWriteAsync(UInt32, Memory.Byte, ReadOnlyMemory.Byte, CancellationToken) [▶ 563]	ReadWrites value data asynchronously to/from the symbol represented by the variableHandle.
	ReadWriteAsync(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte, CancellationToken) [▶ 564]	Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer
	RegisterAdsStateChangedAsync [▶ 565]	Registers for AdsStateChanged [▶ 655] events as an asynchronous operation.
	RegisterSymbolVersionChanged [▶ 650]	Registers the symbol version changed.
	RegisterSymbolVersionChangedAsync [▶ 565]	Registers the symbol version changed asynchronously.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryAddDeviceNotification(String, Int32, NotificationSettings, Object, UInt32.) [▶ 567]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 652] event.
	TryAddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.) [▶ 568]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 652] event.










	Name	Description
	TryAddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32., UInt32.) [▶ 570]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event.
	TryAddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., UInt32.) [▶ 571]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 654] event.
	TryCreateVariableHandle [▶ 572]	Read (determine) the Symbol handle by its name/path
	TryDeleteDeviceNotification [▶ 572]	Deletes a registered notification.
	TryDeleteVariableHandle [▶ 573]	Releases the specified symbol/variable handle synchronously.
 	TryInvokeRpcMethod(String, String, .Object, Object.) [▶ 575]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(String, String, .Object., .Object., Object.) [▶ 577]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(IRpcCallableInstance, IRpcMethod, .Object., .AnyTypeSpecifier, .AnyTypeSpecifier, .Object..., Object.) [▶ 580]	Invokes the rpc method.
 	TryInvokeRpcMethod(String, String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object..., Object.) [▶ 578]	Invokes the rpc method.

	Name	Description
	TryRead(UInt32, Memory.Byte., Int32.) [▶ 585]	Reads the value synchronously data of the symbol, that is represented by the variable handle into the readBuffer.
	TryRead(UInt32, UInt32, Memory.Byte., Int32.) [▶ 586]	Reads data synchronously from an ADS device and writes it to the given stream.
	TryRead(UInt32, UInt32, Memory.Byte., Int32, Int32.) [▶ 586]	Reads data synchronously from an ADS device and writes it to the given stream.
	TryReadDataType [▶ 587]	Call this method to obtain information about the specified data type.
	TryReadState(StateInfo.) [▶ 589]	Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful.
	TryReadState(Int32, StateInfo.) [▶ 589]	Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful.
	TryReadSymbol [▶ 590]	Call this method to obtain information about the individual symbols (variables) in ADS devices.
	TryReadValue(ISymbol, Object.) [▶ 592]	Reads the value of a symbol and returns the value as object. The parameter type must have the same layout as the ADS symbol.
	TryReadValue(String, Type, Object.) [▶ 594]	Reads the value of a symbol and returns the value as object.
	TryReadValue.T.(ISymbol, T.) [▶ 593]	Reads the value of a symbol and returns it as an object.
	TryReadValue.T.(String, T.) [▶ 591]	Reads the value of a symbol and returns the value as object.
	TryReadWrite(UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.) [▶ 597]	ReadWrites value data synchronously to/from the symbol represented by the variableHandle.
	TryReadWrite(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.) [▶ 598]	Writes data synchronously to an ADS device and reads data from that device.
	TryReadWrite(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32, Int32.) [▶ 599]	Writes data synchronously to an ADS device and then Reads data from this device.





	Name	Description
	TryWrite(UInt32, ReadOnlyMemory.Byte) [▶ 603]	Writes the value data synchronously that is represented in the writeBuffer to the symbol with the specified variableHandle.
	TryWrite(UInt32, UInt32, ReadOnlyMemory.Byte) [▶ 604]	Writes data synchronously to an ADS device.
	TryWrite(UInt32, UInt32, ReadOnlyMemory.Byte, Int32) [▶ 604]	Writes data synchronously to an ADS device.
	TryWriteControl(StateInfo) [▶ 606]	Changes the ADS status and the device status of an ADS server.
	TryWriteControl(StateInfo, Int32) [▶ 606]	Changes the ADS status and the device status of an ADS server.
	TryWriteControl(StateInfo, ReadOnlyMemory.Byte) [▶ 608]	Changes the ADS status and the device status of an ADS server.
	TryWriteControl(StateInfo, ReadOnlyMemory.Byte, Int32) [▶ 609]	Changes the ADS status and the device status of an ADS server.
	TryWriteValue(ISymbol, Object) [▶ 612]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	TryWriteValue(String, Object) [▶ 611]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	TryWriteValue.T.(ISymbol, T) [▶ 613]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	TryWriteValue.T.(String, T) [▶ 611]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	UnregisterAdsStateChangedAsync [▶ 614]	Registers for AdsStateChanged [▶ 655] events as an asynchronous operation.
	UnregisterSymbolVersionChanged [▶ 651]	Unregisters the symbol version changed.
	UnregisterSymbolVersionChangedAsync [▶ 614]	Unregisters the symbol version changed asynchronous.

	Name	Description
	<u>Write(UInt32, ReadOnlyMemory.Byte.)</u> [▶ 619]	Writes data synchronously to an ADS device.
	<u>Write(UInt32, UInt32)</u> [▶ 616]	Trigger Client Method/Command.
	<u>Write(UInt32, UInt32, Int32)</u> [▶ 617]	Trigger Client Method/Command.
	<u>Write(UInt32, UInt32, ReadOnlyMemory.Byte.)</u> [▶ 619]	Writes data synchronously to an ADS device.
	<u>Write(UInt32, UInt32, ReadOnlyMemory.Byte., Int32)</u> [▶ 620]	Writes data synchronously to an ADS device.
	<u>WriteAny(UInt32, Object)</u> [▶ 621]	Writes an object synchronously to an ADS device.
	<u>WriteAny(UInt32, Object, .Int32.)</u> [▶ 622]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	<u>WriteAny(UInt32, UInt32, Object)</u> [▶ 623]	Writes an object synchronously to an ADS device.
	<u>WriteAny(UInt32, UInt32, Object, .Int32.)</u> [▶ 623]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	<u>WriteAny(UInt32, UInt32, Object, .Int32., Int32)</u> [▶ 624]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	<u>WriteAnyAsync(UInt32, Object, CancellationToken)</u> [▶ 626]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	<u>WriteAnyAsync(UInt32, Object, .Int32., CancellationToken)</u> [▶ 627]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	<u>WriteAnyAsync(UInt32, UInt32, Object, CancellationToken)</u> [▶ 628]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	<u>WriteAnyAsync(UInt32, UInt32,</u>	write any as an asynchronous operation.













	Name	Description
	Object, .Int32., CancellationToken [▶ 628]	
	WriteAnyStringAsync(String, String, Int32, Encoding, CancellationToken) [▶ 630]	write any string as an asynchronous operation.
	WriteAnyStringAsync(UInt32, String, Int32, Encoding, CancellationToken) [▶ 631]	write any string as an asynchronous operation.
	WriteAsync(UInt32, ReadOnlyMemory.Byte, CancellationToken) [▶ 634]	Writes the value data asynchronously that is represented by the writeBuffer to the symbol specified by the variableHandle.
	WriteAsync(UInt32, UInt32, CancellationToken) [▶ 633]	Triggers a write call at the specified IndexGroup/IndexOffset asynchronously.
	WriteAsync(UInt32, UInt32, ReadOnlyMemory.Byte, CancellationToken) [▶ 635]	Writes the data / Value asynchronously into the specified writeBuffer.
	WriteControl(StateInfo) [▶ 637]	Changes the ADS status and the device status of an ADS server.
	WriteControl(StateInfo, Int32) [▶ 637]	Changes the ADS status and the device status of an ADS server.
	WriteControl(StateInfo, ReadOnlyMemory.Byte) [▶ 639]	Changes the ADS status and the device status of an ADS server.
	WriteControl(StateInfo, ReadOnlyMemory.Byte, Int32) [▶ 640]	Changes the ADS status and the device status of an ADS server.
	WriteControlAsync(AdsState, UInt16, CancellationToken) [▶ 641]	Changes the ADS status and device status of the ADS server asynchronously.













	Name	Description
	WriteControlAsync(AdsState, UInt16, ReadOnlyMemory.Byte, CancellationToken) [▶ 642]	Writes the AdsState [▶ 729] and device state to the ADS device.
	WriteSymbolAsync [▶ 643]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	WriteValue(ISymbol, Object) [▶ 646]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValue(String, Object) [▶ 644]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	WriteValue.T.(ISymbol, T) [▶ 646]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValue.T.(String, T) [▶ 645]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	WriteValueAsync(ISymbol, Object, CancellationToken) [▶ 648]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValueAsync.T.(ISymbol, T, CancellationToken) [▶ 649]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValueAsync.T.(String, T, CancellationToken) [▶ 648]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
















Extension Methods




	Name	Description
	PollAdsState(IObservable.Unit.) [▶ 3294]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
	PollAdsState(IObservable.Unit.) [▶ 1256]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState(TimeSpan) [▶ 3295]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
	PollAdsState(TimeSpan) [▶ 1257]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)

	Name	Description
	PollAdsState2(IObservable.Unit.) [▶ 1271]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsState2(TimeSpan) [▶ 1272]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState2Async(IObservable.Unit., CancellationToken) [▶ 1274]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsState2Async(TimeSpan, CancellationToken) [▶ 1275]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsStateAsync(IObservable.Unit., CancellationToken) [▶ 3296]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
 	PollAdsStateAsync(IObservable.Unit., CancellationToken) [▶ 1259]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsStateAsync(TimeSpan, CancellationToken) [▶ 3298]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
 	PollAdsStateAsync(TimeSpan, CancellationToken) [▶ 1260]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollDeviceState(IObservable.Unit.) [▶ 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollDeviceState(TimeSpan) [▶ 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollDeviceStateAsync(IObservable.Unit., CancellationToken) [▶ 1278]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)








	Name	Description
	PollDeviceStateAsync(TimeSpan, CancellationToken) [▶ 1279]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollSystemServiceState [▶ 3282]	Polls the state of the system service. (Defined by SystemServiceExtension [▶ 3281].)
	PollSystemServiceStateAsync [▶ 3283]	Polls the system service state asynchronously (Defined by SystemServiceExtension [▶ 3281].)
	PollValues(ISymbol, Type, IObservable.Unit.) [▶ 1310]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, TimeSpan) [▶ 1311]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, IObservable.Unit.) [▶ 1300]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, TimeSpan) [▶ 1301]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32., TimeSpan) [▶ 1316]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, IObservable.Unit., Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, TimeSpan, Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32., TimeSpan) [▶ 1304]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, IObservable.Unit., Func.Exception, Object.) [▶ 1305]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, TimeSpan, Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	PollValues(ISymbol, Type, .Int32., IObservable.Unit., Func.Exception, Object.) [▶ 1321]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32., TimeSpan, Func.Exception, Object.) [▶ 1322]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32., IObservable.Unit., Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32., TimeSpan, Func.Exception, Object.) [▶ 1307]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, IObservable.Unit.) [▶ 1308]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, TimeSpan) [▶ 1309]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, IObservable.Unit.) [▶ 1294]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, TimeSpan) [▶ 1295]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, IObservable.Unit., Func.Exception, T.) [▶ 1314]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, TimeSpan, Func.Exception, T.) [▶ 1315]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, .Int32., IObservable.Unit.) [▶ 1312]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, .Int32., TimeSpan) [▶ 1313]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	PollValues.T.(String, IObservable.Unit, Func.Exception, T.) [▶ 1298]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, TimeSpan, Func.Exception, T.) [▶ 1299]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, .Int32, IObservable.Unit.) [▶ 1296]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	PollValues.T.(String, .Int32, TimeSpan) [▶ 1297]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, .Int32, IObservable.Unit, Func.Exception, T.) [▶ 1318]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	PollValues.T.(ISymbol, .Int32, TimeSpan, Func.Exception, T.) [▶ 1319]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, .Int32, IObservable.Unit, Func.Exception, T.) [▶ 1302]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	PollValues.T.(String, .Int32, TimeSpan, Func.Exception, T.) [▶ 1303]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues2.T.(ISymbol, .Int32, IObservable.Unit.) [▶ 1330]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues2.T.(String, .Int32, IObservable.Unit.) [▶ 1329]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	ReadSysServState [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (Defined by SystemServiceExtension [▶ 3281].)
	ReadSysServStateAsync [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (asynchronous) (Defined by SystemServiceExtension [▶ 3281].)

	Name	Description
	ReadWithFallback(UInt32, UInt32, Memory.Byte., UInt32, Boolean.) [▶ 1219]	Overloaded. Ads Read with fallback. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallback(UInt32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., Boolean.) [▶ 1220]	Overloaded. Ads Read with Fallback. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallbackAsync(UInt32, UInt32, UInt32, Memory.Byte., CancellationToken) [▶ 1221]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallbackAsync(UInt32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., CancellationToken) [▶ 1222]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedRead(UInt32, UInt32, Memory.Byte., Int32, TimeSpan) [▶ 1224]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedRead(UInt32, UInt32, Memory.Byte., Int32, TimeSpan, Func.ResultRead, Boolean.) [▶ 1225]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedReadAsync(UInt32, UInt32, Memory.Byte., Int32, TimeSpan, CancellationToken) [▶ 1226]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedReadAsync(UInt32, UInt32,	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)

	Name	Description
	Memory.Byte., Int32, TimeSpan, Func.ResultRead, Boolean., CancellationToken) [▶ 1227]	
	RestartTwinCATAsync [▶ 3285]	Sends a SystemService Restart to the target system and waits until the system has finished the state changes (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [▶ 3281].)
	SetAdsState [▶ 3299]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [▶ 3292].)
	SetAdsStateAsync [▶ 3300]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [▶ 3292].)
	WaitUntilRestarted [▶ 3286]	Waits until the Restart is detected on the client (SystemService, Port 10000) (Defined by SystemServiceExtension [▶ 3281].)
	WaitUntilRestartedAsync [▶ 3287]	Waits until the SystemService Restart is detected on the client (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [▶ 3281].)
 	WhenAdsStateChanges [▶ 1261]	Gets an observable sequence of AdsState [▶ 729]s. (Defined by AdsClientExtensions [▶ 1249].)
	WhenNotification(ISymbol) [▶ 1263]	Overloaded. Gets an observable sequence of Notification [▶ 1100]s. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenNotification(ISymbolCollection) [▶ 1264]	Overloaded. Gets an observable sequence of Notification [▶ 1100] objects. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenNotification(IList.ISymbol, NotificationSettings) [▶ 1268]	Overloaded. Gets an observable sequence of Notification [▶ 1100] objects. (Defined by AdsClientExtensions [▶ 1249].)
	WhenNotification(ISymbol, NotificationSettings) [▶ 1265]	Overloaded. Gets an observable sequence of SymbolValueNotification [▶ 1331]s. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenNotification(String, Type, NotificationSettings) [▶ 1324]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [▶ 1281].)
 	WhenNotification.T.(String, NotificationSettings) [▶ 1323]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [▶ 1281].)
	WhenSymbolVersionChanges [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)


	Name	Description
	WhenSymbolVersionChanges(ISchedule) [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenValueChanged [▶ 1345]	Observable sequence of Values driven by ADS Notifications on the specified symbol. (Defined by ValueSymbolExtensions [▶ 1333].)
 	WriteValues.T. (String, IObservable.T.) [▶ 1326]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)
 	WriteValues.T. (String, IObservable.T., Action.Exception.) [▶ 1327]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)

Reference

[AdsConnection Class](#) [[▶ 412](#)]



[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

-  [AdsClientExtensions.WhenNotification Method \(IAdsConnection, ISymbolCollection, NotificationSettings\)](#) [[▶ 1266](#)]

6.2.5.2.1 AdsConnection.AddDeviceNotification Method

Overload List

	Name	Description
	AddDeviceNotification(String, Int32, NotificationSettings, Object) [▶ 467]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 652] event.
	AddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object) [▶ 468]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.1.1 **AdsConnection.AddDeviceNotification Method (String, Int32, NotificationSettings, Object)**

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotification](#) [▶ 652] event.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint AddDeviceNotification(
    string variableName,
    int dataSize,
    NotificationSettings settings,
    Object? userData
)
```

Parameters

variableName	Type: System.String Name of the variable.
dataSize	Type: System.Int32 Maximum amount of data in bytes to receive with this ADS Notification.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)

Return Value

Type: [UInt32](#)
The notification handle.

Implements

[IAdsNotifications.AddDeviceNotification\(String, Int32, NotificationSettings, Object\)](#) [▶ 971]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Remarks

The
dataSize

Parameter defines the amount of bytes, that will be attached to the [AdsNotification](#) [▶ 652] as value. Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\)](#) [▶ 495] should always called when the notification is not used anymore.

Reference

[AdsConnection Class](#) [▶ 412]

[AddDeviceNotification Overload \[▶ 466\]](#)[TwinCAT.Ads Namespace \[▶ 179\]](#)[AdsConnection.AdsNotification \[▶ 652\]](#)[AdsConnection.DeleteDeviceNotification\(UInt32\) \[▶ 495\]](#)[AddDeviceNotification Overload \[▶ 971\]](#)[AddDeviceNotificationAsync Overload \[▶ 973\]](#)[TryAddDeviceNotification Overload \[▶ 985\]](#)

6.2.5.2.1.2 **AdsConnection.AddDeviceNotification Method (UInt32, UInt32, Int32, NotificationSettings, Object)**

Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint AddDeviceNotification(
    uint indexGroup,
    uint indexOffset,
    int dataSize,
    NotificationSettings settings,
    Object? userData
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
dataSize	Type: System.Int32 Maximum amount of data in bytes to receive with this ADS Notification.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)

Return Value

Type: [UInt32](#)
The notification handle.

Implements

[IAdsNotifications.AddDeviceNotification\(UInt32, UInt32, Int32, NotificationSettings, Object\) \[▶ 972\]](#)

Exceptions

Exception	Condition
ObjectDisposedException	

Exception	Condition
ClientNotConnectedException [▶ 67]	

Remarks

The
dataSize



Parameter defines the amount of bytes, that will be attached to the [AdsNotification](#) [[▶ 652](#)] as value. Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\)](#) [[▶ 495](#)] should always called when the notification is not used anymore.

Reference

- [AdsConnection Class](#) [[▶ 412](#)]
- [AddDeviceNotification Overload](#) [[▶ 466](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]
- [AdsConnection.DeleteDeviceNotification\(UInt32\)](#) [[▶ 495](#)]
- [AdsConnection.AdsNotification](#) [[▶ 652](#)]
- [AdsConnection.AdsNotificationError](#) [[▶ 654](#)]
- [AddDeviceNotification Overload](#) [[▶ 971](#)]
- [TryAddDeviceNotification Overload](#) [[▶ 985](#)]
- [AddDeviceNotificationAsync Overload](#) [[▶ 973](#)]

6.2.5.2.2 AdsConnection.AddDeviceNotificationAsync Method

Overload List

	Name	Description
	AddDeviceNotificationAsync(String, Int32, NotificationSettings, Object, CancellationToken) [▶ 470]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 991] event.
	AddDeviceNotificationAsync(UInt32, UInt32, Int32, NotificationSettings, Object, CancellationToken) [▶ 471]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 652] event.

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.2.1 **AdsConnection.AddDeviceNotificationAsync Method (String, Int32, NotificationSettings, Object, CancellationToken)**

Connects a variable to the ADS client asynchronously. The ADS client will be notified by the [AdsNotification](#) [► 991] event.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultHandle> AddDeviceNotificationAsync (
    string symbolPath,
    int dataSize,
    NotificationSettings settings,
    Object? userData,
    CancellationToken cancel
)
```

Parameters

symbolPath	Type: System.String The symbol/instance path of the ADS variable.
dataSize	Type: System.Int32 Maximum amount of data in bytes to receive with this ADS Notification.
settings	Type: TwinCAT.Ads.NotificationSettings [► 1106] The notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
cancel	Type: System.Threading.CancellationToken The Cancellation token.

Return Value

Type: [Task.ResultHandle](#) [► 1136].

A task that represents the asynchronous 'AddDeviceNotification' operation. The [ResultHandle](#) [► 1136] type parameter contains the created handle (Handle) and the [ErrorCode](#) [► 1120] after execution.

Implements

[IAdsNotifications.AddDeviceNotificationAsync\(String, Int32, NotificationSettings, Object, CancellationToken\)](#) [► 974]

Remarks

The
dataSize

Parameter defines the amount of bytes, that will be attached to the [AdsNotification](#) [▶ 652] as value. Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [▶ 984] should always be called when the notification is not used anymore.

Reference

[AdsConnection Class](#) [▶ 412]

[AddDeviceNotificationAsync Overload](#) [▶ 469]

[TwinCAT.Ads Namespace](#) [▶ 179]

[IAdsNotifications.AdsNotification](#) [▶ 991]

[IAdsNotifications.DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [▶ 984]

[AddDeviceNotification Overload](#) [▶ 971]

[AddDeviceNotificationAsync Overload](#) [▶ 973]

[TryAddDeviceNotification Overload](#) [▶ 985]

6.2.5.2.2 AdsConnection.AddDeviceNotificationAsync Method (UInt32, UInt32, Int32, NotificationSettings, Object, CancellationToken)

Connects a variable to the ADS client asynchronously. The ADS client will be notified by the [AdsNotification](#) [▶ 652] event.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultHandle> AddDeviceNotificationAsync (
    uint indexGroup,
    uint indexOffset,
    int dataSize,
    NotificationSettings settings,
    Object? userData,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group number of the requested ADS service.
indexOffset	Type: System.UInt32 The index offset number of the requested ADS service.
dataSize	Type: System.Int32 Maximum amount of data in bytes to receive with this ADS Notification.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
cancel	Type: System.Threading.CancellationTok The Cancellation token.

Return Value

Type: [Task.ResultHandle](#) [[▶ 1136](#)].

A task that represents the asynchronous 'AddDeviceNotification' operation. The [ResultHandle](#) [[▶ 1136](#)] type parameter contains the created handle (Handle) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsNotifications.AddDeviceNotificationAsync\(UInt32, UInt32, Int32, NotificationSettings, Object, CancellationToken\)](#) [[▶ 975](#)]

Remarks

The

dataSize

Parameter defines the amount of bytes, that will be attached to the [AdsNotification](#) [[▶ 652](#)] as value. Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [[▶ 495](#)] should always be called when the notification is not used anymore.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[AddDeviceNotificationAsync Overload](#) [[▶ 469](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[AdsConnection.AdsNotification](#) [[▶ 652](#)]



[AdsConnection.DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [[▶ 495](#)]



[AddDeviceNotification Overload](#) [[▶ 971](#)]

[AddDeviceNotificationAsync Overload](#) [[▶ 973](#)]

[TryAddDeviceNotification Overload](#) [[▶ 985](#)]

6.2.5.2.3 AdsConnection.AddDeviceNotificationEx Method**Overload List**

	Name	Description
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type) [▶ 473]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 654] event.
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32.) [▶ 474]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 654] event.

	Name	Description
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type) [▶ 475]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 654] event.
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32.) [▶ 476]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive ('AnyType') types are allowed for the parameter type.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.3.1 AdsConnection.AddDeviceNotificationEx Method (String, NotificationSettings, Object, Type)

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotificationEx](#) [[▶ 654](#)] event.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint AddDeviceNotificationEx(
    string symbolPath,
    NotificationSettings settings,
    Object? userData,
    Type type
)
```

Parameters

symbolPath	Type: System.String Symbol/Instance path of the ADS variable.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
type	Type: System.Type Type of the object stored in the event argument ('AnyType')

Return Value

Type: [UInt32](#)
The notification handle.

Implements

[IAdsNotifications.AddDeviceNotificationEx\(String, NotificationSettings, Object, Type\)](#) [[▶ 977](#)]

Remarks

Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\) \[▶ 495\]](#) should always be called when the notification is not used anymore.

Reference

[AdsConnection Class \[▶ 412\]](#)

[AddDeviceNotificationEx Overload \[▶ 472\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

[AdsConnection.AdsNotificationEx \[▶ 654\]](#)

[AdsConnection.DeleteDeviceNotification\(UInt32\) \[▶ 495\]](#)

[AddDeviceNotificationEx Overload \[▶ 976\]](#)

[AddDeviceNotificationExAsync Overload \[▶ 981\]](#)

[TryAddDeviceNotificationEx Overload \[▶ 988\]](#)

6.2.5.2.3.2 **AdsConnection.AddDeviceNotificationEx Method (String, NotificationSettings, Object, Type, .Int32.)**

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotificationEx \[▶ 654\]](#) event.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint AddDeviceNotificationEx(
    string symbolPath,
    NotificationSettings settings,
    Object? userData,
    Type type,
    int[]? args
)
```

Parameters

symbolPath	Type: System.String Symbol/Instance path of the ADS variable.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
type	Type: System.Type Type of the object stored in the event argument ('AnyType')
args	Type: System.Int32 . Additional arguments (for 'AnyType')

Return Value

Type: [UInt32](#)

The notification handle.

Implements

[IAdsNotifications.AddDeviceNotificationEx\(String, NotificationSettings, Object, Type, .Int32.\)](#) [▶ 977]

Remarks

Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\)](#) [▶ 495] should always called when the notification is not used anymore.

Reference

[AdsConnection Class](#) [▶ 412]

[AddDeviceNotificationEx Overload](#) [▶ 472]

[TwinCAT.Ads Namespace](#) [▶ 179]

[AdsConnection.AdsNotificationEx](#) [▶ 654]

[AdsConnection.DeleteDeviceNotification\(UInt32\)](#) [▶ 495]

[AddDeviceNotificationEx Overload](#) [▶ 976]

[AddDeviceNotificationExAsync Overload](#) [▶ 981]

[TryAddDeviceNotificationEx Overload](#) [▶ 988]

6.2.5.2.3.3 AdsConnection.AddDeviceNotificationEx Method (UInt32, UInt32, NotificationSettings, Object, Type)

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotificationEx](#) [▶ 654] event.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint AddDeviceNotificationEx(
    uint indexGroup,
    uint indexOffset,
    NotificationSettings settings,
    Object? userData,
    Type type
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
type	Type: System.Type Type of the object stored in the event argument ('AnyType')

Return ValueType: [UInt32](#)

The notification handle.

Implements[IAdsNotifications.AddDeviceNotificationEx\(UInt32, UInt32, NotificationSettings, Object, Type\)](#) [► 978]**Remarks**

Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\)](#) [► 495] should always be called when the notification is not used anymore.

Reference[AdsConnection Class](#) [► 412][AddDeviceNotificationEx Overload](#) [► 472][TwinCAT.Ads Namespace](#) [► 179][AdsConnection.DeleteDeviceNotification\(UInt32\)](#) [► 495][AdsConnection.AdsNotificationEx](#) [► 654][AdsConnection.AdsNotificationError](#) [► 654][AddDeviceNotificationEx Overload](#) [► 976][TryAddDeviceNotificationEx Overload](#) [► 988][AddDeviceNotificationExAsync Overload](#) [► 981]**6.2.5.2.3.4 AdsConnection.AddDeviceNotificationEx Method (UInt32, UInt32, NotificationSettings, Object, Type, .Int32.)**

Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive ('AnyType') types are allowed for the parameter type.

Namespace: [TwinCAT.Ads](#) [► 179]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public uint AddDeviceNotificationEx(
    uint indexGroup,
    uint indexOffset,
    NotificationSettings settings,
    Object? userData,
    Type type,
    int[] args
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
------------	--

indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data.
type	Type: System.Type Type of the object stored in the event argument.
args	Type: System.Int32 . Additional arguments for 'AnyType' types.

Return Value

Type: [UInt32](#)
The notification handle.

Implements

[IAdsNotifications.AddDeviceNotificationEx\(UInt32, UInt32, NotificationSettings, Object, Type, .Int32.\)](#) [▶ [980](#)]

Remarks



Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\)](#) [▶ [495](#)] should always called when the notification is not used anymore.

Reference

- [AdsConnection Class](#) [▶ [412](#)]
- [AddDeviceNotificationEx Overload](#) [▶ [472](#)]
- [TwinCAT.Ads Namespace](#) [▶ [179](#)]
- [AdsConnection.AdsNotificationEx](#) [▶ [654](#)]
- [AddDeviceNotificationEx Overload](#) [▶ [976](#)]
- [AddDeviceNotificationExAsync Overload](#) [▶ [981](#)]
- [TryAddDeviceNotificationEx Overload](#) [▶ [988](#)]

6.2.5.2.4 AdsConnection.AddDeviceNotificationExAsync Method

Overload List

	Name	Description
	AddDeviceNotificationExAsync(String, NotificationSettings, Object, Type, .Int32., CancellationTok en) [▶ 478]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 654] event.
	AddDeviceNotificationExAsync(UInt32, UInt32,	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 654] event.

	Name	Description
	NotificationSettings , Object , Type , .Int32 , CancellationToken) [▶ 479]	

Reference

[AdsConnection Class](#) [\[▶ 412\]](#)

[TwinCAT.Ads Namespace](#) [\[▶ 179\]](#)

6.2.5.2.4.1 **AdsConnection.AddDeviceNotificationExAsync Method (String, NotificationSettings, Object, Type, .Int32., CancellationTokent)**

Connects a variable to the ADS client asynchronously. The ADS client will be notified by the [AdsNotificationEx](#) [\[▶ 654\]](#) event.

Namespace: [TwinCAT.Ads](#) [\[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultHandle> AddDeviceNotificationExAsync(
    string symbolPath,
    NotificationSettings settings,
    Object? userData,
    Type type,
    int[]? args,
    CancellationTokent cancel
)
```

Parameters

symbolPath	Type: System.String The symbol/instance path of the ADS variable.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
type	Type: System.Type Type of the object stored in the event argument ('AnyType')
args	Type: .System.Int32 . Additional arguments (for 'AnyType')
cancel	Type: System.Threading.CancellationTokent The Cancellation token.

Return Value

Type: [Task.ResultHandle](#) [\[▶ 1136\]](#).

A task that represents the asynchronous 'AddDeviceNotification' operation. The [ResultHandle](#) [\[▶ 1136\]](#) type parameter contains the created handle ([Handle](#) [\[▶ 1138\]](#)) and the [ErrorCode](#) [\[▶ 1120\]](#) after execution.

Implements

[IAdsNotifications.AddDeviceNotificationExAsync\(String, NotificationSettings, Object, Type, .Int32., CancellationToken\)](#) [[▶ 981](#)]

Remarks

Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [[▶ 495](#)] should always be called when the notification is not used anymore.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[AddDeviceNotificationExAsync Overload](#) [[▶ 477](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[AdsConnection.AdsNotificationEx](#) [[▶ 654](#)]

[AdsConnection.DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [[▶ 495](#)]

[AddDeviceNotificationEx Overload](#) [[▶ 976](#)]

[AddDeviceNotificationExAsync Overload](#) [[▶ 981](#)]

[TryAddDeviceNotificationEx Overload](#) [[▶ 988](#)]

6.2.5.2.4.2 AdsConnection.AddDeviceNotificationExAsync Method (UInt32, UInt32, NotificationSettings, Object, Type, .Int32., CancellationToken)

Connects a variable to the ADS client asynchronously. The ADS client will be notified by the [AdsNotificationEx](#) [[▶ 654](#)] event.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultHandle> AddDeviceNotificationExAsync (
    uint indexGroup,
    uint indexOffset,
    NotificationSettings settings,
    Object? userData,
    Type type,
    int[]? args,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The settings.

userData	Type: System.Object This object can be used to store user specific data.
type	Type: System.Type Type of the object stored in the event argument, only Primitive 'AnyTypes' allowed.
args	Type: .System.Int32 . Additional arguments (for 'AnyType')
cancel	Type: System.Threading.CancellationToken The Cancellation token.

Return Value

Type: [Task.ResultHandle](#) [[▶ 1136](#)].

A task that represents the asynchronous 'AddDeviceNotification' operation. The [ResultHandle](#) [[▶ 1136](#)] type parameter contains the created handle (Handle) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsNotifications.AddDeviceNotificationExAsync\(UInt32, UInt32, NotificationSettings, Object, Type, .Int32, CancellationToken\)](#) [[▶ 982](#)]

Remarks

If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive types (AnyType) are supported by this method. Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [[▶ 495](#)] should always called when the notification is not used anymore.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[AddDeviceNotificationExAsync Overload](#) [[▶ 477](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[AdsConnection.DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [[▶ 495](#)]

[AdsConnection.AdsNotificationEx](#) [[▶ 654](#)]

[AdsConnection.AdsNotificationError](#) [[▶ 654](#)]

[AddDeviceNotificationEx Overload](#) [[▶ 976](#)]

[TryAddDeviceNotificationEx Overload](#) [[▶ 988](#)]

[AddDeviceNotificationExAsync Overload](#) [[▶ 981](#)]

6.2.5.2.5 **AdsConnection.CleanupSymbolTable Method**

Clears the internal symbol cache.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void CleanupSymbolTable()
```

Implements

[IAdsSymbolicAccess.CleanupSymbolTable.](#) [[▶ 10701](#)]

Remarks

Previously stored symbol information is cleared. As a consequence the symbol information must be obtained from the ADS server again if accessed, which which needs an extra ADS round trip.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.6 **AdsConnection.Close Method**

Closes the [AdsConnection](#) [[▶ 412](#)]

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Close()
```

Implements

[IConnection.Close.](#) [[▶ 83](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.7 **AdsConnection.Connect Method**

(Re)Connects the [IConnection](#) [[▶ 79](#)] when disconnected.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Connect()
```

Return Value

Type: [Boolean](#)

true if the [AdsConnection](#) [▶ 412] is reconnected, false otherwise.

Implements

[IConnection.Connect.](#) [▶ 84]

Reference

[AdsConnection Class](#) [▶ 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.8 AdsConnection.CreateSymbolLoader Method

Creates a new instance of the [Symbol loader](#) [▶ 1760] with the specified mode.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public IAdsSymbolLoader CreateSymbolLoader(
    ISession session,
    ISymbolLoaderSettings settings
)
```

Parameters

session	Type: TwinCAT.ISession [▶ 92] The session (for session orientated loads/symbols). Can be NULL if not present.
settings	Type: TwinCAT.ISymbolLoaderSettings [▶ 104] The settings.

Return Value

Type: [IAdsSymbolLoader](#) [▶ 1760]

The [IAdsSymbolLoader](#) [▶ 1760] interface of the Symbol loader.

Exceptions

Exception	Condition
ObjectDisposedException	
ObjectDisposedException	

Remarks

The Symbol Loader (V2) supports the following [modes](#) [▶ 159]. [Flat](#) [▶ 159]The flat mode organizes the Symbols in a flat list. At the beginning this List caches only the root symbol objects, which can be enumerated. To access the sub elements like structure fields or array elements use the [SubSymbols](#) [▶ 2696] collection. The property get accessor generates the subsymbols lazy on the fly (performance optimized) and stores them internally as weak reference (memory optimized). This mode is available in all .NET versions.[VirtualTree](#) [▶ 159]On top of the behaviour of the [Flat](#) [▶ 159], the virtual tree mode organizes the Symbols hierarchically with parent-child relationships. That eases the access to the hierarchical structure but

needs slightly more preprocessing of the data. This mode is available in all .NET Versions. [DynamicTree \[► 159\]](#) The Dynamic tree mode organizes the Symbols hierarchically and (dynamically) creates struct members, array elements and enum fields on the fly. 'Dynamically' means here not only lazy creation like in [Flat \[► 159\]](#), but furthermore real creation of type safe .NET complex types/instances as representatives of the TwinCAT Symbol objects/types. This feature is only available on platforms that support the Dynamic Language Runtime (DLR); actually all .NET Framework Version larger than 4.0. Virtual instances means, that all Symbols are ordered within a tree structure. For that symbol nodes that are not located on a fixed address, a Virtual Symbol will be created. Setting the virtualInstance parameter to 'false' means, that the located symbols will be returned in a flattened list.

Examples

The following sample shows how to create a dynamic version of the SymbolLoader V2. The dynamic symbol loader makes use of the Dynamic Language Runtime (DLR) of the .NET Framework. That means Structures, Arrays and Enumeration types and instances are generated 'on-the-fly' during symbol Browsing. These created dynamic objects are a one to one representation of the Symbol Server target objects (e.g the IEC61131 types on the PLC). Dynamic language features are only available from .NET4 upwards.

Dynamic Tree Mode

```
namespace Sample
{
    using System;
    using System.Diagnostics;
    using System.Threading;
    using TwinCAT;
    using TwinCAT.Ads;
    using TwinCAT.Ads.TypeSystem;
    using TwinCAT.Ads.ValueAccess;
    using TwinCAT.TypeSystem;
    using TwinCAT.TypeSystem.Generic;
    using TwinCAT.ValueAccess;

    class SymbolBrowserProgramV2DynamicTree
    {
        #region CODE_SAMPLE_SIMPLEDYNAMIC
        /// <summary>
        /// Defines the entry point of the application.
        /// </summary>
        /// <param name="args">The arguments.</param>
        static async void Main(string[] args)
        {
            // Get the AdsAddress from command-line arguments
            AmsAddress address = ArgParser.Parse(args);

            CancellationTokenSource cancelSource = new CancellationTokenSource();
            CancellationToken cancel = cancelSource.Token;

            using (AdsClient client = new AdsClient())
            {
                // Connect to the target device
                client.Connect(address);

                // Usage of "dynamic" Type and Symbols (>= .NET4 only)
                SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.DynamicTree);
                IAdsSymbolLoader dynLoader = (IAdsSymbolLoader)SymbolLoaderFactory.Create(client, settings);

            #endregion

            // Set the Default setting for Notifications
            dynLoader.DefaultNotificationSettings = new NotificationSettings(AdsTransMode.OnChange, 200,
            2000);

            // Get the Symbols (Dynamic Symbols)
            var resultSymbols = await ((IDynamicSymbolLoader)dynLoader).GetDynamicSymbolsAsync(cancel);

            dynamic dynamicSymbols = resultSymbols.Symbols;
            dynamic adsPort = dynamicSymbols.TwinCAT_SystemInfoVarList._AppInfo.AdsPort;

            #region CODE_SAMPLE_SIMPLEDYNAMIC

            // Access Main Symbol with Dynamic Language Runtime support (DLR)
            // Dynamically created property "Main"
```

```

//dynamic symMain = dynamicSymbols.Main;

// Main is an 'VirtualSymbol' / Organizational unit that doesn't have a value
// Calling ReadValue is not allowed
//bool test = symMain.HasValue;
//dynamic invalid = symMain.ReadValue();

//Reading TaskInfo Value
//
With calling ReadValueAsync() a 'snapshot' of the Symbols Instance is taken (reading async)
ResultReadValueAccess resultRead = await dynamicSymbols.TwinCAT_SystemInfoVarList._TaskInfo.
ReadValueAsync(cancel);
dynamic vTaskInfoArray = resultRead.Value;

// Getting the Snapshot time in UTC format
DateTimeOffset timeStamp1 = vTaskInfoArray.TimeStamp;

// Getting TaskInfo Symbol for Task 1
dynamic symTaskInfo1 = dynamicSymbols.TwinCAT_SystemInfoVarList._TaskInfo[1];

// Getting CycleCount Symbol
dynamic symCycleCount = symTaskInfo1.CycleCount;

// Take Snapshot value of the ApplicationInfo struct
resultRead = await dynamicSymbols.TwinCAT_SystemInfoVarList._AppInfo.ReadValueAsync(cancel);
dynamic vAppInfo = resultRead.Value;

// Get the UTC Timestamp of the snapshot
DateTimeOffset timeStamp2 = vAppInfo.TimeStamp;

// Access the ProjectName of the ApplicationInfo Snapshot (type-safe!)
string projectNameValue = vAppInfo.ProjectName;

// Reading the CycleCount Value
resultRead = await symTaskInfo1.CycleCount.ReadValueAsync(cancel); // Taking a Value Sna
psshot
int cycleCountValue = (int)resultRead.Value;
#endregion

// Registering for dynamic "ValueChanged" events for the Values
// Using Default Notification settings
symCycleCount.ValueChanged += new EventHandler<ValueChangedEventArgs>(cycleCount_ValueChange
d);

// Override default notification settings
symTaskInfo1.NotificationSettings = new NotificationSettings(AdsTransMode.Cyclic, 500, 0);

// Register for ValueChanged event.
symTaskInfo1.ValueChanged += new EventHandler<ValueChangedEventArgs>(taskInfo1Value_ValueCha
nged); // Struct Type

Thread.Sleep(10000); // Sleep main thread for 10 Seconds
}
Console.WriteLine("CycleCount Changed events received: {0}", _cycleCountEvents);
Console.WriteLine("taskInfo1 Changed events received: {0}", _taskInfo1Events);

Console.WriteLine("");
Console.WriteLine("Press [Enter] for leave:");
Console.ReadLine();
}

static object _notificationSynchronizer = new object();
static int _cycleCountEvents = 0;

/// <summary>
/// Handler function for the CycleCount ValueChanged event.
/// </summary>
/// <param name="sender">The sender.</param>
/// <param name="e">The e.</param>
static void cycleCount_ValueChanged(object sender, ValueChangedEventArgs e)
{
    lock(_notificationSynchronizer)
    {
        Interlocked.Increment(ref _cycleCountEvents);
        // val is a type safe value of int!
        dynamic val = e.Value;
        uint intVal = val;

        DateTimeOffset changedTime = e.DateTime.ToLocalTime(); // Convert UTC to local time
    }
}

```

```

        Console.WriteLine("CycleCount changed to: {0}, TimeStamp: {1}", intVal, changedTime.ToString(
("HH:mm:ss:fff"));
    }
}

static int _taskInfoEvents = 0;

/// <summary>
/// Handler function for the TaskInfo ValueChanged event.
/// </summary>
/// <param name="sender">The sender.</param>
/// <param name="e">The e.</param>
static void taskInfoValue_ValueChanged(object sender, ValueChangedEventArgs e)
{
    lock (_notificationSynchronizer)
    {
        Interlocked.Increment(ref _taskInfoEvents);
        dynamic val = e.Value;
        DateTimeOffset changedTime = e.DateTime.ToLocalTime(); // Convert to local time

        // Val is a during Runtime created struct type and contains
        // the same Properties as related PLC object.
        int cycleTime = val.CycleTime;
        Console.WriteLine("TaskInfoValue changed TimeStamp: {0}", changedTime.ToString("HH:mm:ss:fff"));
    }
}
}
}
}
}
}

```

The following sample shows how to create a static (non dynamic) version of the SymbolLoader V2. The static symbol loader in version 2 is a nearly code compatible version of the Dynamic Loader, only the dynamic creation of objects is not available. The reason for supporting this mode is that .NET Framework Versions lower than Version 4.0 (CLR2) doesn't support the Dynamic Language Runtime (DLR). The SymbolLoader V2 static object is supported from .NET 2.0 on.

Virtual Tree Mode

```

using System;
using System.Threading;
using System.Diagnostics;
using TwinCAT;
using TwinCAT.Ads;
using TwinCAT.TypeSystem;
using TwinCAT.TypeSystem.Generic;
using TwinCAT.Ads.ValueAccess;
using TwinCAT.Ads.TypeSystem;

namespace Sample
{
    class SymbolBrowserProgramV2VirtualTree
    {
        /// <summary>
        /// Defines the entry point of the application.
        /// </summary>
        /// <param name="args">The arguments.</param>
        static void Main(string[] args)
        {
            ConsoleLogger logger = new ConsoleLogger();

            Console.WriteLine("");
            Console.WriteLine("Press [Enter] for start:");
            Console.ReadLine();

            //logger.Active = false;

            Stopwatch stopper = new Stopwatch();

            // Parse the command-line arguments
            AmsAddress address = ArgParser.Parse(args);

            stopper.Start();

            using (AdsClient client = new AdsClient())
            {
                //client.Synchronize = false;

                // Connect the AdsClient to the device target.
            }
        }
    }
}

```

```

        client.Connect(address);

        // Creates the Symbol Objects as hierarchical tree
        SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoaderSettings.VirtualTree, ValueAccessMode.IndexGroupOffsetPreferred);
        ISymbolLoader symbolLoader = SymbolLoaderFactory.Create(client, settings);

        // Dump Datatypes from Target Device
        Console.WriteLine(string.Format("Dumping '{0}' DataTypes:", symbolLoader.DataTypes.Count));
        foreach (IDataType type in symbolLoader.DataTypes)
        {
            logger.DumpType(type);
        }
        Console.WriteLine("");

        // Dump Symbols from target device
        Console.WriteLine("Dumping '{0}' Symbols:", symbolLoader.Symbols.Count);
        foreach (ISymbol symbol in symbolLoader.Symbols)
        {
            logger.DumpSymbol(symbol, 0);
        }
        stopper.Stop();
        TimeSpan elapsed = stopper.Elapsed;

        Console.WriteLine("");
        Console.WriteLine("Browsing complete tree: {0},
({1} DataTypes, {2} Symbols)", elapsed, logger.DataTypesCount, logger.SymbolsCount);
        Console.WriteLine("Press [Enter] for leave:");
        Console.ReadLine();
    }
}

```

Examples

The SymbolLoader V2 static object is supported from .NET 2.0 on.

Flat Mode

```

using System;
using System.Diagnostics;
using System.Threading;
using TwinCAT;
using TwinCAT.Ads;
using TwinCAT.Ads.TypeSystem;
using TwinCAT.Ads.ValueAccess;
using TwinCAT.TypeSystem;
using TwinCAT.TypeSystem.Generic;

namespace Sample
{
    class SymbolBrowserProgramV2Flat
    {
        /// <summary>
        /// Defines the entry point of the application.
        /// </summary>
        /// <param name="args">The arguments.</param>
        static void Main(string[] args)
        {
            ConsoleLogger logger = new ConsoleLogger();

            Console.WriteLine("");
            Console.WriteLine("Press [Enter] for start:");
            Console.ReadLine();

            //logger.Active = false;

            Stopwatch stopper = new Stopwatch();

            // Parse the command line arguments
            AmsAddress address = ArgParser.Parse(args);

            stopper.Start();

            // Create the ADS Client
            using (AdsClient client = new AdsClient())
            {
                //client.Synchronize = false;

                // Connect to Address

```

```

        client.Timeout = 30000;
        client.Connect(address);

        // Creates the Symbol Objects in Flat Mode (Flat list)
        SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.Flat, ValueAccessMo
de.IndexGroupOffsetPreferred);
        ISymbolLoader symbolLoader = SymbolLoaderFactory.Create(client, settings);

        // Dump Datatypes from Target Device
        Console.WriteLine(string.Format("Dumping '{0}' DataTypes:", symbolLoader.DataTypes.Count));
        foreach (IDataType type in symbolLoader.DataTypes)
        {
            logger.DumpType(type);
        }

        Console.WriteLine("");

        // Dump Symbols from target device
        Console.WriteLine("Dumping '{0}' Symbols:", symbolLoader.Symbols.Count);
        foreach (ISymbol symbol in symbolLoader.Symbols)
        {
            logger.DumpSymbol(symbol, 0);
        }
        stopper.Stop();
        TimeSpan elapsed = stopper.Elapsed;

        Console.WriteLine("");
        Console.WriteLine("Browsing complete tree: {0},
({1} DataTypes, {2} Symbols)", elapsed, logger.DataTypesCount, logger.DataTypesCount);
        Console.WriteLine("Press [Enter] for leave:");
        Console.ReadLine();
    }

```

Examples

Argument Parser

```

public static class ArgParser
{
    /// <summary>
    /// Parses the arguments.
    /// </summary>
    /// <param name="args">The arguments.</param>
    /// <returns>AmsAddress.</returns>
    public static AmsAddress Parse(string[] args)
    {
        AmsNetId netId = AmsNetId.Local;
        int port = 851;

        if (args != null)
        {
            if (args.Length > 0 && args[0] != null)
                netId = AmsNetId.Parse(args[0]);

            if (args.Length > 1 && args[1] != null)
                port = int.Parse(args[1]);
        }
        return new AmsAddress(netId, port);
    }
}

```

Dumping Symbols

```

/// <summary>
/// Console logger
/// </summary>
public class ConsoleLogger
{
    public ConsoleLogger()
    {
    }
    bool _active = true;

    /// <summary>
    /// Gets or sets a value indicating whether this ConsoleLogger is active.
    /// </summary>
    /// <value><c>true</c> if active; otherwise, <c>false</c>.</value>

```

```

public bool Active
{
    get { return _active; }
    set
    {
        _active = value;
    }
}

int _dataTypes = 0;

/// <summary>
/// Gets the number of dumped dataTypes.
/// </summary>
/// <value>The data types count.</value>
public int DataTypesCount
{
    get { return _dataTypes; }
}

int _symbols = 0;

/// <summary>
/// Gets the number of dumped symbols
/// </summary>
/// <value>The symbols count.</value>
public int SymbolsCount
{
    get { return _symbols; }
}

/// <summary>
/// Dumps the data type.
/// </summary>
/// <param name="dataType">Data Type.</param>
public void DumpType(IDataType dataType)
{
    WriteLine(string.Format("DataType: {0}, Category: {1}, Size: {2}", dataType.Name, dataType.Category, dataType.Size));

    switch (dataType.Category)
    {
        case DataTypeCategory.Alias:
            IAliasType alias = (IAliasType)dataType;
            WriteLine(GetPrefix(1) + string.Format("Alias BaseType: {0}", alias.BaseTypeName));
            break;

        case DataTypeCategory.Enum:
            //IEnumType<ushort> enumType = (IEnumType<ushort>)dataType;
            IEnumType enumType = (IEnumType)dataType;
            WriteLine(GetPrefix(1) + string.Format("Enum BaseType: {0}", enumType.BaseTypeName));

            foreach (IEnumValue enumValue in enumType.EnumValues)
            {
                WriteLine(GetPrefix(2) + string.Format("Name: {0}, Value: {1}", enumValue.Name, enumValue.Primitive));
            }
            break;
        case DataTypeCategory.Array:
            IArrayType arrayType = (IArrayType)dataType;
            int i = 0;

            foreach (IDimension dim in arrayType.Dimensions)
            {
                WriteLine(GetPrefix(2) + string.Format("{0}: LowerBound: {1}, Elements: {2}", i +
, dim.LowerBound, dim.ElementCount));
            }
            break;
        case DataTypeCategory.Struct:
            IStructType structType = (IStructType)dataType;

            foreach (IMember member in structType.Members)
            {
                WriteLine(GetPrefix(2) + string.Format("Offset {0}: Name: {1}, Type: {2}", member.Offset
, member.InstanceName, member.TypeName));
            }
            break;
    }
}

```



```

        default:
            break;
    }

    foreach (ITypeAttribute attribute in dataType.Attributes)
    {
        WriteLine(GetPrefix(1) + string.Format("{0} : {1} }", attribute.Name, attribute.Value));
    }
    if (!string.IsNullOrEmpty(dataType.Comment))
    {
        WriteLine(GetPrefix(1) + string.Format("Comment: {0}", dataType.Comment));
    }

    IRpcCallableType rpcCallable = dataType as IRpcCallableType;

    if (rpcCallable != null)
    {
        foreach (IRpcMethod rpcMethod in rpcCallable.RpcMethods)
        {
            if (string.IsNullOrEmpty(rpcMethod.Comment))
                WriteLine(GetPrefix(1) + string.Format("Method: {0}", rpcMethod));
            else
                WriteLine(GetPrefix(1) + string.Format("Method: {0}, Comment: {1}", rpcMethod, rpcMethod
.Comment));
        }
    }
    _dataTypes++;
}

///// <summary>
///// Dumps the Datatype to Console
///// </summary>
///// <param name="dataType">DataType.</param>
//public void DumpType(ITcAdsDataType dataType)
//{
//    // Dump the Attributes (PLC Metadata)
//    foreach (ITypeAttribute attribute in dataType.Attributes)
//    {
//        WriteLine(GetPrefix(1) + string.Format("{0} : {1} }", attribute.Name, attribute.Value
));
//    }

//    WriteLine(string.Format("DataType: {0}, Category: {1}, Size: {2}", dataType.Name, dataType
.Category, dataType.Size));

//    if (dataType.BaseType != null)
//    {
//        WriteLine(GetPrefix(1) + string.Format("BaseType: {0}", dataType.BaseType));
//    }

//    switch (dataType.Category)
//    {
//        case DataTypeCategory.Enum:
//            foreach (IEnumValue enumValue in dataType.EnumValues)
//            {
//                WriteLine(GetPrefix(2) + string.Format("Name: {0}, Value: {1}", enumValue.Name, enumVa
lue.Primitive));
//            }
//            break;
//        case DataTypeCategory.Array:
//            int i = 0;
//            foreach (IDimension dim in dataType.Dimensions)
//            {
//                WriteLine(GetPrefix(2) + string.Format("{0}: LowerBound: {1}, Elements: {2}", i+
, dim.LowerBound, dim.ElementCount));
//            }
//            break;
//        case DataTypeCategory.Struct:
//            foreach (ITcAdsSubItem subItem in dataType.SubItems)
//            {
//                WriteLine(GetPrefix(2) + string.Format("Offset {0}: Name: {1}, Type: {2}", subItem.Off
set, subItem.SubItemName, subItem.Name));
//            }
//            break;
//        default:
//            break;
//    }
//    _dataTypes++;
//}

```

```

/// <summary>
/// Dump Symbol
/// </summary>
/// <param name="symbol">The symbol.</param>
/// <param name="level">Output indentation level</param>
public void DumpSymbol(ISymbol symbol, int level)
{
    IDataTypeInfo type = symbol.DataTypeInfo as IDataTypeInfo;

    foreach (ITypeAttribute attribute in symbol.Attributes)
    {
        WriteLine(GetPrefix(level) + string.Format("{0} : {1} }}", attribute.Name, attribute.Value));
    }

    WriteLine(GetPrefix(level) + string.Format("{0} : {1} (IG: 0x{2} IO: 0x{3} size:
{4})", symbol.InstanceName, symbol.TypeName, ((IAdsSymbol)symbol).IndexGroup.ToString("x"), ((IAdsSymbol)symbol).IndexOffset.ToString("x"), symbol.Size));

    if (symbol.Category == DataTypeCategory.Array)
    {
        IArrayInstance arrInstance = (IArrayInstance)symbol;
        IArrayType arrType = (IArrayType)symbol.DataTypeInfo;

        int count = 0;
        level++;

        foreach (ISymbol arrayElement in arrInstance.Elements)
        {
            DumpSymbol(arrayElement, level);
            count++;

            if (count > 20) // Write only the first 20 to limit output
                break;
        }
    }
    else if (symbol.Category == DataTypeCategory.Struct)
    {
        IStructInstance structInstance = (IStructInstance)symbol;
        IStructType structType = (IStructType)symbol.DataTypeInfo;

        level++;

        foreach (ISymbol member in structInstance.MemberInstances)
        {
            DumpSymbol(member, level);
        }
    }
    _symbols++;
}

///// <summary>
///// Dumps the specified Symbol to the Console
///// </summary>
///// <param name="symbol">The symbol.</param>
///// <param name="level">The level.</param>
//public void DumpSymbol(IAdsSymbol2 symbol, int level)
//{
//    // Dump Attributes of the Symbol
//    foreach (ITypeAttribute attribute in symbol.Attributes)
//    {
//        WriteLine(GetPrefix(level) + string.Format("{0} : {1} }}", attribute.Name, attribute.Value));
//    }

//    ITcAdsSymbolBrowser subSymbolProvider = (ITcAdsSymbolBrowser)symbol;

//    // Dump The Symbol
//    WriteLine(GetPrefix(level) + string.Format("{0} : {1} ({2}, IG: 0x{3} IO: 0x{4} size:
{6} subCount:
{5})", symbol.Name, symbol.TypeName, symbol.DataTypeId, symbol.IndexGroup.ToString("x"), symbol.IndexOffset.ToString("x"), subSymbolProvider.SubSymbols.Count, symbol.Size));
//    level++;

//    // Dump all SubSymbols with indentation
//    foreach (IAdsSymbol2 subSymbol in ((ITcAdsSymbolBrowser)symbol).SubSymbols)
//    {
//        DumpSymbol(subSymbol, level);
//    }
//    _symbols++;

```

```

    //}

    /// <summary>
    /// Dump namespace.
    /// </summary>
    /// <param name="ns">The namespace.</param>
    public void DumpNamespace(INamespace<IDataType> ns)
    {
        WriteLine("Namespace: {0}, DataTypes: {1}", ns.Name, ns.DataTypes.Count);

        foreach (IDataType type in ns.DataTypes)
        {
            DumpType(type);
        }
    }

    /// <summary>
    /// Get the indentation prefix
    /// </summary>
    /// <param name="level">The level.</param>
    /// <returns>System.String.</returns>
    public string GetPrefix(int level)
    {
        return "".PadLeft(level * 3);
    }

    /// <summary>
    /// Writes a line to the Console
    /// </summary>
    /// <param name="message">The message.</param>
    public void WriteLine(string message)
    {
        if (Active)
        {
            Console.WriteLine(message);
        }
    }

    /// <summary>
    /// Writes a line to the console
    /// </summary>
    /// <param name="format">The format.</param>
    /// <param name="args">The arguments.</param>
    public void WriteLine(string format, params object[] args)
    {
        if (Active)
        {
            Console.WriteLine(format, args);
        }
    }
}

```

Examples

The following sample shows how to call (Remote Procedures / Methods) with Virtual Symbols

RPC Call in Virtual Mode

```

class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);
        }
    }
}

```

```

        // Get the Symbols (Dynamic Symbols)

        IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

        // Call a Method that has the following signature (within MAIN Program)
        /* {attribute 'TcRpcEnable'}
        METHOD PUBLIC M_Add : INT
        VAR_INPUT
            i1 : INT := 0;
            i2 : INT := 0;
        END_VAR
        */

        short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

        // Call a Method that has no parameter and returns VOID
        main.InvokeRpcMethod("M_Method1", new object[] {});

        //Browsing RpcMethods
        foreach(IRpcMethod method in main.RpcMethods)
        {
            string methodName = method.Name;

            foreach(IRpcMethodParameter parameter in method.Parameters)
            {
                string parameterName = parameter.Name;
                string parameterType = parameter.TypeName;
            }
        }
    }
}

```

Examples

The following sample shows how to call (Remote Procedures / Methods) with Dynamic Symbols.

RPC Call in Dynamic Mode

```

namespace Sample
{
    using System;
    using System.Diagnostics;
    using System.Threading;
    using TwinCAT;
    using TwinCAT.Ads;
    using TwinCAT.Ads.TypeSystem;
    using TwinCAT.Ads.ValueAccess;
    using TwinCAT.TypeSystem;
    using TwinCAT.TypeSystem.Generic;

    class RpcCallDynamicProgram
    {
        /// <summary>
        /// Defines the entry point of the application.
        /// </summary>
        /// <param name="args">The arguments.</param>
        static async void Main(string[] args)
        {
            // Get the AdsAddress from command-line arguments
            AmsAddress address = ArgParser.Parse(args);
            CancellationTokenSource cancelSource = new CancellationTokenSource();
            CancellationToken cancel = cancelSource.Token;

            using (AdsClient client = new AdsClient())
            {
                // Connect to the target device
                client.Connect(address);

                SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.DynamicTree);
                ISymbolLoader dynLoader = SymbolLoaderFactory.Create(client, settings);

                // Get the Symbols (Dynamic Symbols)
                ResultDynamicSymbols resultGetSymbols = await ((IDynamicSymbolLoader)dynLoader).GetDynamicSymbolsAsync(cancel);
                dynamic symbols = resultGetSymbols.Symbols;
                dynamic main = symbols.Main; // Gets the MAIN Instance of the PLC Program
            }
        }
    }
}

```

```

// Call a Method that has the following signature (within MAIN Program)

/* {attribute 'TcRpcEnable'}
METHOD PUBLIC M_Add : INT
VAR_INPUT
i1 : INT := 0;
i2 : INT := 0;
END_VAR
*/

short result = main.M_Add(3,4); // Synchronous Call

// Call a Method that has no parameter and returns VOID
main.M_Method1(); // Synchronous call

//Browsing Rpc Methods
foreach (IRpcMethod method in main.RpcMethods)
{
    string methodName = method.Name;

    foreach (IRpcMethodParameter parameter in method.Parameters)
    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
}
}
}

```

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

[TwinCAT.Ads.TypeSystem.SymbolLoaderFactory](#) [► 1923]

6.2.5.2.9 AdsConnection.CreateVariableHandle Method

Generates a unique handle for an ADS variable.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public uint CreateVariableHandle(
    string variableName
)

```

Parameters

variableName	Type: System.String Name of the ADS variable
--------------	---

Return Value

Type: [UInt32](#)

The handle of the ADS Variable.

Implements

[IAdsHandle.CreateVariableHandle\(String\)](#) [► 956]

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.10 AdsConnection.CreateVariableHandleAsync Method

Determines the Symbol handle by its instance path asynchronously.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultHandle> CreateVariableHandleAsync(
    string variableName,
    CancellationToken cancel
)
```

Parameters

variableName	Type: System.String Name of the variable.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultHandle](#) [► 1136].

A task that represents the asynchronous 'CreateVariableHandle' operation. The [ResultHandle](#) [► 1136] parameter contains the variable handle ([Handle](#) [► 1138]) and the [ErrorCode](#) [► 1120] after execution.

Implements

[IAdsHandle.CreateVariableHandleAsync\(String, CancellationToken\)](#) [► 956]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [► 67]	

Remarks

It is a good practice to release all variable handles after use to regain internal resources in the TwinCAT subsystem. The composite method to this [CreateVariableHandleAsync\(String, CancellationToken\)](#) is the [DeleteVariableHandleAsync\(UInt32, CancellationToken\)](#) [► 497]

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

[AdsConnection.DeleteVariableHandleAsync\(UInt32, CancellationToken\)](#) [► 497]

[AdsConnection.TryCreateVariableHandle\(String, UInt32.\)](#) [[▶ 572](#)]

[AdsConnection.CreateVariableHandle\(String\)](#) [[▶ 493](#)]

6.2.5.2.11 **AdsConnection.DeleteDeviceNotification Method**

Deletes an existing notification.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void DeleteDeviceNotification(
    uint notificationHandle
)
```

Parameters

notificationHandle	Type: System.UInt32 Handle of the notification.
--------------------	--

Implements

[IAdsNotifications.DeleteDeviceNotification\(UInt32\)](#) [[▶ 983](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.12 **AdsConnection.DeleteDeviceNotificationAsync Method**

Deletes a registered notification asynchronously.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAds> DeleteDeviceNotificationAsync(
    uint notificationHandle,
    CancellationToken cancel
)
```

Parameters

notificationHandle	Type: System.UInt32 Notification handle.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds](#) [► 1116].

A task that represents the asynchronous 'DeleteDeviceNotification' operation. The [ErrorCode](#) [► 1120] property contains the ADS error code after execution.

Implements

[IAdsNotifications.DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [► 984]

Remarks

This is the complementary method to [AddDeviceNotificationAsync Overload](#) [► 973] overloads and should be called when the notification is not needed anymore the free TwinCAT realtime resources.

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

[AddDeviceNotificationAsync Overload](#) [► 973]

[IAdsNotifications.AdsNotification](#) [► 991]

[TryAddDeviceNotification Overload](#) [► 985]

[AddDeviceNotification Overload](#) [► 971]

6.2.5.2.13 AdsConnection.DeleteVariableHandle Method

Releases the handle of a ADS variable again.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void DeleteVariableHandle(
    uint variableHandle
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable
----------------	---

Implements

[IAdsHandle.DeleteVariableHandle\(UInt32\)](#) [► 957]

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.14 AdsConnection.DeleteVariableHandleAsync Method

Releases the handle of a ADS variable again (asynchronously)

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAds> DeleteVariableHandleAsync(
    uint variableHandle,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds](#) [[▶ 1116](#)].

A task that represents the asynchronous 'ReadState' operation. The [ResultAds](#) [[▶ 1116](#)] parameter contains the [ErrorCode](#) [[▶ 1120](#)] of the ADS communication after execution.

Implements

[IAdsHandle.DeleteVariableHandleAsync\(UInt32, CancellationToken\)](#) [[▶ 958](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Remarks

It is a good practice to release all variable handles after use to regain internal resources in the TwinCAT subsystem. The composite method to this [DeleteVariableHandleAsync\(UInt32, CancellationToken\)](#) [[▶ 958](#)] is the [CreateVariableHandleAsync\(String, CancellationToken\)](#) [[▶ 956](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[AdsConnection.CreateVariableHandleAsync\(String, CancellationToken\)](#) [[▶ 494](#)]

[AdsConnection.TryDeleteVariableHandle\(UInt32\)](#) [[▶ 573](#)]

[AdsConnection.DeleteVariableHandle\(UInt32\)](#) [[▶ 496](#)]

6.2.5.2.15 **AdsConnection.Disconnect Method**

Disconnects this [IConnection](#) [[▶ 79](#)].

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Disconnect()
```

Return Value

Type: [Boolean](#)

true if the [AdsConnection](#) [[▶ 412](#)] is disconnected, false if the connection was already disconnected.

Implements

[IConnection.Disconnect](#). [[▶ 84](#)]

Exceptions

Exception	Condition
ObjectDisposedException	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.16 **AdsConnection.Dispose Method**

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Dispose()
```

Implements

[IDisposable.Dispose](#).

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.17 AdsConnection.InvokeRpcMethod Method

Overload List

	Name	Description
	InvokeRpcMethod(String, .Object.) [▶ 499]	Invokes the specified RPC Method
	InvokeRpcMethod(String, .Object., .Object.) [▶ 501]	Invokes the specified RPC Method
	InvokeRpcMethod(String, .Object., AnyTypeSpecifier) [▶ 504]	Invokes the specified RPC Method
	InvokeRpcMethod(String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier., .Object.) [▶ 502]	Invokes the specified RPC Method

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.17.1 AdsConnection.InvokeRpcMethod Method (String, String, .Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object? InvokeRpcMethod(
    string symbolPath,
    string methodName,
    Object[]? inParameters
)
```

Parameters

symbolPath	Type: System.String The symbol path.
methodName	Type: System.String The method name.

inParameters	Type: <u>System.Object</u> . The input parameters or NULL
--------------	--

Return Value

Type: [Object](#)

The return value of the Method (as object).

Remarks

This method only supports primitive data types as inParameters. Any available outparameters will be ignored. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[InvokeRpcMethod Overload](#) [[▶ 499](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.17.2 AdsConnection.InvokeRpcMethod Method (String, String, .Object., .Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object? InvokeRpcMethod(
    string symbolPath,
    string methodName,
    Object[]? inParameters,
    out Object[]? outParameters
)
```

Parameters

symbolPath	Type: System.String The symbol path.
methodName	Type: System.String The method name.
inParameters	Type: .System.Object. The input parameters or NULL
outParameters	Type: .System.Object.. The output parameters.

Return Value

Type: [Object](#)

The return value of the Method (as object).

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [[▶ 2094](#)] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);
    }
}
```

```

using (AdsClient client = new AdsClient())
{
    //client.Synchronize = false;

    // Connect to the target device
    client.Connect(address);

    SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
    ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

    // Get the Symbols (Dynamic Symbols)

    IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
the PLC Program

    // Call a Method that has the following signature (within MAIN Program)
    /* {attribute 'TcRpcEnable'}
    METHOD PUBLIC M_Add : INT
    VAR_INPUT
        i1 : INT := 0;
        i2 : INT := 0;
    END_VAR
    */

    short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

    // Call a Method that has no parameter and returns VOID
    main.InvokeRpcMethod("M_Method1", new object[] {});

    //Browsing RpcMethods
    foreach(IRpcMethod method in main.RpcMethods)
    {
        string methodName = method.Name;

        foreach(IRpcMethodParameter parameter in method.Parameters)
        {
            string parameterName = parameter.Name;
            string parameterType = parameter.TypeName;
        }
    }
}

```

Reference

[AdsConnection Class \[► 412\]](#)

[InvokeRpcMethod Overload \[► 499\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.17.3 AdsConnection.InvokeRpcMethod Method (String, String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object..)

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public Object? InvokeRpcMethod(
    string symbolPath,
    string methodName,
    Object[]? inParameters,
    AnyTypeSpecifier[]? outSpecifiers,
    AnyTypeSpecifier? retSpecifier,
    out Object[]? outParameters
)

```

Parameters

symbolPath	Type: <u>System.String</u> The symbol path.
methodName	Type: <u>System.String</u> The method name.
inParameters	Type: <u>System.Object</u> . The parameters.
outSpecifiers	Type: <u>.TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094]</u> . The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: <u>TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094]</u> The ret specifier (specifying the return value) or NULL.
outParameters	Type: <u>System.Object</u> .. The out parameters.

Return Value

Type: Object
The return value of the Method (as object).

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});
        }
    }
}
```

```

// Call a Method that has no parameter and returns VOID
main.InvokeRpcMethod("M_Method1", new object[] {});

//Browsing RpcMethods
foreach(IRpcMethod method in main.RpcMethods)
{
    string methodName = method.Name;

    foreach(IRpcMethodParameter parameter in method.Parameters)
    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
}
}

```

Reference

[AdsConnection Class](#) [► 412]

[InvokeRpcMethod Overload](#) [► 499]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.17.4 AdsConnection.InvokeRpcMethod Method (String, String, .Object., AnyTypeSpecifier)

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public Object? InvokeRpcMethod(
    string symbolPath,
    string methodName,
    Object[]? inParameters,
    AnyTypeSpecifier? retSpecifier
)

```

Parameters

symbolPath	Type: System.String The symbol path.
methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [► 2094] The ret specifier (specifying the return value) or NULL.

Return Value

Type: [Object](#)

The return value of the Method (as object).

Exceptions

Exception	Condition
<u>NotImplementedException</u>	

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters and retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (retSpecifier) is not necessary and should not be set.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */







            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```

Reference[AdsConnection Class \[► 412\]](#)[InvokeRpcMethod Overload \[► 499\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.5.2.18 AdsConnection.InvokeRpcMethodAsync Method****Overload List**

	Name	Description
 	InvokeRpcMethodAsync(String, String, .Object., CancellationToken) [► 506]	Invokes the specified RPC Method asynchronously
 	InvokeRpcMethodAsync(IRpcCallableInstance, IRpcMethod, .Object., .AnyTypeSpecifier, .AnyTypeSpecifier, CancellationToken) [► 510]	Invokes the specified RPC Method asynchronously
 	InvokeRpcMethodAsync(String, String, .Object., .AnyTypeSpecifier, .AnyTypeSpecifier, CancellationToken) [► 508]	Invokes the specified RPC Method asynchronously

Reference[AdsConnection Class \[► 412\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.5.2.18.1 AdsConnection.InvokeRpcMethodAsync Method (String, String, .Object., CancellationToken)**

Invokes the specified RPC Method asynchronously

Namespace: [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public Task<ResultRpcMethod> InvokeRpcMethodAsync (
    string symbolPath,
    string methodName,
```

```
Object[]? inParameters,
CancellationTokn cancel
)
```

Parameters

symbolPath	Type: System.String The symbol/Instance path of the symbol.
methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
cancel	Type: System.Threading.CancellationTokn The cancellation token

Return Value

Type: [Task.ResultRpcMethod](#) [▶ 1176].

A task that represents the asynchronous 'InvokeRpcMethod' operation. The [ResultRpcMethod](#) [▶ 1176] results contains the return value together with the output parameters.

Implements

[IAdsRpcInvoke.InvokeRpcMethodAsync\(String, String, .Object., CancellationTokn\)](#) [▶ 1021]

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [▶ 2094] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
```

```

        i2 : INT := 0;
    END_VAR
*/

short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

// Call a Method that has no parameter and returns VOID
main.InvokeRpcMethod("M_Method1", new object[] {});

//Browsing RpcMethods
foreach(IRpcMethod method in main.RpcMethods)
{
    string methodName = method.Name;

    foreach(IRpcMethodParameter parameter in method.Parameters)
    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
}
}

```

Reference

[AdsConnection Class \[▶ 412\]](#)

[InvokeRpcMethodAsync Overload \[▶ 506\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.5.2.18.2 AdsConnection.InvokeRpcMethodAsync Method (String, String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, CancellationToken)

Invokes the specified RPC Method asynchronously

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public Task<ResultRpcMethod> InvokeRpcMethodAsync(
    string symbolPath,
    string methodName,
    Object[]? inParameters,
    AnyTypeSpecifier[]? outSpecifiers,
    AnyTypeSpecifier? retSpecifier,
    CancellationToken cancel
)

```

Parameters

symbolPath	Type: System.String The symbol/Instance path of the symbol.
methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094] . The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094] The ret specifier (specifying the return value) or NULL.

cancel	Type: System.Threading.CancellationToken The cancellation token
--------	--

Return Value

Type: [Task.ResultRpcMethod](#) [▶ 1176].

A task that represents the asynchronous 'InvokeRpcMethod' operation. The [ResultRpcMethod](#) [▶ 1176] results contains the return value together with the output parameters.

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set. [ReturnValue](#) [▶ 1179] and the [ErrorCode](#) [▶ 1120] of the ADS communication after execution.

Implements

[IAdsRpcInvoke.InvokeRpcMethodAsync\(String, String, .Object, .AnyTypeSpecifier, AnyTypeSpecifier, CancellationToken\)](#) [▶ 1022]

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach (IRpcMethod method in main.RpcMethods)
            {
```

```

        string methodName = method.Name;

        foreach(IRpcMethodParameter parameter in method.Parameters)
        {
            string parameterName = parameter.Name;
            string parameterType = parameter.TypeName;
        }
    }
}

```

Reference

[AdsConnection Class](#) [► 412]

[InvokeRpcMethodAsync Overload](#) [► 506]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.18.3 AdsConnection.InvokeRpcMethodAsync Method (IRpcCallableInstance, IRpcMethod, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, CancellationToken)

Invokes the specified RPC Method asynchronously

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public Task<ResultRpcMethod> InvokeRpcMethodAsync(
    IRpcCallableInstance symbol,
    IRpcMethod method,
    Object[]? inParameters,
    AnyTypeSpecifier[]? outSpecifiers,
    AnyTypeSpecifier? retSpecifier,
    CancellationToken cancel
)

```

Parameters

symbol	Type: TwinCAT.TypeSystem.IRpcCallableInstance [► 2606] The RPC callable symbol."
method	Type: TwinCAT.TypeSystem.IRpcMethod [► 2625] The method.
inParameters	Type: .System.Object . The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [► 2094]. The out specifiers.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [► 2094] The ret specifier.
cancel	Type: System.Threading.CancellationToken The cancellation token

Return Value

Type: [Task.ResultRpcMethod](#) [► 1176].

A task that represents the asynchronous 'InvokeRpcMethod' operation. The [ResultRpcMethod](#) [► 1176] results contains the return value together with the output parameters.

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters are allowed to be empty or NULL. [ReturnValue \[▶ 1179\]](#) and the [ErrorCode \[▶ 1120\]](#) of the ADS communication after execution.

Implements

[IAdsRpcInvoke.InvokeRpcMethodAsync\(IRpcCallableInstance, IRpcMethod, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, CancellationToken\) \[▶ 1024\]](#)

Exceptions

Exception	Condition
ObjectDisposedException	this.Name
ClientNotConnectedException [▶ 67]	

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```

class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)

```

```

    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
}
}

```

Reference




[AdsConnection Class](#) [► 412]

[InvokeRpcMethodAsync Overload](#) [► 506]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.19 AdsConnection.Read Method

Overload List

	Name	Description
	Read(UInt32, Memory.Byte.) [► 514]	Reads the value from the symbol that is represented by the handle.
	Read(UInt32, UInt32, Memory.Byte.) [► 515]	Reads data synchronously from an ADS device and writes it to the given readBuffer
	Read(UInt32, UInt32, Memory.Byte., Int32.) [► 516]	Reads data synchronously from an ADS device and writes it to the given stream.

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.19.1 AdsConnection.Read Method (UInt32, Memory`1)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```

public int Read(
    uint variableHandle,
    Memory buffer
)

```

Parameters

variableHandle Type: [System.UInt32](#)

buffer Type: [Memory](#)

Return ValueType: [Int32](#)**Reference**[AdsConnection Class](#) [[▶ 412](#)][Read Overload](#) [[▶ 512](#)][TwinCAT.Ads Namespace](#) [[▶ 179](#)]**6.2.5.2.19.2 AdsConnection.Read Method (UInt32, UInt32, Memory`1)****Namespace:** [TwinCAT.Ads](#) [[▶ 179](#)]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14**Syntax****C#**

```
public int Read(  
    uint indexGroup,  
    uint indexOffset,  
    Memory readBuffer  
)
```

ParametersindexGroup Type: [System.UInt32](#)indexOffset Type: [System.UInt32](#)readBuffer Type: [Memory](#)**Return Value**Type: [Int32](#)**Reference**[AdsConnection Class](#) [[▶ 412](#)][Read Overload](#) [[▶ 512](#)][TwinCAT.Ads Namespace](#) [[▶ 179](#)]**6.2.5.2.19.3 AdsConnection.Read Method (UInt32, UInt32, Memory`1, Void)****Namespace:** [TwinCAT.Ads](#) [[▶ 179](#)]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14**Syntax****C#**

```
public int Read(  
    uint indexGroup,  
    uint indexOffset,
```

```
Memory readBuffer,
void timeout
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
readBuffer	Type: Memory
timeout	Type: System.Void

Return Value

Type: [Int32](#)

Reference

[AdsConnection Class](#) [► 412]

[Read Overload](#) [► 512]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.19.4 AdsConnection.Read Method (UInt32, Memory.Byte.)

Reads the value from the symbol that is represented by the handle.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Read(
    uint variableHandle,
    Memory<byte> buffer
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
buffer	Type: System.Memory.Byte. The buffer.

Return Value

Type: [Int32](#)
System.Int32.

Implements

[IAdsHandle.Read\(UInt32, Memory.Byte.\)](#) [► 959]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[Read Overload](#) [[▶ 512](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.19.5 **AdsConnection.Read Method (UInt32, UInt32, Memory.Byte.)**

Reads data synchronously from an ADS device and writes it to the given readBuffer

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Read(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
readBuffer	Type: System.Memory.Byte . Memory location, where to read the data.

Return Value

Type: [Int32](#)

Number of successfully returned (read) data bytes.

Implements

[IAdsReadWrite2.Read\(UInt32, UInt32, Memory.Byte.\)](#) [[▶ 1002](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[Read Overload](#) [[▶ 512](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.19.6 AdsConnection.Read Method (UInt32, UInt32, Memory.Byte., Int32)

Reads data synchronously from an ADS device and writes it to the given stream.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Read(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    int timeout
)
```

Parameters

indexGroup	Type: System.UInt32 The index group number of the requested ADS service.
indexOffset	Type: System.UInt32 The index offset number of the requested ADS service.
readBuffer	Type: System.Memory.Byte. The read buffer.
timeout	Type: System.Int32 The timeout.

Return Value

Type: [Int32](#)

Number of successfully returned data bytes.

Implements

[IAdsReadWriteTimeoutAccess.Read\(UInt32, UInt32, Memory.Byte., Int32\)](#) [[▶ 1006](#)]

Reference




[AdsConnection Class](#) [[▶ 412](#)]







[Read Overload](#) [[▶ 512](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.20 AdsConnection.ReadAny Method

Overload List

	Name	Description
	ReadAny.T.(UInt32) [▶ 517]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, Type) [▶ 519]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, .Int32.) [▶ 518]	Reads data synchronously from an ADS device and writes it to an object.

	Name	Description
	ReadAny.T.(UInt32, UInt32) [► 519]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, Type, .Int32.) [► 520]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type) [► 522]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, UInt32, .Int32.) [► 521]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type, .Int32.) [► 523]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type, .Int32., Int32) [► 524]	Reads data synchronously from an ADS device and writes it to an object.

Reference

[AdsConnection Class \[► 412\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.20.1 AdsConnection.ReadAny.T. Method (UInt32)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T ReadAny<T>(
    uint variableHandle
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
----------------	--

Type Parameters

T	The type of the value to read.
---	--------------------------------

Return Value

Type: T
The value of the read symbol.

Implements

[IAdsAnyAccess.ReadAny.T.\(UInt32\) \[► 810\]](#)

Reference

[AdsConnection Class](#) [► 412]

[ReadAny Overload](#) [► 516]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.20.2 AdsConnection.ReadAny.T. Method (UInt32, .Int32.)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T ReadAny<T>(
    uint variableHandle,
    int[]? args
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
args	Type: .System.Int32 . Additional arguments.

Type Parameters

T	The type of the value to read.
---	--------------------------------

Return Value

Type: T

The value of the read symbol.

Implements

[IAdsAnyAccess.ReadAny.T.\(UInt32, .Int32.\)](#) [► 811]

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [► 2927].
string[]	
Array	

Reference

[AdsConnection Class](#) [► 412]

[ReadAny Overload \[► 516\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.20.3 **AdsConnection.ReadAny Method (UInt32, Type)**

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object ReadAny(  
    uint variableHandle,  
    Type type  
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
type	Type: System.Type Type of the object to be read.

Return Value

Type: [Object](#)
The read object.

Implements

[IAdsAnyAccess.ReadAny\(UInt32, Type\) \[► 812\]](#)

Reference

[AdsConnection Class \[► 412\]](#)

[ReadAny Overload \[► 516\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.20.4 **AdsConnection.ReadAny.T. Method (UInt32, UInt32)**

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T ReadAny<T>(  
    uint indexGroup,  
    uint indexOffset  
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.

Type Parameters

T	The type of the object to be read.
---	------------------------------------

Return Value

Type: T
The read value.

Implements

[IAdsAnyAccess.ReadAny.T.\(UInt32, UInt32\)](#) [[▶ 812](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[ReadAny Overload](#) [[▶ 516](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.20.5 AdsConnection.ReadAny Method (UInt32, Type, .Int32.)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Object ReadAny(
    uint variableHandle,
    Type type,
    int[]? args
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
type	Type: System.Type Type of the object to be read.
args	Type: .System.Int32 . Additional arguments.

Return Value

Type: [Object](#)
The read value.

Implements

[IAdsAnyAccess.ReadAny\(UInt32, Type, .Int32.\)](#) [▶ 813]

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference

[AdsConnection Class](#) [▶ 412]

[ReadAny Overload](#) [▶ 516]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.20.6 AdsConnection.ReadAny.T. Method (UInt32, UInt32, .Int32.)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T ReadAny<T>(
    uint indexGroup,
    uint indexOffset,
    int[]? args
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
args	Type: .System.Int32 . Additional arguments.

Type Parameters

T	The type of the object to be read.
---	------------------------------------

Return Value

Type: T
The read value.

Implements

[IAdsAnyAccess.ReadAny.T.\(UInt32, UInt32, .Int32.\)](#) [▶ 814]

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference

[AdsConnection Class](#) [▶ 412]

[ReadAny Overload](#) [▶ 516]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.20.7 AdsConnection.ReadAny Method (UInt32, UInt32, Type)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object ReadAny(
    uint indexGroup,
    uint indexOffset,
    Type type
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
type	Type: System.Type Type of the object to be read.

Return Value

Type: [Object](#)
The read value.

Implements

[IAdsAnyAccess.ReadAny\(UInt32, UInt32, Type\)](#) [▶ 815]

Reference

[AdsConnection Class](#) [▶ 412]

[ReadAny Overload](#) [▶ 516]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.20.8 AdsConnection.ReadAny Method (UInt32, UInt32, Type, .Int32.)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object ReadAny(
    uint indexGroup,
    uint indexOffset,
    Type type,
    int[]? args
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
type	Type: System.Type Type of the object to be read.
args	Type: .System.Int32 . Additional arguments.

Return Value

Type: [Object](#)
The read value.

Implements

[IAdsAnyAccess.ReadAny\(UInt32, UInt32, Type, .Int32.\)](#) [▶ 815]

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference[AdsConnection Class](#) [► 412][ReadAny Overload](#) [► 516][TwinCAT.Ads Namespace](#) [► 179]**6.2.5.2.20.9 AdsConnection.ReadAny Method (UInt32, UInt32, Type, .Int32., Int32)**

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Object ReadAny(
    uint indexGroup,
    uint indexOffset,
    Type type,
    int[] args,
    int timeout
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
type	Type: System.Type Type of the object to be read.
args	Type: .System.Int32. Additional arguments.
timeout	Type: System.Int32 The timeout.

Return Value

Type: [Object](#)

The Value of the data marshalled to the specified type.

Implements

[IAdsReadWriteTimeoutAccess.ReadAny\(UInt32, UInt32, Type, .Int32., Int32\)](#) [► 1007]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [► 67]	

Remarks

If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args. The type is limited to Primitive types ('AnyType').

Reference









[AdsConnection Class \[▶ 412\]](#)

[ReadAny Overload \[▶ 516\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.5.2.21 AdsConnection.ReadAnyAsync Method

Overload List

	Name	Description
	ReadAnyAsync.T. (UInt32, CancellationToken) [▶ 526]	Reads data synchronously from an ADS device.
	ReadAnyAsync(UInt32, Type, CancellationToken) [▶ 528]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T. (UInt32, .Int32, CancellationToken) [▶ 527]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T. (UInt32, UInt32, CancellationToken) [▶ 529]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, Type, .Int32, CancellationToken) [▶ 529]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, UInt32, Type, CancellationToken) [▶ 532]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T. (UInt32, UInt32, .Int32, CancellationToken) [▶ 531]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, UInt32,	Reads data asynchronously from an ADS device and writes it to an object.

	Name	Description
	Type, .Int32 , CancellationToken) [► 532]	

Reference

[AdsConnection Class \[► 412\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.21.1 **AdsConnection.ReadAnyAsync.T. Method (UInt32, CancellationTokentoken)**

Reads data synchronously from an ADS device.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultValue<T>> ReadAnyAsync<T>(
    uint variableHandle,
    CancellationTokentoken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable/symbol handle.
cancel	Type: System.Threading.CancellationTokentoken The cancellation token.

Type Parameters

T	The Type of the value to be read.
---	-----------------------------------

Return Value

Type: [Task.ResultValue \[► 1181\].T](#)..

A task that represents the asynchronous read operation. The [ResultAnyValue \[► 1129\]](#) parameter contains the read value ([Value \[► 1185\]](#)) and the [ErrorCode \[► 1120\]](#) after execution.

Implements

[IAdsAnyAccess.ReadAnyAsync.T.\(UInt32, CancellationTokentoken\) \[► 817\]](#)

Remarks

As object types only primitive types are supported.

Reference

[AdsConnection Class \[► 412\]](#)

[ReadAnyAsync Overload \[► 525\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

Also see about this

 [ResultAnyValue.Value Property](#) [[▶ 1131](#)]

6.2.5.2.21.2 AdsConnection.ReadAnyAsync.T. Method (UInt32, .Int32., CancellationTokentoken)

Reads data asynchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultValue<T>> ReadAnyAsync<T>(
    uint variableHandle,
    int[]? args,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
args	Type: .System.Int32. Additional arguments.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Type Parameters

T	Type of the object to be read
---	-------------------------------

Return Value

Type: [Task.ResultValue](#) [[▶ 1181](#)].T..

A task that represents the asynchronous read operation. The [ResultValue.TValue.](#) [[▶ 1181](#)] parameter contains the read value ([Value](#) [[▶ 1185](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsAnyAccess.ReadAnyAsync.T.\(UInt32, .Int32., CancellationTokentoken\)](#) [[▶ 818](#)]

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference[AdsConnection Class](#) [► 412][ReadAnyAsync Overload](#) [► 525][TwinCAT.Ads Namespace](#) [► 179]**6.2.5.2.21.3 AdsConnection.ReadAnyAsync Method (UInt32, Type, CancellationToken)**

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [► 179]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public Task<ResultAnyValue> ReadAnyAsync(
    uint variableHandle,
    Type type,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable/symbol handle.
type	Type: System.Type Type of the object to be read.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return ValueType: [Task.ResultAnyValue](#) [► 1129].

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [► 1129] parameter contains the read value ([Value](#) [► 1185]) and the [ErrorCode](#) [► 1120] after execution.

Implements[IAdsAnyAccess.ReadAnyAsync\(UInt32, Type, CancellationToken\)](#) [► 819]**Remarks**

As object types only primitive types are supported.

Reference[AdsConnection Class](#) [► 412][ReadAnyAsync Overload](#) [► 525][TwinCAT.Ads Namespace](#) [► 179]**Also see about this** [ResultAnyValue.Value Property](#) [► 1131]

6.2.5.2.21.4 **AdsConnection.ReadAnyAsync.T. Method (UInt32, UInt32, CancellationToken)**

Reads data asynchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultValue<T>> ReadAnyAsync<T>(
    uint indexGroup,
    uint indexOffset,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Type Parameters

T

Return Value

Type: [Task.ResultValue](#) [[▶ 1181](#)].T..
The asynchronous result.

Return Value

Type: [Task.ResultValue](#) [[▶ 1181](#)].T..

A task that represents the asynchronous read operation. The [ResultValue.TValue.](#) [[▶ 1181](#)] parameter contains the read value ([Value](#) [[▶ 1185](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsAnyAccess.ReadAnyAsync.T.\(UInt32, UInt32, CancellationToken\)](#) [[▶ 820](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[ReadAnyAsync Overload](#) [[▶ 525](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.21.5 **AdsConnection.ReadAnyAsync Method (UInt32, Type, .Int32., CancellationToken)**

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAnyValue> ReadAnyAsync(
    uint variableHandle,
    Type type,
    int[]? args,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
type	Type: System.Type Type of the object to be read.
args	Type: .System.Int32 . Additional arguments.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAnyValue](#) [► 1129].

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [► 1129] parameter contains the read value ([Value](#) [► 1185]) and the [ErrorCode](#) [► 1120] after execution.

Implements

[IAdsAnyAccess.ReadAnyAsync\(UInt32, Type, .Int32., CancellationToken\)](#) [► 821]

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [► 2927].
string[]	
Array	

Reference

[AdsConnection Class](#) [► 412]

[ReadAnyAsync Overload](#) [► 525]

[TwinCAT.Ads Namespace](#) [► 179]

Also see about this

 [ResultAnyValue.Value Property](#) [► 1131]

6.2.5.2.21.6 **AdsConnection.ReadAnyAsync.T. Method (UInt32, UInt32, .Int32., CancellationToken)**

Reads data asynchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultValue<T>> ReadAnyAsync<T>(
    uint indexGroup,
    uint indexOffset,
    int[]? args,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
args	Type: .System.Int32 . Additional arguments.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Type Parameters

T	The type of the result value.
---	-------------------------------

Return Value

Type: [Task.ResultValue](#) [[▶ 1181](#)].T..

A task that represents the asynchronous read operation. The [ResultValue.TValue.](#) [[▶ 1181](#)] parameter contains the read value ([Value](#) [[▶ 1185](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsAnyAccess.ReadAnyAsync.T.\(UInt32, UInt32, .Int32., CancellationToken\)](#) [[▶ 822](#)]

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[ReadAnyAsync Overload \[► 525\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.21.7 **AdsConnection.ReadAnyAsync Method (UInt32, UInt32, Type, CancellationToken)**

Reads data asynchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAnyValue> ReadAnyAsync (
    uint indexGroup,
    uint indexOffset,
    Type type,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
type	Type: System.Type Type of the object to be read.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAnyValue \[► 1129\]](#).

A task that represents the asynchronous read operation. The [ResultAnyValue \[► 1129\]](#) parameter contains the read value ([Value \[► 1185\]](#)) and the [ErrorCode \[► 1120\]](#) after execution.

Implements

[IAdsAnyAccess.ReadAnyAsync\(UInt32, UInt32, Type, CancellationToken\) \[► 823\]](#)

Reference

[AdsConnection Class \[► 412\]](#)

[ReadAnyAsync Overload \[► 525\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

Also see about this

 [ResultAnyValue.Value Property \[► 1131\]](#)

6.2.5.2.21.8 **AdsConnection.ReadAnyAsync Method (UInt32, UInt32, Type, .Int32., CancellationToken)**

Reads data asynchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAnyValue> ReadAnyAsync(
    uint indexGroup,
    uint indexOffset,
    Type type,
    int[]? args,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
type	Type: System.Type Type of the object to be read.
args	Type: .System.Int32 . Additional arguments.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAnyValue](#) [[▶ 1129](#)].

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [[▶ 1129](#)] parameter contains the read value ([Value](#) [[▶ 1185](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsAnyAccess.ReadAnyAsync\(UInt32, UInt32, Type, .Int32., CancellationToken\)](#) [[▶ 823](#)]

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[ReadAnyAsync Overload](#) [[▶ 525](#)]



[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

 [ResultAnyValue.Value Property](#) [[▶ 1131](#)]

6.2.5.2.22 AdsConnection.ReadAnyString Method

Overload List

	Name	Description
	ReadAnyString(UInt32, Int32, Encoding) [▶ 534]	Reads a string from the specified symbol/variable.
	ReadAnyString(UInt32, UInt32, Int32, Encoding) [▶ 535]	Reads as string from a specified address.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.22.1 AdsConnection.ReadAnyString Method (UInt32, Int32, Encoding)

Reads a string from the specified symbol/variable.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string ReadAnyString(
    uint variableHandle,
    int len,
    Encoding encoding
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
len	Type: System.Int32 The length.
encoding	Type: System.Text.Encoding The encoding.

Return Value

Type: [String](#)
The string value.

Implements

[IAdsAnyAccess.ReadAnyString\(UInt32, Int32, Encoding\)](#) [[▶ 825](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[ReadAnyString Overload](#) [[▶ 534](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.22.2 AdsConnection.ReadAnyString Method (UInt32, UInt32, Int32, Encoding)

Reads as string from a specified address.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string ReadAnyString(
    uint indexGroup,
    uint indexOffset,
    int len,
    Encoding encoding
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
len	Type: System.Int32 The string length to be read.
encoding	Type: System.Text.Encoding The encoding.

Return Value

Type: [String](#)
System.String.

Implements

[IAdsAnyAccess.ReadAnyString\(UInt32, UInt32, Int32, Encoding\)](#) [[▶ 825](#)]

Exceptions

Exception	Condition
ObjectDisposedException	

Exception	Condition
ClientNotConnectedException [► 67]	

Reference



[AdsConnection Class \[► 412\]](#)

[ReadAnyString Overload \[► 534\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.23 AdsConnection.ReadAnyStringAsync Method

Overload List

	Name	Description
	ReadAnyStringAsync(UInt32, Int32, Encoding, CancellationToken) [► 536]	Reads a string asynchronously from the specified symbol/variable
	ReadAnyStringAsync(UInt32, UInt32, Int32, Encoding, CancellationToken) [► 537]	read any string as an asynchronous operation.

Reference

[AdsConnection Class \[► 412\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.23.1 AdsConnection.ReadAnyStringAsync Method (UInt32, Int32, Encoding, CancellationToken)

Reads a string asynchronously from the specified symbol/variable

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAnyValue> ReadAnyStringAsync (
    uint variableHandle,
    int len,
    Encoding encoding,
    CancellationToken cancel
)
```


Parameters

variableHandle	Type: System.UInt32 The variable handle.
len	Type: System.Int32 The length.
encoding	Type: System.Text.Encoding The encoding.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAnyValue](#) [[▶ 1129](#)].

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [[▶ 1129](#)] parameter contains the read string ([Value](#) [[▶ 1185](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsAnyAccess.ReadAnyStringAsync\(UInt32, Int32, Encoding, CancellationToken\)](#) [[▶ 827](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[ReadAnyStringAsync Overload](#) [[▶ 536](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

 [ResultAnyValue.Value Property](#) [[▶ 1131](#)]

6.2.5.2.23.2 **AdsConnection.ReadAnyStringAsync Method (UInt32, UInt32, Int32, Encoding, CancellationToken)**

read any string as an asynchronous operation.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultAnyValue> ReadAnyStringAsync (
    uint indexGroup,
    uint indexOffset,
    int len,
    Encoding encoding,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
len	Type: System.Int32 The string length to be read.
encoding	Type: System.Text.Encoding The encoding.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAnyValue](#) [► 1129].

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [► 1129] parameter contains the read value (Value) and the [ErrorCode](#) [► 1120] after execution.

Implements

[IAdsAnyAccess.ReadAnyStringAsync\(UInt32, UInt32, Int32, Encoding, CancellationToken\)](#) [► 827]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [► 67]	



Reference

[AdsConnection Class](#) [► 412]

[ReadAnyStringAsync Overload](#) [► 536]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.24 AdsConnection.ReadAsync Method**Overload List**

	Name	Description
	ReadAsync(UInt32, Memory.Byte., CancellationToken) [► 540]	Reads the value data of the symbol asynchronously into the readBuffer.
	ReadAsync(UInt32, UInt32, Memory.Byte., CancellationToken) [► 541]	Reads the data asynchronously from specified IndexGroup/IndexOffset

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.24.1 AdsConnection.ReadAsync Method (UInt32, Memory`1, Void)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Task<ResultRead> ReadAsync(  
    uint variableHandle,  
    Memory readBuffer,  
    void cancel  
)
```

Parameters

variableHandle Type: [System.UInt32](#)

readBuffer Type: [Memory](#)

cancel Type: [System.Void](#)

Return Value

Type: [Task.ResultRead](#) [► 1143].

Reference

[AdsConnection Class](#) [► 412]

[ReadAsync Overload](#) [► 538]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.24.2 AdsConnection.ReadAsync Method (UInt32, UInt32, Memory`1, Void)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Task<ResultRead> ReadAsync(  
    uint indexGroup,  
    uint indexOffset,  
    Memory readBuffer,  
    void cancel  
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
readBuffer	Type: Memory
cancel	Type: System.Void

Return Value

Type: [Task.ResultRead](#) [▶ 1143].

Reference

[AdsConnection Class](#) [▶ 412]

[ReadAsync Overload](#) [▶ 538]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.24.3 **AdsConnection.ReadAsync Method (UInt32, Memory.Byte., Cancellation.Token)**

Reads the value data of the symbol asynchronously into the readBuffer.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultRead> ReadAsync(
    uint variableHandle,
    Memory<byte> readBuffer,
    Cancellation.Token cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
readBuffer	Type: System.Memory.Byte. The buffer.
cancel	Type: System.Threading.Cancellation.Token The cancellation token.

Return Value

Type: [Task.ResultRead](#) [▶ 1143].

A task that represents the asynchronous read operation. The [ResultRead](#) [▶ 1143] parameter contains the total number of bytes read into the buffer ([ReadBytes](#) [▶ 1145]) and the [ErrorCode](#) [▶ 1120] after execution..

Implements

[IAdsHandle.ReadAsync\(UInt32, Memory.Byte., Cancellation.Token\)](#) [▶ 959]

Reference

[AdsConnection Class](#) [► 412]

[ReadAsync Overload](#) [► 538]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.24.4 AdsConnection.ReadAsync Method (UInt32, UInt32, Memory.Byte., CancellationToken)

Reads the data asynchronously from specified IndexGroup/IndexOffset

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultRead> ReadAsync (
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
readBuffer	Type: System.Memory.Byte. The read buffer, memory area where the data is written.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultRead](#) [► 1143].

A task that represents the asynchronous read operation. The [ResultRead](#) [► 1143] parameter contains the total number of bytes read into the buffer ([ReadBytes](#) [► 1145]) and the [ErrorCode](#) [► 1120] after execution..

Implements

[IAdsReadWrite.ReadAsync\(UInt32, UInt32, Memory.Byte., CancellationToken\)](#) [► 996]

Reference

[AdsConnection Class](#) [► 412]

[ReadAsync Overload](#) [► 538]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.25 AdsConnection.ReadDataType Method

Call this method to obtain information about the specified data type.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataType ReadDataType(
    string typeName
)
```

Parameters

typeName	Type: System.String Name of the data type (without namespace)
----------	--

Return Value

Type: [IDataType](#) [► 2475]

An containing the requested type.

Implements

[IAdsSymbolicAccess.ReadDataType\(String\)](#) [► 1070]

Exceptions

Exception	Condition
ArgumentOutOfRangeException	typeName
ObjectDisposedException	
ClientNotConnectedException [► 67]	

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsSymbolicAccess.TryReadDataType\(String, IDataType.\)](#) [► 1079]

[IAdsSymbolicAccess.ReadDataTypeAsync\(String, CancellationToken\)](#) [► 1070]

6.2.5.2.26 **AdsConnection.ReadDataTypeAsync Method**

read data type as an asynchronous operation.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultValue<IDataType>> ReadDataTypeAsync(
    string typeName,
    CancellationToken cancel
)
```

Parameters

typeName	Type: System.String Name of the data type.
cancel	Type: System.Threading.CancellationToken The cancel token.

Return Value

Type: [Task.ResultValue](#) [[▶ 1181](#)].[IDataType](#) [[▶ 2475](#)].

A task that represents the asynchronous 'ReadDataType' operation. The [ResultValue.TValue](#). [[▶ 1181](#)] parameter contains the read value ([Value](#) [[▶ 1185](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsSymbolicAccess.ReadDataTypeAsync\(String, CancellationToken\)](#) [[▶ 1070](#)]

Exceptions

Exception	Condition
ArgumentOutOfRangeException	typeName
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsSymbolicAccess.ReadDataType\(String\)](#) [[▶ 1070](#)]

[IAdsSymbolicAccess.TryReadDataType\(String, IDataType.\)](#) [[▶ 1079](#)]

6.2.5.2.27 AdsConnection.ReadDeviceInfo Method

Reads the identification and version number of an ADS server.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DeviceInfo ReadDeviceInfo()
```

Return Value

Type: [DeviceInfo](#) [[▶ 801](#)]

DeviceInfo struct containing the name of the device and the version information.

Implements[IAdsConnection.ReadDeviceInfo. \[► 914\]](#)**Reference**[AdsConnection Class \[► 412\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.5.2.28 AdsConnection.ReadDeviceInfoAsync Method**

Reads the identification and version number of an ADS server.

Namespace: [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public Task<ResultDeviceInfo> ReadDeviceInfoAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return ValueType: [Task.ResultDeviceInfo \[► 1132\]](#).

A task that represents the asynchronous 'ReadDeviceState' operation. The [ResultDeviceInfo \[► 1132\]](#) parameter contains the value [DeviceInfo \[► 1134\]](#) and the [ErrorCode \[► 1120\]](#) of the ADS communication after execution.



Implements[IAdsConnection.ReadDeviceInfoAsync\(CancellationToken\) \[► 914\]](#)**Exceptions**

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [► 67]	

Reference[AdsConnection Class \[► 412\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.29 **AdsConnection.ReadState Method**

Overload List

	Name	Description
	ReadState. [▸ 545]	Reads the ADS status and the device status from an ADS server.
	ReadState(Int32) [▸ 546]	Reads the ADS status and the device status from an ADS server.

Reference

[AdsConnection Class \[▸ 412\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.5.2.29.1 **AdsConnection.ReadState Method**

Reads the ADS status and the device status from an ADS server.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public StateInfo ReadState()
```

Return Value

Type: [StateInfo \[▸ 1200\]](#)

The ADS statue and device status.

Implements

[IAdsStateProvider.ReadState. \[▸ 1058\]](#)

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▸ 67]	

Remarks

Not all ADS Servers support the State ADS Request.

Reference

[AdsConnection Class \[▸ 412\]](#)

[ReadState Overload \[▸ 545\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.5.2.29.2 **AdsConnection.ReadState Method (Int32)**

Reads the ADS status and the device status from an ADS server.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public StateInfo ReadState(
    int timeout
)
```

Parameters

timeout	Type: System.Int32 The timeout.
---------	--

Return Value

Type: [StateInfo](#) [[▶ 1200](#)]

The ADS statue and device status.

Implements

[IAdsStateControlTimeout.ReadState\(Int32\)](#) [[▶ 1049](#)]

Exceptions

Exception	Condition
ObjectDisposedException	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[ReadState Overload](#) [[▶ 545](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.30 **AdsConnection.ReadStateAsync Method**

Read the ADS State asynchronously

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultReadDeviceState> ReadStateAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultReadDeviceState \[▸ 1157\]](#).

A task that represents the asynchronous 'ReadState' operation. The [ResultReadDeviceState \[▸ 1157\]](#) parameter contains the value [State \[▸ 1159\]](#) and the [ErrorCode \[▸ 1120\]](#) of the ADS communication after execution.

Implements

[IAdsStateProvider.ReadStateAsync\(CancellationTokens\) \[▸ 1058\]](#)

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▸ 67]	

Remarks

Not all ADS Servers support the State ADS Request

Reference

[AdsConnection Class \[▸ 412\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.5.2.31 AdsConnection.ReadSymbol Method

Call this method to obtain information about the individual symbols (variables) in ADS devices.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IAdsSymbol ReadSymbol(
    string name
)
```

Parameters

name	Type: System.String Name of the symbol.
------	--

Return Value

Type: [IAdsSymbol \[▸ 1755\]](#)

A [IAdsSymbol \[▸ 1755\]](#) containing the requested symbol information or null if symbol could not be found.

Implements

[IAdsSymbolicAccess.ReadSymbol\(String\) \[▸ 1071\]](#)

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.32 AdsConnection.ReadSymbolAsync Method

Call this method to obtain information about the individual symbols (variables) in ADS devices.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultValue<IAdsSymbol>> ReadSymbolAsync (
    string name,
    CancellationToken cancel
)
```

Parameters

name	Type: System.String Name of the symbol.
cancel	Type: System.Threading.CancellationToken The cancel token.

Return Value

Type: [Task.ResultValue](#) [► 1181].[IAdsSymbol](#) [► 1755].

A task that represents the asynchronous 'ReadSymbolInfo' operation. The [ResultValue.TValue](#). [► 1181] parameter contains the read value ([Value](#) [► 1185]) and the [ErrorCode](#) [► 1120] after execution.

Implements

[IAdsSymbolicAccess.ReadSymbolAsync\(String, CancellationToken\)](#) [► 1072]




Reference


[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.33 AdsConnection.ReadValue Method

Overload List

	Name	Description
	ReadValue(IAdsSymbol) [► 549]	Reads the value of a symbol and returns it as an object. Strings and all primitive datatypes (UInt32, Int32, Bool etc.) are supported. Arrays and structures cannot be read.
	ReadValue.T. (IAdsSymbol) [► 550]	Reads the value of a symbol and returns it as an object.
	ReadValue.T.(String) [► 549]	Reads the value of a symbol and returns the value. The parameter type must have the same layout as the ADS symbol.

	Name	Description
	ReadValue(String, Type) [► 551]	Reads the value of a symbol and returns the value as object. The parameter type must have the same layout as the ADS symbol.

Reference

[AdsConnection Class \[► 412\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.33.1 AdsConnection.ReadValue.T. Method (String)

Reads the value of a symbol and returns the value. The parameter type must have the same layout as the ADS symbol.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T ReadValue<T>(
    string name
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
------	--

Type Parameters

T	The value type
---	----------------

Return Value

Type: T
Value of the symbol

Implements

[IAdsSymbolicAccess.ReadValue.T.\(String\) \[► 1073\]](#)

Reference

[AdsConnection Class \[► 412\]](#)

[ReadValue Overload \[► 548\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.33.2 AdsConnection.ReadValue Method (ISymbol)

Reads the value of a symbol and returns it as an object. Strings and all primitive datatypes(UInt32, Int32, Bool etc.) are supported. Arrays and structures cannot be read.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object ReadValue(
    ISymbol symbol
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol that should be read.
--------	--

Return Value

Type: [Object](#)

The value of the symbol as an object.

Implements

[IAdsSymbolicAccess.ReadValue\(ISymbol\)](#) [[▶ 1073](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[ReadValue Overload](#) [[▶ 548](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.33.3 AdsConnection.ReadValue.T. Method (ISymbol)

Reads the value of a symbol and returns it as an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T ReadValue<T>(
    ISymbol symbol
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol that should be read.
--------	--

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: T

The value of the symbol.

Implements

[IAdsSymbolicAccess.ReadValue.T.\(ISymbol\) \[▸ 1074\]](#)

Remarks

Supported types are limited to 'AnyTypes', what includes all primitive types (UInt32, Int32, Bool etc.), strings, and Arrays that are compositions of 'AnyTypes'.

Reference

[AdsConnection Class \[▸ 412\]](#)

[ReadValue Overload \[▸ 548\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.5.2.33.4 AdsConnection.ReadValue Method (String, Type)

Reads the value of a symbol and returns the value as object. The parameter type must have the same layout as the ADS symbol.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object ReadValue(  
    string name,  
    Type type  
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
type	Type: System.Type Managed type of the ADS symbol.

Return Value

Type: [Object](#)
Value of the symbol

Implements

[IAdsSymbolicAccess.ReadValue\(String, Type\) \[▸ 1075\]](#)

Reference





[AdsConnection Class \[▸ 412\]](#)

[ReadValue Overload \[▸ 548\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.5.2.34 AdsConnection.ReadValueAsync Method

Overload List

	Name	Description
	ReadValueAsync(ISymbol, CancellationToken) [▶ 553]	Reads the value of a symbol and returns it as an object. Strings and all primitive datatypes (UInt32, Int32, Bool etc.) are supported. Arrays and structures cannot be read.
	ReadValueAsync.T.(ISymbol, CancellationToken) [▶ 554]	Reads the value of a symbol asynchronously and returns it as an object.
	ReadValueAsync.T.(String, CancellationToken) [▶ 552]	Reads the value of a symbol asynchronously.
	ReadValueAsync(String, Type, CancellationToken) [▶ 555]	Reads the value of a symbol asynchronously.

Reference

[AdsConnection Class](#) [▶ 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.34.1 AdsConnection.ReadValueAsync.T. Method (String, CancellationToken)

Reads the value of a symbol asynchronously.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultValue<T>> ReadValueAsync<T>(
    string name,
    CancellationToken cancel
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
cancel	Type: System.Threading.CancellationToken The cancel token.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [Task.ResultValue](#) [[▶ 1181](#)].T..

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [[▶ 1129](#)] parameter contains the read value ([Value](#) [[▶ 1185](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsSymbolicAccess.ReadValueAsync.T.\(String, CancellationToken\)](#) [[▶ 1076](#)]

Remarks

The parameter type must have the same layout as the ADS symbol.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[ReadValueAsync Overload](#) [[▶ 552](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.34.2 **AdsConnection.ReadValueAsync Method (ISymbol, CancellationToken)**

Reads the value of a symbol and returns it as an object. Strings and all primitive datatypes (UInt32, Int32, Bool etc.) are supported. Arrays and structures cannot be read.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultAnyValue> ReadValueAsync (
    ISymbol symbol,
    CancellationToken cancel
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol that should be read.
cancel	Type: System.Threading.CancellationToken The cancel token.

Return Value

Type: [Task.ResultAnyValue](#) [[▶ 1129](#)].

The value of the symbol as an object.

Implements

[IAdsSymbolicAccess.ReadValueAsync\(ISymbol, CancellationToken\)](#) [[▶ 1077](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[ReadValueAsync Overload](#) [[▶ 552](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.34.3 **AdsConnection.ReadValueAsync.T. Method (ISymbol, Cancellation-Token)**

Reads the value of a symbol asynchronously and returns it as an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultValue<T>> ReadValueAsync<T>(
    ISymbol symbol,
    CancellationToken cancel
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol that should be read.
cancel	Type: System.Threading.CancellationToken The cancel token.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [Task.ResultValue](#) [[▶ 1181](#)].T..

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [[▶ 1129](#)] parameter contains the read value ([Value](#) [[▶ 1185](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsSymbolicAccess.ReadValueAsync.T.\(ISymbol, Cancellation-Token\)](#) [[▶ 1078](#)]

Remarks

Supported types are limited to 'AnyTypes', what includes all primitive types (UInt32, Int32, Bool etc.), strings, and Arrays that are compositions of 'AnyTypes'.

Reference

[AdsConnection Class](#) [► 412]

[ReadValueAsync Overload](#) [► 552]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.34.4 **AdsConnection.ReadValueAsync Method (String, Type, CancellationTok**

Reads the value of a symbol asynchronously.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAnyValue> ReadValueAsync (
    string name,
    Type type,
    CancellationTok
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
type	Type: System.Type Managed type of the ADS symbol.
cancel	Type: System.Threading.CancellationTok The cancel token.

Return Value

Type: [Task.ResultAnyValue](#) [► 1129].

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [► 1129] parameter contains the read value ([Value](#) [► 1185]) and the [ErrorCode](#) [► 1120] after execution.

Implements

[IAdsSymbolicAccess.ReadValueAsync\(String, Type, CancellationTok\)](#) [► 1079]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedExcep tion [► 67]	

Remarks

The parameter type must have the same layout as the ADS symbol.

Reference

[AdsConnection Class](#) [► 412]

[ReadValueAsync Overload](#) [[▶ 552](#)]




[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

 [ResultAnyValue.Value Property](#) [[▶ 1131](#)]

6.2.5.2.35 AdsConnection.ReadWrite Method

Overload List

	Name	Description
	ReadWrite(UInt32, Memory.Byte, ReadOnlyMemory.Byte) [▶ 558]	Writes data synchronously to an ADS device and then Reads data from that target.
	ReadWrite(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte) [▶ 559]	Writes data synchronously to an ADS device and then Reads data from this device into the readBuffer
	ReadWrite(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte, Int32) [▶ 560]	Writes data synchronously to an ADS device and then Reads data from this device.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.35.1 AdsConnection.ReadWrite Method (UInt32, Memory`1, Void)

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public int ReadWrite(
    uint variableHandle,
    Memory readBuffer,
    void writeBuffer
)
```

Parameters

variableHandle Type: [System.UInt32](#)

readBuffer Type: [Memory](#)

writeBuffer Type: [System.Void](#)

Return ValueType: [Int32](#)**Reference**[AdsConnection Class](#) [[▶ 412](#)][ReadWrite Overload](#) [[▶ 556](#)][TwinCAT.Ads Namespace](#) [[▶ 179](#)]**6.2.5.2.35.2 AdsConnection.ReadWrite Method (UInt32, UInt32, Memory`1, Void)****Namespace:** [TwinCAT.Ads](#) [[▶ 179](#)]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14**Syntax****C#**

```
public int ReadWrite(  
    uint indexGroup,  
    uint indexOffset,  
    Memory readBuffer,  
    void writeBuffer  
)
```

ParametersindexGroup Type: [System.UInt32](#)indexOffset Type: [System.UInt32](#)readBuffer Type: [Memory](#)writeBuffer Type: [System.Void](#)**Return Value**Type: [Int32](#)**Reference**[AdsConnection Class](#) [[▶ 412](#)][ReadWrite Overload](#) [[▶ 556](#)][TwinCAT.Ads Namespace](#) [[▶ 179](#)]**6.2.5.2.35.3 AdsConnection.ReadWrite Method (UInt32, UInt32, Memory`1, Void, Byte)****Namespace:** [TwinCAT.Ads](#) [[▶ 179](#)]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public int ReadWrite(
    uint indexGroup,
    uint indexOffset,
    Memory readBuffer,
    void writeBuffer,
    byte timeout
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
readBuffer	Type: Memory
writeBuffer	Type: System.Void
timeout	Type: System.Byte

Return Value

Type: [Int32](#)

Reference

[AdsConnection Class](#) [► 412]

[ReadWrite Overload](#) [► 556]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.35.4 **AdsConnection.ReadWrite Method (UInt32, Memory.Byte., ReadOnlyMemory.Byte.)**

Writes data synchronously to an ADS device and then Reads data from that target.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ReadWrite(
    uint variableHandle,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

variableHandle	Type: System.UInt32 Variable handle.
readBuffer	Type: System.Memory.Byte. The read buffer.

writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.
-------------	---

Return Value

Type: [Int32](#)
Number of successfully returned data bytes.

Implements

[IAdsHandle.ReadWrite\(UInt32, Memory.Byte., ReadOnlyMemory.Byte.\)](#) [[▶ 960](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[ReadWrite Overload](#) [[▶ 556](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[AdsConnection.TryReadWrite\(UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.\)](#) [[▶ 597](#)]

[AdsConnection.ReadWriteAsync\(UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken\)](#) [[▶ 563](#)]

6.2.5.2.35.5 AdsConnection.ReadWrite Method (UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte.)

Writes data synchronously to an ADS device and then Reads data from this device into the readBuffer

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ReadWrite(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

indexGroup	Type: System.UInt32 The index group number of the requested ADS service.
indexOffset	Type: System.UInt32 The index offset number of the requested ADS service.
readBuffer	Type: System.Memory.Byte . The read buffer.

writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.
-------------	---

Return Value

Type: [Int32](#)

Number of successfully returned (read) data bytes.

Implements

[IAdsReadWrite2.ReadWrite\(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte.\)](#) [[▸ 1002](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▸ 67]	

Reference

[AdsConnection Class](#) [[▸ 412](#)]

[ReadWrite Overload](#) [[▸ 556](#)]

[TwinCAT.Ads Namespace](#) [[▸ 179](#)]

6.2.5.2.35.6 **AdsConnection.ReadWrite Method (UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32)**

Writes data synchronously to an ADS device and then Reads data from this device.

Namespace: [TwinCAT.Ads](#) [[▸ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ReadWrite(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer,
    int timeout
)
```

Parameters

indexGroup	Type: System.UInt32 The index group number of the requested ADS service.
indexOffset	Type: System.UInt32 The index offset number of the requested ADS service.
readBuffer	Type: System.Memory.Byte . The read buffer.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.

timeout	Type: System.Int32 The timeout.
---------	--

Return Value

Type: [Int32](#)
Number of successfully returned data bytes.

Implements

[IAdsReadWriteTimeoutAccess.ReadWrite\(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32\)](#)
[\[▶ 1008\]](#)

Reference



[AdsConnection Class \[▶ 412\]](#)

[ReadWrite Overload \[▶ 556\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.5.2.36 AdsConnection.ReadWriteAsync Method

Overload List

	Name	Description
	ReadWriteAsync(UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken) [▶ 563]	ReadWrites value data asynchronously to/from the symbol represented by the variableHandle.
	ReadWriteAsync(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken) [▶ 564]	Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer

Reference

[AdsConnection Class \[▶ 412\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.5.2.36.1 AdsConnection.ReadWriteAsync Method (UInt32, Memory`1, Void, Byte)

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Task<ResultReadWrite> ReadWriteAsync(  
    uint variableHandle,  
    Memory readBuffer,  
    void writeBuffer,  
    byte cancel  
)
```

Parameters

variableHandle	Type: System.UInt32
readBuffer	Type: Memory
writeBuffer	Type: System.Void
cancel	Type: System.Byte

Return Value

Type: [Task.ResultReadWrite](#) [[▶ 1163](#)].

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[ReadWriteAsync Overload](#) [[▶ 561](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.36.2 **AdsConnection.ReadWriteAsync Method (UInt32, UInt32, Memory`1, Void, Byte)**

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Task<ResultReadWrite> ReadWriteAsync(  
    uint indexGroup,  
    uint indexOffset,  
    Memory readBuffer,  
    void writeBuffer,  
    byte cancel  
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
readBuffer	Type: Memory

writeBuffer Type: [System.Void](#)

cancel Type: [System.Byte](#)

Return Value

Type: [Task.ResultReadWrite](#) [[▶ 1163](#)].

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[ReadWriteAsync Overload](#) [[▶ 561](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.36.3 AdsConnection.ReadWriteAsync Method (UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken)

ReadWrites value data asynchronously to/from the symbol represented by the variableHandle.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultReadWrite> ReadWriteAsync(
    uint variableHandle,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 Variable handle.
readBuffer	Type: System.Memory.Byte. The read data / value
writeBuffer	Type: System.ReadOnlyMemory.Byte. The write data / value.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultReadWrite](#) [[▶ 1163](#)].

A task that represents the asynchronous 'ReadWrite' operation. The [ResultReadWrite](#) [[▶ 1163](#)] parameter contains the total number of bytes read into the buffer ([ReadBytes](#) [[▶ 1145](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsHandle.ReadWriteAsync\(UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken\)](#) [[▶ 961](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[ReadWriteAsync Overload](#) [[▶ 561](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.36.4 **AdsConnection.ReadWriteAsync Method (UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken)**

Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultReadWrite> ReadWriteAsync(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
readBuffer	Type: System.Memory.Byte. The read buffer.
writeBuffer	Type: System.ReadOnlyMemory.Byte. The write buffer.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultReadWrite](#) [[▶ 1163](#)].

A task that represents the asynchronous 'ReadWrite' operation. The [ResultReadWrite](#) [[▶ 1163](#)] parameter contains the total number of bytes read into the buffer ([ReadBytes](#) [[▶ 1145](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsReadWrite.ReadWriteAsync\(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken\)](#)
[\[▶ 997\]](#)

Reference

[AdsConnection Class \[▶ 412\]](#)

[ReadWriteAsync Overload \[▶ 561\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.5.2.37 AdsConnection.RegisterAdsStateChangedAsync Method

Registers for [AdsStateChanged \[▶ 655\]](#) events as an asynchronous operation.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultAds> RegisterAdsStateChangedAsync (
    EventHandler<AdsStateChangedEventArgs> handler,
    CancellationToken cancel
)
```

Parameters

handler	Type: System.EventHandler.AdsStateChangedEventArgs [▶ 729] . The handler function to be registered for AdsStateChanged calls.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds \[▶ 1116\]](#).

A task that represents the asynchronous 'RegisterAdsStateChanged' operation. The [ResultAds \[▶ 1116\]](#) parameter contains the state the [ErrorCode \[▶ 1120\]](#) of the ADS communication after execution.

Implements

[IAdsStateProvider.RegisterAdsStateChangedAsync\(EventHandler.AdsStateChangedEventArgs., CancellationToken\)](#) [\[▶ 1059\]](#)

Reference

[AdsConnection Class \[▶ 412\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.5.2.38 AdsConnection.RegisterSymbolVersionChangedAsync Method

Registers the symbol version changed asynchronously.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAds> RegisterSymbolVersionChangedAsync (
    EventHandler<AdsSymbolVersionChangedEventArgs> handler,
    CancellationToken cancel
)
```

Parameters

handler	Type: System.EventHandler.AdsSymbolVersionChangedEventArgs [▶ 739]. The handler function to register.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds](#) [▶ 1116].

A task that represents the asynchronous 'RegisterSymbolVersionChanged' operation. The [ResultAds](#) [▶ 1116] parameter contains the value [ErrorCode](#) [▶ 1120] of the ADS communication after execution.

Implements

[IAdsSymbolChangedProvider.RegisterSymbolVersionChangedAsync\(EventHandler.AdsSymbolVersionChangedEventArgs, CancellationToken\)](#) [▶ 1062]

Exceptions

Exception	Condition
ObjectDisposedException	



Reference

[AdsConnection Class](#) [▶ 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.39 AdsConnection.TryAddDeviceNotification Method

Overload List

	Name	Description
	TryAddDeviceNotification(String, Int32, NotificationSettings, Object, UInt32.) [▶ 567]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 652] event.
	TryAddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.) [▶ 568]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 652] event.

Reference

[AdsConnection Class \[▶ 412\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.5.2.39.1 AdsConnection.TryAddDeviceNotification Method (String, Int32, NotificationSettings, Object, UInt32.)

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotification \[▶ 652\]](#) event.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryAddDeviceNotification(
    string variableName,
    int dataSize,
    NotificationSettings settings,
    Object? userData,
    out uint handle
)
```

Parameters

variableName	Type: System.String Name of the variable.
dataSize	Type: System.Int32 Maximum amount of data in bytes to receive with this ADS Notification.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The notification settings.
userData	Type: System.Object This object can be used to store user specific data.
handle	Type: System.UInt32 . The notification handle.

Return Value

Type: [AdsErrorCode \[▶ 664\]](#)

The ADS ErrorCode.

Implements

[IAdsNotifications.TryAddDeviceNotification\(String, Int32, NotificationSettings, Object, UInt32.\) \[▶ 985\]](#)

Exceptions

Exception	Condition
ObjectDisposedException	

Remarks

The
dataSize

Parameter defines the amount of bytes, that will be attached to the [AdsNotification](#) [[▶ 652](#)] as value. Because notifications allocate TwinCAT system resources, a complementary call to [TryDeleteDeviceNotification\(UInt32\)](#) [[▶ 572](#)] should always be called when the notification is not used anymore.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryAddDeviceNotification Overload](#) [[▶ 566](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[AdsConnection.AdsNotification](#) [[▶ 652](#)]

[AdsConnection.TryDeleteDeviceNotification\(UInt32\)](#) [[▶ 572](#)]

[AddDeviceNotification Overload](#) [[▶ 971](#)]

[AddDeviceNotificationAsync Overload](#) [[▶ 973](#)]

[TryAddDeviceNotification Overload](#) [[▶ 985](#)]

6.2.5.2.39.2 **AdsConnection.TryAddDeviceNotification Method (UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.)**

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotification](#) [[▶ 652](#)] event.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryAddDeviceNotification(
    uint indexGroup,
    uint indexOffset,
    int dataSize,
    NotificationSettings settings,
    Object? userData,
    out uint handle
)
```

Parameters

indexGroup	Type: System.UInt32 The index group number of the requested ADS service.
indexOffset	Type: System.UInt32 The index offset number of the requested ADS service.
dataSize	Type: System.Int32 Maximum amount of data in bytes to receive with this ADS Notification.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
handle	Type: System.UInt32 . The notification handle.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
 The ADS error code.

Implements

[IAdsNotifications.TryAddDeviceNotification\(UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.\)](#) [[▶ 986](#)]

Remarks

The
 dataSize



Parameter defines the amount of bytes, that will be attached to the [AdsNotification](#) [[▶ 652](#)] as value. Because notifications allocate TwinCAT system resources, a complementary call to [TryDeleteDeviceNotification\(UInt32\)](#) [[▶ 990](#)] should always called when the notification is not used anymore.

Reference

- [AdsConnection Class](#) [[▶ 412](#)]
- [TryAddDeviceNotification Overload](#) [[▶ 566](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]
- [AdsConnection.TryDeleteDeviceNotification\(UInt32\)](#) [[▶ 572](#)]
- [AdsConnection.AdsNotification](#) [[▶ 652](#)]
- [AdsConnection.AdsNotificationError](#) [[▶ 654](#)]
- [AddDeviceNotification Overload](#) [[▶ 971](#)]
- [TryAddDeviceNotification Overload](#) [[▶ 985](#)]
- [Overload:TwinCAT.Ads.IAdsNotifications..AddDeviceNotificationAsync] Overload

6.2.5.2.40 AdsConnection.TryAddDeviceNotificationEx Method

Overload List

	Name	Description
	TryAddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32., UInt32.) [▶ 570]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event.
	TryAddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., UInt32.) [▶ 571]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 654] event.

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.40.1 **AdsConnection.TryAddDeviceNotificationEx Method (String, NotificationSettings, Object, Type, .Int32., UInt32.)**

Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryAddDeviceNotificationEx(
    string symbolPath,
    NotificationSettings settings,
    Object? userData,
    Type type,
    int[]? args,
    out uint handle
)
```

Parameters

symbolPath	Type: System.String The symbol path..
settings	Type: TwinCAT.Ads.NotificationSettings [► 1106] The settings.
userData	Type: System.Object This object can be used to store user specific data.
type	Type: System.Type Type of the object stored in the event argument.
args	Type: .System.Int32. Additional arguments.
handle	Type: System.UInt32. The handle.

Return Value

Type: [AdsErrorCode](#) [► 664]

The handle of the notification.

Implements

[IAdsNotifications.TryAddDeviceNotificationEx\(String, NotificationSettings, Object, Type, .Int32., UInt32.\)](#) [► 988]

Exceptions

Exception	Condition
ObjectDisposedException	

Reference

[AdsConnection Class](#) [► 412]

[TryAddDeviceNotificationEx Overload \[▶ 569\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.5.2.40.2 **AdsConnection.TryAddDeviceNotificationEx Method (UInt32, UInt32, NotificationSettings, Object, Type, .Int32., UInt32.)**

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotificationEx \[▶ 654\]](#) event.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryAddDeviceNotificationEx(
    uint indexGroup,
    uint indexOffset,
    NotificationSettings settings,
    Object? userData,
    Type type,
    int[]? args,
    out uint handle
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
type	Type: System.Type Type of the object stored in the event argument ('AnyType')
args	Type: .System.Int32. The 'AnyType' arguments.
handle	Type: System.UInt32. The notification handle.

Return Value

Type: [AdsErrorCode \[▶ 664\]](#)
The ADS Error code.

Implements

[IAdsNotifications.TryAddDeviceNotificationEx\(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., UInt32.\) \[▶ 989\]](#)

Remarks

If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive types (AnyType) are supported by this method. Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\) \[▶ 495\]](#) should always called when the notification is not used anymore.

Reference[AdsConnection Class](#) [► 412][TryAddDeviceNotificationEx Overload](#) [► 569][TwinCAT.Ads Namespace](#) [► 179][AdsConnection.DeleteDeviceNotification\(UInt32\)](#) [► 495][AdsConnection.AdsNotificationEx](#) [► 654][AdsConnection.AdsNotificationError](#) [► 654][AddDeviceNotificationEx Overload](#) [► 976][TryAddDeviceNotificationEx Overload](#) [► 988][AddDeviceNotificationExAsync Overload](#) [► 981]**6.2.5.2.41 AdsConnection.TryCreateVariableHandle Method**

Read (determine) the Symbol handle by its name/path

Namespace: [TwinCAT.Ads](#) [► 179]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public AdsErrorCode TryCreateVariableHandle(
    string symbolName,
    out uint variableHandle
)
```

Parameters

symbolName	Type: System.String SymbolName / Path.
variableHandle	Type: System.UInt32 . The handle.

Return ValueType: [AdsErrorCode](#) [► 664]

AdsErrorCode.

Implements[IAdsHandle.TryCreateVariableHandle\(String, UInt32.\)](#) [► 961]**Reference**[AdsConnection Class](#) [► 412][TwinCAT.Ads Namespace](#) [► 179]**6.2.5.2.42 AdsConnection.TryDeleteDeviceNotification Method**

Deletes a registered notification.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryDeleteDeviceNotification(  
    uint notificationHandle  
)
```

Parameters

notificationHandle	Type: System.UInt32 Notification handle.
--------------------	---

Return Value

Type: [AdsErrorCode](#) [► 664]

The ADS error code.

Implements

[IAdsNotifications.TryDeleteDeviceNotification\(UInt32\)](#) [► 990]

Remarks

This is the complementary method to [TryAddDeviceNotification Overload](#) [► 985] overloads and should be called when the notification is not needed anymore the free TwinCAT realtime resources.

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

[AddDeviceNotification Overload](#) [► 971]

[AdsConnection.AdsNotification](#) [► 652]

[TryAddDeviceNotification Overload](#) [► 985]

[AddDeviceNotificationAsync Overload](#) [► 973]

6.2.5.2.43 AdsConnection.TryDeleteVariableHandle Method

Releases the specified symbol/variable handle synchronously.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryDeleteVariableHandle(  
    uint variableHandle  
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable
----------------	---

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS error code.

Implements

[IAdsHandle.TryDeleteVariableHandle\(UInt32\)](#) [[▶ 962](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Remarks

It is a good practice to release all variable handles after use to regain internal resources in the TwinCAT subsystem. The composite method to this [TryDeleteVariableHandle\(UInt32\)](#) is the [TryCreateVariableHandle\(String, UInt32\)](#) [[▶ 572](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]



[TwinCAT.Ads Namespace](#) [[▶ 179](#)]





[AdsConnection.TryCreateVariableHandle\(String, UInt32\)](#) [[▶ 572](#)]

[AdsConnection.DeleteVariableHandleAsync\(UInt32, CancellationToken\)](#) [[▶ 497](#)]

[AdsConnection.DeleteVariableHandle\(UInt32\)](#) [[▶ 496](#)]

6.2.5.2.44 AdsConnection.TryInvokeRpcMethod Method**Overload List**

	Name	Description
	TryInvokeRpcMethod(String, String, Object, Object) [▶ 575]	Invokes the specified RPC Method
	TryInvokeRpcMethod(String, String, Object, Object, Object) [▶ 577]	Invokes the specified RPC Method

	Name	Description
 	TryInvokeRpcMethod(IRpcCallableInstance, IRpcMethod, Object, AnyTypeSpecifier, AnyTypeSpecifier, Object, Object.) [▶ 580]	Invokes the rpc method.
 	TryInvokeRpcMethod(String, String, Object, AnyTypeSpecifier, AnyTypeSpecifier, Object, Object.) [▶ 578]	Invokes the rpc method.

Reference

[AdsConnection Class](#) [▶ 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.44.1 AdsConnection.TryInvokeRpcMethod Method (String, String, Object, Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryInvokeRpcMethod(
    string symbolPath,
    string methodName,
    Object[]? inParameters,
    out Object?? retValue
)
```

Parameters

symbolPath	Type: System.String The symbol path.
methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
retValue	Type: System.Object . The return value of the RPC method as object.

Return Value

Type: [AdsErrorCode](#) [▶ [664](#)]
The ADS Error Code.

Implements

[IAdsRpcInvoke.TryInvokeRpcMethod\(String, String, Object, Object.\)](#) [▶ [1026](#)]

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [▶ [2094](#)] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] { (short) 3, (short) 4 });

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```


Reference

[AdsConnection Class \[► 412\]](#)

[TryInvokeRpcMethod Overload \[► 574\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.44.2 **AdsConnection.TryInvokeRpcMethod Method (String, String, .Object., .Object., Object.)**

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryInvokeRpcMethod(
    string symbolPath,
    string methodName,
    Object[]? inParameters,
    out Object[]? outParameters,
    out Object?? retValue
)
```

Parameters

symbolPath	Type: System.String The symbol path.
methodName	Type: System.String The method name.
inParameters	Type: .System.Object. The parameters.
outParameters	Type: .System.Object.. The out parameters.
retValue	Type: System.Object. The return value of the RPC method as object.

Return Value

Type: [AdsErrorCode \[► 664\]](#)

The ADS Error Code.

Implements

[IAdsRpcInvoke.TryInvokeRpcMethod\(String, String, .Object., .Object., Object.\) \[► 1028\]](#)

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier \[► 2094\]](#) specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```

class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}

```

Reference

[AdsConnection Class \[► 412\]](#)

[TryInvokeRpcMethod Overload \[► 574\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.44.3 **AdsConnection.TryInvokeRpcMethod Method (String, String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object., Object.)**

Invokes the rpc method.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryInvokeRpcMethod(
    string symbolPath,
    string methodName,
    Object[]? inParameters,
    AnyTypeSpecifier[]? outSpecifiers,
    AnyTypeSpecifier? retSpecifier,
    out Object[]? outParameters,
    out Object?? retValue
)
```

Parameters

symbolPath	Type: System.String The symbol.
methodName	Type: System.String Name of the method.
inParameters	Type: .System.Object . The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094]. The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094] The ret specifier (specifying the return value) or NULL.
outParameters	Type: .System.Object . The out parameters.
retValue	Type: System.Object . The return value of the RPC method./>

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
AdsErrorCode.

Implements

[IAdsRpcInvoke.TryInvokeRpcMethod\(String, String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object.. Object.\)](#) [[▶ 1030](#)]

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
    }
}
```

```

AmsAddress address = ArgParser.Parse(args);

using (AdsClient client = new AdsClient())
{
    //client.Synchronize = false;

    // Connect to the target device
    client.Connect(address);

    SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
    ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

    // Get the Symbols (Dynamic Symbols)

    IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
the PLC Program

    // Call a Method that has the following signature (within MAIN Program)
    /* {attribute 'TcRpcEnable'}
    METHOD PUBLIC M_Add : INT
    VAR_INPUT
        i1 : INT := 0;
        i2 : INT := 0;
    END_VAR
    */

    short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

    // Call a Method that has no parameter and returns VOID
    main.InvokeRpcMethod("M_Method1", new object[] {});

    //Browsing RpcMethods
    foreach(IRpcMethod method in main.RpcMethods)
    {
        string methodName = method.Name;

        foreach(IRpcMethodParameter parameter in method.Parameters)
        {
            string parameterName = parameter.Name;
            string parameterType = parameter.TypeName;
        }
    }
}
}

```

Reference

[AdsConnection Class \[► 412\]](#)

[TryInvokeRpcMethod Overload \[► 574\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.44.4 AdsConnection.TryInvokeRpcMethod Method (IRpcCallableInstance, IRpcMethod, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object., Object.)

Invokes the rpc method.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public AdsErrorCode TryInvokeRpcMethod(
    IRpcCallableInstance symbol,
    IRpcMethod method,
    Object[]? inParameters,
    AnyTypeSpecifier[]? outSpecifiers,
    AnyTypeSpecifier? retSpecifier,

```

```

    out Object[]? outParameters,
    out Object?? retVal
)

```

Parameters

symbol	Type: TwinCAT.TypeSystem.IRpcCallableInstance [▶ 2606] The RPC callable symbol
method	Type: TwinCAT.TypeSystem.IRpcMethod [▶ 2625] The method.
inParameters	Type: .System.Object. The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094]. The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094] The ret specifier (specifying the return value) or NULL.
outParameters	Type: .System.Object.. The out parameters.
retValue	Type: System.Object. The return value of the RPC method./>

Return Value

Type: [AdsErrorCode](#) [▶ 664]
AdsErrorCode.

Implements

[IRpcCallableInstance.TryInvokeRpcMethod\(IRpcCallableInstance, IRpcMethod, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object., Object.\)](#) [▶ 1031]

Exceptions

Exception	Condition
ObjectDisposedException	this.Name
ClientNotConnectedException [▶ 67]	

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outParameters, are allowed to be empty or NULL.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```

class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);
    }
}

```

```

using (AdsClient client = new AdsClient())
{
    //client.Synchronize = false;

    // Connect to the target device
    client.Connect(address);

    SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
    ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

    // Get the Symbols (Dynamic Symbols)

    IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
the PLC Program

    // Call a Method that has the following signature (within MAIN Program)
    /* {attribute 'TcRpcEnable'}
    METHOD PUBLIC M_Add : INT
    VAR_INPUT
        i1 : INT := 0;
        i2 : INT := 0;
    END_VAR
    */

    short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

    // Call a Method that has no parameter and returns VOID
    main.InvokeRpcMethod("M_Method1", new object[] {});

    //Browsing RpcMethods
    foreach(IRpcMethod method in main.RpcMethods)
    {
        string methodName = method.Name;

        foreach(IRpcMethodParameter parameter in method.Parameters)
        {
            string parameterName = parameter.Name;
            string parameterType = parameter.TypeName;
        }
    }
}

```

Reference




[AdsConnection Class](#) [► 412]

[TryInvokeRpcMethod Overload](#) [► 574]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.45 AdsConnection.TryRead Method

Overload List

	Name	Description
	TryRead(UInt32, Memory.Byte., Int32.) [► 585]	Reads the value synchronously data of the symbol, that is represented by the variable handle into the readBuffer.
	TryRead(UInt32, UInt32, Memory.Byte., Int32.) [► 586]	Reads data synchronously from an ADS device and writes it to the given stream.
	TryRead(UInt32, UInt32, Memory.Byte., Int32, Int32.) [► 586]	Reads data synchronously from an ADS device and writes it to the given stream.

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.45.1 AdsConnection.TryRead Method (UInt32, Memory`1, Void)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public AdsErrorCode TryRead(  
    uint variableHandle,  
    Memory readBuffer,  
    void readBytes  
)
```

Parameters

variableHandle Type: [System.UInt32](#)

readBuffer Type: [Memory](#)

readBytes Type: [System.Void](#)

Return Value

Type: [AdsErrorCode](#) [► 664]

Reference

[AdsConnection Class](#) [► 412]

[TryRead Overload](#) [► 582]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.45.2 AdsConnection.TryRead Method (UInt32, UInt32, Memory`1, Void)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public AdsErrorCode TryRead(  
    uint indexGroup,  
    uint indexOffset,  
    Memory readBuffer,  
    void readBytes  
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
readBuffer	Type: Memory
readBytes	Type: System.Void

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryRead Overload](#) [[▶ 582](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.45.3 **AdsConnection.TryRead Method (UInt32, UInt32, Memory`1, Void, Byte)**

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax**C#**

```
public AdsErrorCode TryRead(  
    uint indexGroup,  
    uint indexOffset,  
    Memory readBuffer,  
    void timeout,  
    byte readBytes  
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
readBuffer	Type: Memory
timeout	Type: System.Void
readBytes	Type: System.Byte

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

Reference[AdsConnection Class](#) [► 412][TryRead Overload](#) [► 582][TwinCAT.Ads Namespace](#) [► 179]**6.2.5.2.45.4 AdsConnection.TryRead Method (UInt32, Memory.Byte., Int32.)**

Reads the value synchronously data of the symbol, that is represented by the variable handle into the readBuffer.

Namespace: [TwinCAT.Ads](#) [► 179]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public AdsErrorCode TryRead(
    uint variableHandle,
    Memory<byte> readBuffer,
    out int readBytes
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
readBuffer	Type: System.Memory.Byte. The read buffer/data
readBytes	Type: System.Int32 . Number of read bytes.

Return ValueType: [AdsErrorCode](#) [► 664]

The ADS error code.

Implements[IAdsHandle.TryRead\(UInt32, Memory.Byte., Int32.\)](#) [► 963]**Exceptions**

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [► 67]	

Reference[AdsConnection Class](#) [► 412][TryRead Overload](#) [► 582][TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.45.5 **AdsConnection.TryRead Method (UInt32, UInt32, Memory.Byte., Int32.)**

Reads data synchronously from an ADS device and writes it to the given stream.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryRead(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    out int readBytes
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
readBuffer	Type: System.Memory.Byte. The read buffer.
readBytes	Type: System.Int32. The number of read bytes.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The ADS error code.

Implements

[IAdsReadWrite.TryRead\(UInt32, UInt32, Memory.Byte., Int32.\)](#) [[▶ 998](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryRead Overload](#) [[▶ 582](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.45.6 **AdsConnection.TryRead Method (UInt32, UInt32, Memory.Byte., Int32, Int32.)**

Reads data synchronously from an ADS device and writes it to the given stream.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryRead(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    int timeout,
    out int readBytes
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
readBuffer	Type: System.Memory.Byte . The read buffer.
timeout	Type: System.Int32 The timeout.
readBytes	Type: System.Int32 . The read bytes.

Return Value

Type: [AdsErrorCode](#) [▶ 664]

The ADS Error code.

Implements

[IAdsReadWriteTimeoutAccess.TryRead\(UInt32, UInt32, Memory.Byte., Int32, Int32.\)](#) [▶ 1008]

Reference

[AdsConnection Class](#) [▶ 412]

[TryRead Overload](#) [▶ 582]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.46 **AdsConnection.TryReadDataType Method**

Call this method to obtain information about the specified data type.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryReadDataType(
    string typeName,
    out IDataTypeInfo? dataType
)
```

Parameters

typeName	Type: System.String Name of the symbol.
dataType	Type: TwinCAT.TypeSystem.IDataType [▶ 2475]. The symbol.

Return Value

Type: [AdsErrorCode](#) [▶ 664]

A [IDataType](#) [▶ 2475] containing the requested symbol information or null if symbol could not be found.

Implements

[IAdsSymbolicAccess.TryReadDataType\(String, IDataType.\)](#) [▶ 1079]

Exceptions

Exception	Condition
ArgumentOutOfRangeException	typeName
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	



Reference

[AdsConnection Class](#) [▶ 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

[IAdsSymbolicAccess.ReadDataType\(String\)](#) [▶ 1070]

6.2.5.2.47 AdsConnection.TryReadState Method**Overload List**

	Name	Description
	TryReadState(StateInfo.) [▶ 589]	Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful.
	TryReadState(Int32, StateInfo.) [▶ 589]	Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful.

Reference

[AdsConnection Class](#) [▶ 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.47.1 **AdsConnection.TryReadState Method (StateInfo.)**

Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryReadState(  
    out StateInfo stateInfo  
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200]. The ADS statue and device status.
-----------	--

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

AdsErrorCode of the ads read state call. Check for AdsErrorCode.NoError to see if call was successful.

Implements

[IAdsStateProvider.TryReadState\(StateInfo.\)](#) [[▶ 1059](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryReadState Overload](#) [[▶ 588](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.47.2 **AdsConnection.TryReadState Method (Int32, StateInfo.)**

Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryReadState(  
    int timeout,  
    out StateInfo stateInfo  
)
```

Parameters

timeout	Type: System.Int32 The timeout.
---------	--

stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200]. The ADS statue and device status.
-----------	--

Return Value

Type: [AdsErrorCode](#) [▶ 664]

AdsErrorCode of the ads read state call. Check for AdsErrorCode.NoError to see if call was successful.

Implements

[IAdsStateControlTimeout.TryReadState\(Int32, StateInfo.\)](#) [▶ 1050]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsConnection Class](#) [▶ 412]

[TryReadState Overload](#) [▶ 588]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.48 AdsConnection.TryReadSymbol Method

Call this method to obtain information about the individual symbols (variables) in ADS devices.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryReadSymbol(
    string symbolPath,
    out IAdsSymbol?? symbol
)
```

Parameters

symbolPath	Type: System.String Name of the symbol.
symbol	Type: TwinCAT.Ads.TypeSystem.IAdsSymbol [▶ 1755]. The symbol.

Return Value

Type: [AdsErrorCode](#) [▶ 664]

A [IAdsSymbol](#) [▶ 1755] containing the requested symbol information or null if symbol could not be found.

Implements

[IAdsSymbolicAccess.TryReadSymbol\(String, IAdsSymbol.\)](#) [[▶ 1080](#)]





Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.49 AdsConnection.TryReadValue Method

Overload List

	Name	Description
	TryReadValue(ISymbol, Object.) [▶ 592]	Reads the value of a symbol and returns the value as object. The parameter type must have the same layout as the ADS symbol.
	TryReadValue.T.(ISymbol, T.) [▶ 593]	Reads the value of a symbol and returns it as an object.
	TryReadValue.T.(String, T.) [▶ 591]	Reads the value of a symbol and returns the value as object.
	TryReadValue(String, Type, Object.) [▶ 594]	Reads the value of a symbol and returns the value as object.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.49.1 AdsConnection.TryReadValue.T. Method (String, T.)

Reads the value of a symbol and returns the value as object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryReadValue<T>(
    string name,
    out T value
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
value	Type: T. The read value of the Symbol.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The [AdsErrorCode](#) [[▶ 664](#)].

Implements

[IAdsSymbolicAccess.TryReadValue.T.\(String, T.\)](#) [[▶ 1081](#)]

Remarks

The parameter type must have the same layout as the ADS symbol.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryReadValue Overload](#) [[▶ 591](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.49.2 AdsConnection.TryReadValue Method (ISymbol, Object.)

Reads the value of a symbol and returns the value as object. The parameter type must have the same layout as the ADS symbol.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public AdsErrorCode TryReadValue(
    ISymbol symbol,
    out Object?? value
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol that should be read.
value	Type: System.Object . The value.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

Value of the symbol

Implements

[IAdsSymbolicAccess.TryReadValue\(ISymbol, Object.\)](#) [[▶ 1082](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Remarks

Supported types are limited to 'AnyTypes', what includes all primitive types (UInt32, Int32, Bool etc.), strings, and Arrays that are compositions of 'AnyTypes'. Structs are not supported.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryReadValue Overload](#) [[▶ 591](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.49.3 AdsConnection.TryReadValue.T. Method (ISymbol, T.)

Reads the value of a symbol and returns it as an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryReadValue<T>(
    ISymbol symbol,
    out T value
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol that should be read.
value	Type: T. The value.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The ADS Error Code

Implements

[IAdsSymbolicAccess.TryReadValue.T.\(ISymbol, T.\)](#) [[▶ 1083](#)]

Remarks

Supported types are limited to 'AnyTypes', what includes all primitive types (UInt32, Int32, Bool etc.), strings, and Arrays that are compositions of 'AnyTypes'.

Reference

[AdsConnection Class](#) [► 412]

[TryReadValue Overload](#) [► 591]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.49.4 AdsConnection.TryReadValue Method (String, Type, Object.)

Reads the value of a symbol and returns the value as object.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public AdsErrorCode TryReadValue(
    string name,
    Type type,
    out Object?? value
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
type	Type: System.Type Managed type of the ADS symbol.
value	Type: System.Object . The read value of the Symbol.

Return Value

Type: [AdsErrorCode](#) [► 664]

The [AdsErrorCode](#) [► 664].

Implements

[IAdsSymbolicAccess.TryReadValue\(String, Type, Object.\)](#) [► 1083]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [► 67]	

Remarks

The parameter type must have the same layout as the ADS symbol.

Reference




[AdsConnection Class](#) [▶ 412]

[TryReadValue Overload](#) [▶ 591]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.50 AdsConnection.TryReadWrite Method

Overload List

	Name	Description
	TryReadWrite(UInt32, Memory.Byte, ReadOnlyMemory.Byte, Int32.) [▶ 597]	ReadWrites value data synchronously to/from the symbol represented by the variableHandle.
	TryReadWrite(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte, Int32.) [▶ 598]	Writes data synchronously to an ADS device and reads data from that device.
	TryReadWrite(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte, Int32, Int32.) [▶ 599]	Writes data synchronously to an ADS device and then Reads data from this device.

Reference

[AdsConnection Class](#) [▶ 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.50.1 AdsConnection.TryReadWrite Method (UInt32, Memory`1, Void, Byte)

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public AdsErrorCode TryReadWrite(
    uint variableHandle,
    Memory readBuffer,
    void writeBuffer,
    byte readBytes
)
```

Parameters

variableHandle	Type: System.UInt32
readBuffer	Type: Memory
writeBuffer	Type: System.Void
readBytes	Type: System.Byte

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryReadWrite Overload](#) [[▶ 595](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.50.2 **AdsConnection.TryReadWrite Method (UInt32, UInt32, Memory`1, Void, Byte)**

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax**C#**

```
public AdsErrorCode TryReadWrite(  
    uint indexGroup,  
    uint indexOffset,  
    Memory readBuffer,  
    void writeBuffer,  
    byte readBytes  
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
readBuffer	Type: Memory
writeBuffer	Type: System.Void
readBytes	Type: System.Byte

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

Reference

[AdsConnection Class](#) [► 412]

[TryReadWrite Overload](#) [► 595]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.50.3 **AdsConnection.TryReadWrite Method (UInt32, UInt32, Memory`1, Void, Byte, ReadOnlyMemory`1)**

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public AdsErrorCode TryReadWrite(  
    uint indexGroup,  
    uint indexOffset,  
    Memory readBuffer,  
    void writeBuffer,  
    byte timeout,  
    ReadOnlyMemory readBytes  
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
readBuffer	Type: Memory
writeBuffer	Type: System.Void
timeout	Type: System.Byte
readBytes	Type: ReadOnlyMemory

Return Value

Type: [AdsErrorCode](#) [► 664]

Reference

[AdsConnection Class](#) [► 412]

[TryReadWrite Overload](#) [► 595]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.50.4 **AdsConnection.TryReadWrite Method (UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.)**

ReadWrites value data synchronously to/from the symbol represented by the variableHandle.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryReadWrite(
    uint variableHandle,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer,
    out int readBytes
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
readBuffer	Type: System.Memory.Byte . The read buffer / read data.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer / write data.
readBytes	Type: System.Int32 . Number of read bytes.

Return Value

Type: [AdsErrorCode](#) [► 664]

The ADS error code.

Implements

[IAdsHandle.TryReadWrite\(UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.\)](#) [► 963]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [► 67]	

Reference

[AdsConnection Class](#) [► 412]

[TryReadWrite Overload](#) [► 595]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.50.5 **AdsConnection.TryReadWrite Method (UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.)**

Writes data synchronously to an ADS device and reads data from that device.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryReadWrite(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer,
    out int readBytes
)
```

Parameters

indexGroup	Type: System.UInt32 The index group number of the requested ADS service.
indexOffset	Type: System.UInt32 The index offset number of the requested ADS service.
readBuffer	Type: System.Memory.Byte . The read buffer.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.
readBytes	Type: System.Int32 . The read bytes.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS Error code.

Implements

[IAdsReadWrite.TryReadWrite\(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.\)](#) [[▶ 998](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryReadWrite Overload](#) [[▶ 595](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.50.6 [AdsConnection.TryReadWrite Method \(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32, Int32.\)](#)

Writes data synchronously to an ADS device and then Reads data from this device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryReadWrite(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer,
    int timeout,
    out int readBytes
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
readBuffer	Type: System.Memory.Byte . The read stream.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write stream.
timeout	Type: System.Int32 The timeout.
readBytes	Type: System.Int32 . The read bytes.

Return Value

Type: [AdsErrorCode](#) [► 664]
The ADS Error code.

Implements

[IAdsReadWriteTimeoutAccess.TryReadWrite\(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32, Int32.\)](#) [► 1009]

Reference



[AdsConnection Class](#) [► 412]


[TryReadWrite Overload](#) [► 595]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.51 AdsConnection.TryWrite Method

Overload List

	Name	Description
	TryWrite(UInt32, ReadOnlyMemory.Byte.) [► 603]	Writes the value data synchronously that is represented in the writeBuffer to the symbol with the specified variableHandle.
	TryWrite(UInt32, UInt32, ReadOnlyMemory.Byte.) [► 604]	Writes data synchronously to an ADS device.

	Name	Description
	TryWrite(UInt32, UInt32, ReadOnlyMemory<Byte>, Int32) [▶ 604]	Writes data synchronously to an ADS device.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.51.1 **AdsConnection.TryWrite Method (UInt32, ReadOnlyMemory`1)**

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public AdsErrorCode TryWrite(
    uint variableHandle,
    ReadOnlyMemory writeBuffer
)
```

Parameters

variableHandle Type: [System.UInt32](#)

writeBuffer Type: [ReadOnlyMemory](#)

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryWrite Overload](#) [[▶ 600](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.51.2 **AdsConnection.TryWrite Method (UInt32, UInt32, ReadOnlyMemory`1)**

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public AdsErrorCode TryWrite(  
    uint indexGroup,  
    uint indexOffset,  
    ReadOnlyMemory writeBuffer  
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
writeBuffer	Type: ReadOnlyMemory

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryWrite Overload](#) [[▶ 600](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.51.3 **AdsConnection.TryWrite Method (UInt32, UInt32, ReadOnlyMemory`1, Void)**

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public AdsErrorCode TryWrite(  
    uint indexGroup,  
    uint indexOffset,  
    ReadOnlyMemory writeBuffer,  
    void timeout  
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
writeBuffer	Type: ReadOnlyMemory
timeout	Type: System.Void

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryWrite Overload](#) [[▶ 600](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.51.4 **AdsConnection.TryWrite Method (UInt32, ReadOnlyMemory.Byte.)**

Writes the value data synchronously that is represented in the writeBuffer to the symbol with the specified variableHandle.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: [TwinCAT.Ads](#) (in [TwinCAT.Ads.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWrite(
    uint variableHandle,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
writeBuffer	Type: System.ReadOnlyMemory.Byte. The write buffer / value.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS error code.

Implements

[IAdsHandle.TryWrite\(UInt32, ReadOnlyMemory.Byte.\)](#) [[▶ 964](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryWrite Overload](#) [[▶ 600](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.51.5 **AdsConnection.TryWrite Method (UInt32, UInt32, ReadOnlyMemory.Byte.)**

Writes data synchronously to an ADS device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWrite(
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
writeBuffer	Type: System.ReadOnlyMemory.Byte. The data buffer to be written.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The ADS error code.

Implements

[IAdsReadWrite.TryWrite\(UInt32, UInt32, ReadOnlyMemory.Byte.\)](#) [[▶ 999](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryWrite Overload](#) [[▶ 600](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.51.6 **AdsConnection.TryWrite Method (UInt32, UInt32, ReadOnlyMemory.Byte., Int32)**

Writes data synchronously to an ADS device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWrite(
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory<byte> writeBuffer,
    int timeout
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.
timeout	Type: System.Int32 The timeout.

Return Value

Type: [AdsErrorCode](#) [▶ 664]
The ADS Error code.

Implements

[IAdsReadWriteTimeoutAccess.TryWrite\(UInt32, UInt32, ReadOnlyMemory.Byte., Int32\)](#) [▶ 1010]

Reference




[AdsConnection Class](#) [▶ 412]


[TryWrite Overload](#) [▶ 600]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.52 AdsConnection.TryWriteControl Method

Overload List

	Name	Description
	TryWriteControl(StationInfo) [▶ 606]	Changes the ADS status and the device status of an ADS server.
	TryWriteControl(StationInfo, Int32) [▶ 606]	Changes the ADS status and the device status of an ADS server.
	TryWriteControl(StationInfo, ReadOnlyMemory.Byte.) [▶ 608]	Changes the ADS status and the device status of an ADS server.

	Name	Description
	TryWriteControl(StateInfo, ReadOnlyMemory<Byte>, Int32) [▶ 609]	Changes the ADS status and the device status of an ADS server.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.52.1 AdsConnection.TryWriteControl Method (StateInfo)

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWriteControl(
    StateInfo stateInfo
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200] New ADS status and device status.
-----------	---

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

AdsErrorCode.

Implements

[IAdsStateControl.TryWriteControl\(StateInfo\)](#) [[▶ 1042](#)]

Exceptions

Exception	Condition
ObjectDisposedException	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryWriteControl Overload](#) [[▶ 605](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.52.2 AdsConnection.TryWriteControl Method (StateInfo, Int32)

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWriteControl(
    StateInfo stateInfo,
    int timeout
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200] New ADS status and device status.
timeout	Type: System.Int32 The timeout.

Return Value

Type: [AdsErrorCode](#) [▶ 664]
AdsErrorCode.

Implements

[IAdsStateControlTimeout.TryWriteControl\(StateInfo, Int32\)](#) [▶ 1051]

Exceptions

Exception	Condition
ObjectDisposedException	

Reference

[AdsConnection Class](#) [▶ 412]

[TryWriteControl Overload](#) [▶ 605]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.52.3 [AdsConnection.TryWriteControl Method \(StateInfo, ReadOnlyMemory`1\)](#)

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public AdsErrorCode TryWriteControl(
    StateInfo stateInfo,
    ReadOnlyMemory writeBuffer
)
```

Parameters

stateInfo Type: [TwinCAT.Ads.StateInfo](#) [▶ 1200]

writeBuffer Type: [ReadOnlyMemory](#)

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryWriteControl Overload](#) [[▶ 605](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.52.4 **AdsConnection.TryWriteControl Method (StateInfo, ReadOnlyMemory`1, Void)**

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934 addedca3e72bc0ea15da1c14

Syntax

C#

```
public AdsErrorCode TryWriteControl(  
    StateInfo stateInfo,  
    ReadOnlyMemory writeBuffer,  
    void timeout  
)
```

Parameters

stateInfo Type: [TwinCAT.Ads.StateInfo](#) [[▶ 1200](#)]

writeBuffer Type: [ReadOnlyMemory](#)

timeout Type: [System.Void](#)

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryWriteControl Overload](#) [[▶ 605](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.52.5 **AdsConnection.TryWriteControl Method (StateInfo, ReadOnlyMemory.Byte.)**

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWriteControl(
    StateInfo stateInfo,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200] New ADS status and device status.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.

Return Value

Type: [AdsErrorCode](#) [▶ 664]
AdsErrorCode.

Implements

[IAdsStateControl.TryWriteControl\(StateInfo, ReadOnlyMemory.Byte.\)](#) [▶ 1043]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsConnection Class](#) [▶ 412]

[TryWriteControl Overload](#) [▶ 605]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.52.6 AdsConnection.TryWriteControl Method (StateInfo, ReadOnlyMemory.Byte., Int32)

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWriteControl(
    StateInfo stateInfo,
    ReadOnlyMemory<byte> writeBuffer,
    int timeout
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [► 1200] New ADS status and device status.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.
timeout	Type: System.Int32 The timeout.

Return Value

Type: [AdsErrorCode](#) [► 664]
AdsErrorCode.

Implements

[IAdsStateControlTimeout.TryWriteControl\(StateInfo, ReadOnlyMemory.Byte., Int32\)](#) [► 1052]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [► 67]	





Reference

[AdsConnection Class](#) [► 412]

[TryWriteControl Overload](#) [► 605]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.53 AdsConnection.TryWriteValue Method**Overload List**

	Name	Description
	TryWriteValue(ISymbol, Object) [► 612]	Writes a value to the symbol. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	TryWriteValue.T.(ISymbol, T) [► 613]	Writes a value to the symbol. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	TryWriteValue(String, Object) [► 611]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	TryWriteValue.T.(String, T) [► 611]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.5.2.53.1 **AdsConnection.TryWriteValue Method (String, Object)**

Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWriteValue (
    string name,
    Object value
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
value	Type: System.Object Object holding the value to be written to the ADS symbol

Return Value

Type: [AdsErrorCode \[▶ 664\]](#)

AdsErrorCode.

Implements

[IAdsSymbolicAccess.TryWriteValue\(String, Object\) \[▶ 1085\]](#)

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsConnection Class \[▶ 412\]](#)

[TryWriteValue Overload \[▶ 610\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.5.2.53.2 **AdsConnection.TryWriteValue.T. Method (String, T)**

Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWriteValue<T>(
    string name,
    T value
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
value	Type: T Object holding the value to be written to the ADS symbol

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
AdsErrorCode.

Implements

[IAdsSymbolicAccess.TryWriteValue.T\(String, T\)](#) [[▶ 1085](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryWriteValue Overload](#) [[▶ 610](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.53.3 AdsConnection.TryWriteValue Method (ISymbol, Object)

Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWriteValue(
    ISymbol symbol,
    Object val
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol the value is written to.
val	Type: System.Object The value to write.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
 AdsErrorCode.

Implements

[IAdsSymbolicAccess.TryWriteValue\(ISymbol, Object\)](#) [[▶ 1086](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryWriteValue Overload](#) [[▶ 610](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.53.4 AdsConnection.TryWriteValue.T. Method (ISymbol, T)

Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWriteValue<T>(
    ISymbol symbol,
    T val
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol the value is written to.
val	Type: T The value to write.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
 AdsErrorCode.

Implements

[IAdsSymbolicAccess.TryWriteValue.T.\(ISymbol, T\)](#) [[▶ 1087](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TryWriteValue Overload](#) [[▶ 610](#)]

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.54 **AdsConnection.UnregisterAdsStateChangedAsync Method**

Registers for [AdsStateChanged \[► 655\]](#) events as an asynchronous operation.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAds> UnregisterAdsStateChangedAsync (
    EventHandler<AdsStateChangedEventArgs> handler,
    CancellationToken cancel
)
```

Parameters

handler	Type: System.EventHandler.AdsStateChangedEventArgs [► 729] . The handler function to be unregistered.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds \[► 1116\]](#).

A task that represents the asynchronous 'UnregisterAdsStateChanged' operation. The [ResultAds \[► 1116\]](#) parameter contains the state the [ErrorCode \[► 1120\]](#) of the ADS communication after execution.

Implements

[IAdsStateProvider.UnregisterAdsStateChangedAsync\(EventHandler.AdsStateChangedEventArgs., CancellationToken\) \[► 1060\]](#)

Reference

[AdsConnection Class \[► 412\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.55 **AdsConnection.UnregisterSymbolVersionChangedAsync Method**

Unregisters the symbol version changed asynchronous.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAds> UnregisterSymbolVersionChangedAsync (
    EventHandler<AdsSymbolVersionChangedEventArgs> handler,
    CancellationToken cancel
)
```

Parameters

handler	Type: System.EventHandler.AdsSymbolVersionChangedEventArgs [▶ 739]. The handler function to unregister.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds](#) [▶ [1116](#)].

A task that represents the asynchronous 'UnregisterSymbolVersionChangedAsync' operation. The [ResultAds](#) [▶ [1116](#)] parameter contains the value [ErrorCode](#) [▶ [1120](#)] of the ADS communication after execution.

Implements

[IAdsSymbolChangedProvider.UnregisterSymbolVersionChangedAsync\(EventHandler.AdsSymbolVersionChangedEventArgs, CancellationToken\)](#) [▶ [1063](#)]

Exceptions

Exception	Condition
ObjectDisposedException	






Reference

[AdsConnection Class](#) [▶ [412](#)]

[TwinCAT.Ads Namespace](#) [▶ [179](#)]

6.2.5.2.56 AdsConnection.Write Method

Overload List

	Name	Description
	Write(UInt32, ReadOnlyMemory.BYTE) [▶ 619]	Writes data synchronously to an ADS device.
	Write(UInt32, UInt32) [▶ 616]	Trigger Client Method/Command.
	Write(UInt32, UInt32, Int32) [▶ 617]	Trigger Client Method/Command.
	Write(UInt32, UInt32, ReadOnlyMemory.BYTE) [▶ 619]	Writes data synchronously to an ADS device.
	Write(UInt32, UInt32, ReadOnlyMemory.BYTE, Int32) [▶ 620]	Writes data synchronously to an ADS device.

Reference

[AdsConnection Class](#) [▶ [412](#)]

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.56.1 AdsConnection.Write Method (UInt32, ReadOnlyMemory`1)

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public void Write(
    uint variableHandle,
    ReadOnlyMemory writeBuffer
)
```

Parameters

variableHandle Type: [System.UInt32](#)

writeBuffer Type: [ReadOnlyMemory](#)

Reference

[AdsConnection Class \[► 412\]](#)

[Write Overload \[► 615\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.56.2 AdsConnection.Write Method (UInt32, UInt32)

Trigger Client Method/Command.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Write(
    uint indexGroup,
    uint indexOffset
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.

Implements

[IAdsReadWrite2.Write\(UInt32, UInt32\) \[► 1003\]](#)

Remarks

This method is used to trigger Client Methods/Commands without parameters.

Reference

[AdsConnection Class \[► 412\]](#)

[Write Overload \[► 615\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.56.3 AdsConnection.Write Method (UInt32, UInt32, Int32)

Trigger Client Method/Command.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public void Write(
    uint indexGroup,
    uint indexOffset,
    int timeout
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
timeout	Type: System.Int32 The timeout.

Remarks

This method is used to trigger Client Methods/Commands without parameters.

Reference

[AdsConnection Class \[► 412\]](#)

[Write Overload \[► 615\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.56.4 AdsConnection.Write Method (UInt32, UInt32, ReadOnlyMemory`1)

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public void Write(  
    uint indexGroup,  
    uint indexOffset,  
    ReadOnlyMemory writeBuffer  
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
writeBuffer	Type: ReadOnlyMemory

Reference

[AdsConnection Class](#) [► 412]

[Write Overload](#) [► 615]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.56.5 AdsConnection.Write Method (UInt32, UInt32, ReadOnlyMemory`1, Void)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public void Write(  
    uint indexGroup,  
    uint indexOffset,  
    ReadOnlyMemory writeBuffer,  
    void timeout  
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
writeBuffer	Type: ReadOnlyMemory
timeout	Type: System.Void

Reference

[AdsConnection Class](#) [► 412]

[Write Overload](#) [► 615]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.56.6 **AdsConnection.Write Method (UInt32, ReadOnlyMemory.Byte.)**

Writes data synchronously to an ADS device.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Write(
    uint variableHandle,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable
writeBuffer	Type: System.ReadOnlyMemory.Byte. The write buffer / value to be written

Implements

[IAdsHandle.Write\(UInt32, ReadOnlyMemory.Byte.\)](#) [▶ 964]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsConnection Class](#) [▶ 412]

[Write Overload](#) [▶ 615]

[TwinCAT.Ads Namespace](#) [▶ 179]

[IAdsHandle.TryWrite\(UInt32, ReadOnlyMemory.Byte.\)](#) [▶ 964]

[IAdsHandle.WriteAsync\(UInt32, ReadOnlyMemory.Byte., CancellationToken\)](#) [▶ 965]

6.2.5.2.56.7 **AdsConnection.Write Method (UInt32, UInt32, ReadOnlyMemory.Byte.)**

Writes data synchronously to an ADS device.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Write(
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The data to write.

Implements

[IAdsReadWrite2.Write\(UInt32, UInt32, ReadOnlyMemory.Byte.\)](#) [[▶ 1004](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[Write Overload](#) [[▶ 615](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.56.8 **AdsConnection.Write Method (UInt32, UInt32, ReadOnlyMemory.Byte., Int32)**

Writes data synchronously to an ADS device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Write(
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory<byte> writeBuffer,
    int timeout
)
```

Parameters

indexGroup	Type: System.UInt32 The index group number of the requested ADS service.
indexOffset	Type: System.UInt32 The index offset number of the requested ADS service.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.
timeout	Type: System.Int32 The timeout.

Implements

[IAdsReadWriteTimeoutAccess.Write\(UInt32, UInt32, ReadOnlyMemory.Byte., Int32\) \[▶ 1011\]](#)

Reference






[AdsConnection Class \[▶ 412\]](#)

[Write Overload \[▶ 615\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.5.2.57 AdsConnection.WriteAny Method

Overload List

	Name	Description
	WriteAny(UInt32, Object) [▶ 621]	Writes an object synchronously to an ADS device.
	WriteAny(UInt32, Object, .Int32.) [▶ 622]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAny(UInt32, UInt32, Object) [▶ 623]	Writes an object synchronously to an ADS device.
	WriteAny(UInt32, UInt32, Object, .Int32.) [▶ 623]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAny(UInt32, UInt32, Object, .Int32., Int32) [▶ 624]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

Reference

[AdsConnection Class \[▶ 412\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.5.2.57.1 AdsConnection.WriteAny Method (UInt32, Object)

Writes an object synchronously to an ADS device.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteAny(
    uint variableHandle,
    Object value
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
value	Type: System.Object Object to write to the ADS device.

Implements

[IAdsAnyAccess.WriteAny\(UInt32, Object\)](#) [► 829]

Reference

[AdsConnection Class](#) [► 412]

[WriteAny Overload](#) [► 621]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.57.2 **AdsConnection.WriteAny Method (UInt32, Object, .Int32.)**

Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteAny(
    uint variableHandle,
    Object value,
    int[]? args
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
value	Type: System.Object Object to write to the ADS device.
args	Type: .System.Int32 . Additional arguments.

Implements

[IAdsAnyAccess.WriteAny\(UInt32, Object, .Int32.\)](#) [[▶ 829](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[WriteAny Overload](#) [[▶ 621](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.57.3 AdsConnection.WriteAny Method (UInt32, UInt32, Object)

Writes an object synchronously to an ADS device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteAny(
    uint indexGroup,
    uint indexOffset,
    Object value
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
value	Type: System.Object Object to write to the ADS device.

Implements

[IAdsAnyAccess.WriteAny\(UInt32, UInt32, Object\)](#) [[▶ 830](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[WriteAny Overload](#) [[▶ 621](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.57.4 AdsConnection.WriteAny Method (UInt32, UInt32, Object, .Int32.)

Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteAny(
    uint indexGroup,
    uint indexOffset,
    Object value,
    int[]? args
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
value	Type: System.Object Object to write to the ADS device.
args	Type: .System.Int32 . Additional arguments.

Implements

[IAdsAnyAccess.WriteAny\(UInt32, UInt32, Object, .Int32.\)](#) [[▶ 830](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[WriteAny Overload](#) [[▶ 621](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.57.5 AdsConnection.WriteAny Method (UInt32, UInt32, Object, .Int32., Int32)

Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteAny(
    uint indexGroup,
    uint indexOffset,
    Object value,
    int[] args,
    int timeout
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.

value	Type: System.Object Object to write to the ADS device.
args	Type: .System.Int32 . Additional arguments.
timeout	Type: System.Int32 The timeout.

Implements

[IAdsReadWriteTimeoutAccess.WriteAny\(UInt32, UInt32, Object, .Int32., Int32\)](#) [▶ 1011]

Exceptions

Exception	Condition
ObjectDisposedException	

Reference





[AdsConnection Class](#) [▶ 412]

[WriteAny Overload](#) [▶ 621]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.58 AdsConnection.WriteAnyAsync Method

Overload List

	Name	Description
	WriteAnyAsync(UInt32, Object, CancellationTok en) [▶ 626]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAnyAsync(UInt32, Object, .Int32., CancellationTok en) [▶ 627]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAnyAsync(UInt32, UInt32, Object, CancellationTok en) [▶ 628]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAnyAsync(UInt32, UInt32, Object, .Int32., CancellationTok en) [▶ 628]	write any as an asynchronous operation.

Reference

[AdsConnection Class](#) [▶ 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.58.1 **AdsConnection.WriteAnyAsync Method (UInt32, Object, CancellationToken)**

Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteAnyAsync(
    uint variableHandle,
    Object value,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
value	Type: System.Object Object to write to the ADS device.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite](#) [► 1187].

A task that represents the asynchronous task operation. The result parameter [ResultWrite](#) [► 1187] of the write operation contains the [ErrorCode](#) [► 1120].

Implements

[IAdsAnyAccess.WriteAnyAsync\(UInt32, Object, CancellationToken\)](#) [► 832]

Remarks

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [► 2927].
string[]	
Array	

Reference

[AdsConnection Class](#) [► 412]

[WriteAnyAsync Overload](#) [► 625]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.58.2 AdsConnection.WriteAnyAsync Method (UInt32, Object, .Int32., CancellationTok

Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteAnyAsync(
    uint variableHandle,
    Object value,
    int[]? args,
    CancellationTok
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
value	Type: System.Object Object to write to the ADS device.
args	Type: .System.Int32 . Additional arguments.
cancel	Type: System.Threading.CancellationTok The cancellation token.

Return Value

Type: [Task.ResultWrite](#) [[▶ 1187](#)].

Task<ResultWrite>.

Implements

[IAdsAnyAccess.WriteAnyAsync\(UInt32, Object, .Int32., CancellationTok\)](#) [[▶ 833](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedExcep tion [▶ 67]	

Remarks

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[WriteAnyAsync Overload \[▶ 625\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.5.2.58.3 **AdsConnection.WriteAnyAsync Method (UInt32, UInt32, Object, CancellationToken)**

Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteAnyAsync(
    uint indexGroup,
    uint indexOffset,
    Object value,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
value	Type: System.Object Object to write to the ADS device.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite \[▶ 1187\]](#).

A task that represents the asynchronous task operation. The result parameter [ResultWrite \[▶ 1187\]](#) of the write operation contains the [ErrorCode \[▶ 1120\]](#).

Implements

[IAdsAnyAccess.WriteAnyAsync\(UInt32, UInt32, Object, CancellationToken\) \[▶ 834\]](#)

Reference

[AdsConnection Class \[▶ 412\]](#)

[WriteAnyAsync Overload \[▶ 625\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.5.2.58.4 **AdsConnection.WriteAnyAsync Method (UInt32, UInt32, Object, .Int32., CancellationToken)**

write any as an asynchronous operation.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteAnyAsync(
    uint indexGroup,
    uint indexOffset,
    Object value,
    int[]? args,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
value	Type: System.Object Object to write to the ADS device.
args	Type: .System.Int32 . Additional arguments.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite](#) [[▶ 1187](#)].

A task that represents the asynchronous task operation. The result parameter [ResultWrite](#) [[▶ 1187](#)] of the write operation contains the [ErrorCode](#) [[▶ 1120](#)].

Implements

[IAdsAnyAccess.WriteAnyAsync\(UInt32, UInt32, Object, .Int32., CancellationToken\)](#) [[▶ 834](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference



[AdsConnection Class](#) [[▶ 412](#)]

[WriteAnyAsync Overload](#) [[▶ 625](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.59 AdsConnection.WriteAnyStringAsync Method

Overload List

	Name	Description
	WriteAnyStringAsync(String, String, Int32, Encoding, CancellationToken) [▶ 630]	write any string as an asynchronous operation.
	WriteAnyStringAsync(UInt32, String, Int32, Encoding, CancellationToken) [▶ 631]	write any string as an asynchronous operation.

Reference

[AdsConnection Class](#) [▶ 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.59.1 AdsConnection.WriteAnyStringAsync Method (String, String, Int32, Encoding, CancellationToken)

write any string as an asynchronous operation.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteAnyStringAsync(
    string symbolPath,
    string value,
    int length,
    Encoding encoding,
    CancellationToken cancel
)
```

Parameters

symbolPath	Type: System.String The symbol path.
value	Type: System.String The value.
length	Type: System.Int32 The length of the string to write
encoding	Type: System.Text.Encoding The encoding.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite \[▸ 1187\]](#).
Task<ResultWrite>.

Remarks

ATTENTION: Potentially this method is unsafe because following data can be overwritten after the string symbol. Please be sure to specify the string length lower than the string size reserved within the process image! The String is written with the specified encoding.

Reference

[AdsConnection Class \[▸ 412\]](#)

[WriteAnyStringAsync Overload \[▸ 630\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.5.2.59.2 AdsConnection.WriteAnyStringAsync Method (UInt32, String, Int32, Encoding, CancellationToken)

write any string as an asynchronous operation.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteAnyStringAsync (
    uint variableHandle,
    string value,
    int length,
    Encoding encoding,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
value	Type: System.String The value.
length	Type: System.Int32 The length of the string to write
encoding	Type: System.Text.Encoding The encoding.
cancel	Type: System.Threading.CancellationToken The cancellation token.




Return Value

Type: [Task.ResultWrite \[▸ 1187\]](#).
Task<ResultWrite>.

Remarks

ATTENTION: Potentially this method is unsafe because following data can be overwritten after the string symbol. Please be sure to specify the string length lower than the string size reserved within the process image! The String is written with the specified encoding.

Reference[AdsConnection Class \[► 412\]](#)[WriteAnyStringAsync Overload \[► 630\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.5.2.60 AdsConnection.WriteAsync Method****Overload List**

	Name	Description
	WriteAsync(UInt32, ReadOnlyMemory.BYTE, CancellationToken) [► 634]	Writes the value data asynchronously that is represented by the writeBuffer to the symbol specified by the variableHandle.
	WriteAsync(UInt32, UInt32, CancellationToken) [► 633]	Triggers a write call at the specified IndexGroup/IndexOffset asynchronously.
	WriteAsync(UInt32, UInt32, ReadOnlyMemory.BYTE, CancellationToken) [► 635]	Writes the data / Value asynchronously into the specified writeBuffer.

Reference[AdsConnection Class \[► 412\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.5.2.60.1 AdsConnection.WriteAsync Method (UInt32, ReadOnlyMemory`1, Void)****Namespace:** [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934 addedca3e72bc0ea15da1c14

Syntax**C#**

```
public Task<ResultWrite> WriteAsync(
    uint variableHandle,
    ReadOnlyMemory writeBuffer,
    void cancel
)
```

ParametersvariableHandle Type: [System.UInt32](#)writeBuffer Type: [ReadOnlyMemory](#)

cancel Type: [System.Void](#)

Return Value

Type: [Task.ResultWrite](#) [[▶ 1187](#)].

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[WriteAsync Overload](#) [[▶ 632](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.60.2 **AdsConnection.WriteAsync Method (UInt32, UInt32, Cancellation token)**

Triggers a write call at the specified IndexGroup/IndexOffset asynchronously.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteAsync(
    uint indexGroup,
    uint indexOffset,
    Cancellation token cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
cancel	Type: System.Threading.Cancellation token The cancellation token.

Return Value

Type: [Task.ResultWrite](#) [[▶ 1187](#)].

A task that represents the asynchronous 'ReadWrite' operation. The [ResultWrite](#) [[▶ 1187](#)] parameter contains the [ErrorCode](#) [[▶ 1120](#)] after execution.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[WriteAsync Overload](#) [[▶ 632](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.60.3 **AdsConnection.WriteAsync Method (UInt32, UInt32, ReadOnlyMemory`1, Void)**

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Task<ResultWrite> WriteAsync(  
    uint indexGroup,  
    uint indexOffset,  
    ReadOnlyMemory writeBuffer,  
    void cancel  
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
writeBuffer	Type: ReadOnlyMemory
cancel	Type: System.Void

Return Value

Type: [Task.ResultWrite](#) [[▶ 1187](#)].

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[WriteAsync Overload](#) [[▶ 632](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.60.4 **AdsConnection.WriteAsync Method (UInt32, ReadOnlyMemory.Byte., CancellationToken)**

Writes the value data asynchronously that is represented by the writeBuffer to the symbol specified by the variableHandle.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteAsync(  
    uint variableHandle,  
    ReadOnlyMemory<byte> writeBuffer,  
    CancellationToken cancel  
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer/value.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite](#) [► 1187].

A task that represents the asynchronous write operation. The [ResultWrite](#) [► 1187] parameter contains the [ErrorCode](#) [► 1120] after execution.

Implements

[IAdsHandle.WriteAsync\(UInt32, ReadOnlyMemory.Byte., CancellationToken\)](#) [► 965]

Reference

[AdsConnection Class](#) [► 412]

[WriteAsync Overload](#) [► 632]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.60.5 **AdsConnection.WriteAsync Method (UInt32, UInt32, ReadOnlyMemory.Byte., CancellationToken)**

Writes the data / Value asynchronously into the specified writeBuffer.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultWrite> WriteAsync(
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory<byte> writeBuffer,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite](#) [[▶ 1187](#)].

A task that represents the asynchronous 'Write' operation. The [ResultWrite](#) [[▶ 1187](#)] parameter contains the [ErrorCode](#) [[▶ 1120](#)] after execution.

Implements

[IAdsReadWrite.WriteAsync\(UInt32, UInt32, ReadOnlyMemory.Byte., CancellationToken\)](#) [[▶ 1000](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	





Reference

[AdsConnection Class](#) [[▶ 412](#)]

[WriteAsync Overload](#) [[▶ 632](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.61 AdsConnection.WriteControl Method**Overload List**

	Name	Description
	WriteControl(StateInfo) [▶ 637]	Changes the ADS status and the device status of an ADS server.
	WriteControl(StateInfo, Int32) [▶ 637]	Changes the ADS status and the device status of an ADS server.
	WriteControl(StateInfo, ReadOnlyMemory.Byte.) [▶ 639]	Changes the ADS status and the device status of an ADS server.
	WriteControl(StateInfo, ReadOnlyMemory.Byte., Int32) [▶ 640]	Changes the ADS status and the device status of an ADS server.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.61.1 **AdsConnection.WriteControl Method (StateInfo)**

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteControl(  
    StateInfo stateInfo  
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200] New ADS status and device status.
-----------	---

Implements

[IAdsStateControl.WriteControl\(StateInfo\)](#) [[▶ 1044](#)]

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[WriteControl Overload](#) [[▶ 636](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.61.2 **AdsConnection.WriteControl Method (StateInfo, Int32)**

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteControl(  
    StateInfo stateInfo,  
    int timeout  
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200] New ADS status and device status.
timeout	Type: System.Int32 The timeout.

Implements

[IAdsStateControlTimeout.WriteControl\(StateInfo, Int32\)](#) [[▶ 1053](#)]

Exceptions

Exception	Condition
ObjectDisposedException	

Reference

[AdsConnection Class](#) [► 412]

[WriteControl Overload](#) [► 636]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.61.3 **AdsConnection.WriteControl Method (StateInfo, ReadOnlyMemory`1)**

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax**C#**

```
public void WriteControl(
    StateInfo stateInfo,
    ReadOnlyMemory writeBuffer
)
```

Parameters

stateInfo Type: [TwinCAT.Ads.StateInfo](#) [► 1200]

writeBuffer Type: [ReadOnlyMemory](#)

Reference

[AdsConnection Class](#) [► 412]

[WriteControl Overload](#) [► 636]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.61.4 **AdsConnection.WriteControl Method (StateInfo, ReadOnlyMemory`1, Void)**

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax**C#**

```
public void WriteControl(
    StateInfo stateInfo,
    ReadOnlyMemory writeBuffer,
    void timeout
)
```

Parameters

- stateInfo Type: [TwinCAT.Ads.StateInfo](#) [[▶ 1200](#)]
- writeBuffer Type: [ReadOnlyMemory](#)
- timeout Type: [System.Void](#)

Reference

- [AdsConnection Class](#) [[▶ 412](#)]
- [WriteControl Overload](#) [[▶ 636](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.61.5 AdsConnection.WriteControl Method (StateInfo, ReadOnlyMemory.Byte.)

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: [TwinCAT.Ads](#) (in [TwinCAT.Ads.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteControl(
    StateInfo stateInfo,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200] New ADS status and device status.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.

Implements

[IAdsStateControl.WriteControl\(StateInfo, ReadOnlyMemory.Byte.\)](#) [[▶ 1045](#)]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

- [AdsConnection Class](#) [[▶ 412](#)]
- [WriteControl Overload](#) [[▶ 636](#)]

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.5.2.61.6 **AdsConnection.WriteControl Method (StateInfo, ReadOnlyMemory.Byte., Int32)**

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteControl(
    StateInfo stateInfo,
    ReadOnlyMemory<byte> writeBuffer,
    int timeout
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▸ 1200] New ADS status and device status.
writeBuffer	Type: System.ReadOnlyMemory.Byte. The write buffer.
timeout	Type: System.Int32 The timeout.

Implements

[IAdsStateControlTimeout.WriteControl\(StateInfo, ReadOnlyMemory.Byte., Int32\) \[▸ 1054\]](#)

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▸ 67]	

Reference



[AdsConnection Class \[▸ 412\]](#)

[WriteControl Overload \[▸ 636\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.5.2.62 AdsConnection.WriteControlAsync Method

Overload List

	Name	Description
	WriteControlAsync(AdsState, UInt16, CancellationToken) [▶ 641]	Changes the ADS status and device status of the ADS server asynchronously.
	WriteControlAsync(AdsState, UInt16, ReadOnlyMemory<byte>, CancellationToken) [▶ 642]	Writes the AdsState [▶ 729] and device state to the ADS device.

Reference

[AdsConnection Class](#) [▶ 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.62.1 AdsConnection.WriteControlAsync Method (AdsState, UInt16, CancellationToken)

Changes the ADS status and device status of the ADS server asynchronously.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAds> WriteControlAsync(
    AdsState state,
    ushort deviceState,
    CancellationToken cancel
)
```

Parameters

state	Type: TwinCAT.Ads.AdsState [▶ 729] The ADS state.
deviceState	Type: System.UInt16 The device state.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds](#) [▶ 1116].

A task that represents the asynchronous 'WriteControl' operation. The [ResultAds](#) [▶ 1116] parameter contains the state the [ErrorCode](#) [▶ 1120] of the ADS communication after execution.

Implements

[IAdsStateControl.WriteControlAsync\(AdsState, UInt16, CancellationToken\)](#) [► 1046]

Reference

[AdsConnection Class](#) [► 412]

[WriteControlAsync Overload](#) [► 641]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.62.2 **AdsConnection.WriteControlAsync Method (AdsState, UInt16, ReadOnlyMemory`1, Void)**

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934 addedca3e72bc0ea15da1c14

Syntax

C#

```
public Task<ResultAds> WriteControlAsync(  
    AdsState adsState,  
    ushort deviceState,  
    ReadOnlyMemory writeBuffer,  
    void cancel  
)
```

Parameters

adsState	Type: TwinCAT.Ads.AdsState [► 729]
deviceState	Type: System.UInt16
writeBuffer	Type: ReadOnlyMemory
cancel	Type: System.Void

Return Value

Type: [Task.ResultAds](#) [► 1116].

Reference

[AdsConnection Class](#) [► 412]

[WriteControlAsync Overload](#) [► 641]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.2.62.3 **AdsConnection.WriteControlAsync Method (AdsState, UInt16, ReadOnlyMemory.Byte., CancellationToken)**

Writes the [AdsState](#) [► 729] and device state to the ADS device.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAds> WriteControlAsync(
    AdsState adsState,
    ushort deviceState,
    ReadOnlyMemory<byte> writeBuffer,
    CancellationToken cancel
)
```

Parameters

adsState	Type: TwinCAT.Ads.AdsState [▶ 729] State of the ads.
deviceState	Type: System.UInt16 State of the device.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds](#) [▶ 1116].

A task that represents the asynchronous 'ReadState' operation. The [ResultAds](#) [▶ 1116] parameter contains the [ErrorCode](#) [▶ 1120] of the ADS communication after execution.

Implements

[IAdsStateControl.WriteControlAsync\(AdsState, UInt16, ReadOnlyMemory.Byte., CancellationToken\)](#) [▶ 1047]

Reference

[AdsConnection Class](#) [▶ 412]

[WriteControlAsync Overload](#) [▶ 641]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.63 AdsConnection.WriteSymbolAsync Method

Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteSymbolAsync(
    string name,
    Object value,
    CancellationToken cancel
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
------	--

value	Type: System.Object Object holding the value to be written to the ADS symbol
cancel	Type: System.Threading.CancellationToken The cancel token.

Return Value

Type: [Task.ResultWrite](#) [▶ 1187].

A task that represents the asynchronous 'WriteSymbol' operation. The [ResultWrite](#) [▶ 1187] parameter contains the [ErrorCode](#) [▶ 1120] after execution.

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	





Reference

[AdsConnection Class](#) [▶ 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.64 AdsConnection.WriteValue Method

Overload List

	Name	Description
	WriteValue(ISymbol, Object) [▶ 646]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValue.T.(ISymbol, T) [▶ 646]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValue(String, Object) [▶ 644]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	WriteValue.T.(String, T) [▶ 645]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Reference

[AdsConnection Class](#) [▶ 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.64.1 AdsConnection.WriteValue Method (String, Object)

Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteValue(
    string name,
    Object value
)
```

Parameters

name	Type: <u>System.String</u> Name of the ADS symbol.
value	Type: <u>System.Object</u> Object holding the value to be written to the ADS symbol

Implements

[IAdsSymbolicAccess.WriteValue\(String, Object\) \[▶ 1088\]](#)

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsConnection Class \[▶ 412\]](#)

[WriteValue Overload \[▶ 644\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.5.2.64.2 AdsConnection.WriteValue.T. Method (String, T)

Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteValue<T>(
    string name,
    T value
)
```

Parameters

name	Type: <u>System.String</u> Name of the ADS symbol.
value	Type: T Object holding the value to be written to the ADS symbol

Type Parameters

T	the value type.
---	-----------------

Implements

[IAdsSymbolicAccess.WriteValue.T.\(String, T\) \[► 1089\]](#)

Reference

[AdsConnection Class \[► 412\]](#)

[WriteValue Overload \[► 644\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.64.3 AdsConnection.WriteValue Method (ISymbol, Object)

Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteValue(
    ISymbol symbol,
    Object val
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol the value is written to.
val	Type: System.Object The value to write.

Implements

[IAdsSymbolicAccess.WriteValue\(ISymbol, Object\) \[► 1089\]](#)

Reference

[AdsConnection Class \[► 412\]](#)

[WriteValue Overload \[► 644\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.2.64.4 AdsConnection.WriteValue.T. Method (ISymbol, T)

Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteValue<T>(
    ISymbol symbol,
    T val
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol the value is written to.
val	Type: T The value to write.

Type Parameters

T	The value type.
---	-----------------

Implements

[IAdsSymbolicAccess.WriteValue.T.\(ISymbol, T\)](#) [[▶ 1090](#)]

Reference




[AdsConnection Class](#) [[▶ 412](#)]

[WriteValue Overload](#) [[▶ 644](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.65 AdsConnection.WriteValueAsync Method

Overload List

	Name	Description
	WriteValueAsync(ISymbol, Object, CancellationToken) [▶ 648]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValueAsync.T.(ISymbol, T, CancellationToken) [▶ 649]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValueAsync.T.(String, T, CancellationToken) [▶ 648]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.65.1 **AdsConnection.WriteValueAsync.T. Method (String, T, CancellationToken)**

Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteValueAsync<T>(
    string name,
    T value,
    CancellationToken cancel
)
```

Parameters

name	Type: System.String Name of the ADS symbol.
value	Type: T Object holding the value to be written to the ADS symbol
cancel	Type: System.Threading.CancellationToken The cancel token.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [Task.ResultWrite](#) [▶ 1187].

A task that represents the asynchronous 'WriteSymbol' operation. The [ResultWrite](#) [▶ 1187] parameter contains the [ErrorCode](#) [▶ 1120] after execution.

Implements

[IAdsSymbolicAccess.WriteValueAsync.T.\(String, T, CancellationToken\)](#) [▶ 1091]

Reference

[AdsConnection Class](#) [▶ 412]

[WriteValueAsync Overload](#) [▶ 647]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.65.2 **AdsConnection.WriteValueAsync Method (ISymbol, Object, CancellationToken)**

Writes a value to the symbol. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. Array and structures are not supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteValueAsync(
    ISymbol symbol,
    Object val,
    CancellationToken cancel
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol the value is written to.
val	Type: System.Object The value to write.
cancel	Type: System.Threading.CancellationToken The cancel token.

Return Value

Type: [Task.ResultWrite](#) [▶ 1187].

Task.

Implements

[IAdsSymbolicAccess.WriteValueAsync\(ISymbol, Object, CancellationToken\)](#) [▶ 1092]

Exceptions

Exception	Condition
ObjectDisposedException	
ClientNotConnectedException [▶ 67]	

Reference

[AdsConnection Class](#) [▶ 412]

[WriteValueAsync Overload](#) [▶ 647]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.2.65.3 AdsConnection.WriteValueAsync.T. Method (ISymbol, T, CancellationToken)

Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWrite> WriteValueAsync<T>(
    ISymbol symbol,
    T val,
    CancellationToken cancel
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▸ 2691] The symbol the value is written to.
val	Type: T The value to write.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [Task.ResultWrite \[▸ 1187\]](#).

A task that represents the asynchronous 'WriteSymbol' operation. The [ResultWrite \[▸ 1187\]](#) parameter contains the [ErrorCode \[▸ 1120\]](#) after execution.

Implements

[IAdsSymbolicAccess.WriteValueAsync.T.\(ISymbol, T, CancellationToken\) \[▸ 1093\]](#)

Reference

[AdsConnection Class \[▸ 412\]](#)

[WriteValueAsync Overload \[▸ 647\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.5.2.66 AdsConnection.RegisterSymbolVersionChanged Method

Registers the symbol version changed.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode RegisterSymbolVersionChanged(
    EventHandler<AdsSymbolVersionChangedEventArgs> handler
)
```

Parameters

handler	Type: System.EventHandler.AdsSymbolVersionChangedEventArgs [▸ 739] . The handler function to register.
---------	---

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
 AdsErrorCode.

Implements

[IAdsSymbolChangedProvider.RegisterSymbolVersionChanged\(EventHandler.AdsSymbolVersionChangedEventArgs.\)](#) [[▶ 1064](#)]

Exceptions

Exception	Condition
ObjectDisposedException	

Reference

[AdsConnection Class](#) [[▶ 412](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.5.2.67 AdsConnection.UnregisterSymbolVersionChanged Method

Unregisters the symbol version changed.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode UnregisterSymbolVersionChanged(
    EventHandler<AdsSymbolVersionChangedEventArgs> handler
)
```

Parameters

handler	Type: System.EventHandler.AdsSymbolVersionChangedEventArgs [▶ 739]. The handler function to unregister.
---------	--

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
 AdsErrorCode.

Implements

[IAdsSymbolChangedProvider.UnregisterSymbolVersionChanged\(EventHandler.AdsSymbolVersionChangedEventArgs.\)](#) [[▶ 1064](#)]

Exceptions

Exception	Condition
ObjectDisposedException	

Reference













[AdsConnection Class \[▶ 412\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.5.3 AdsConnection Events

The [AdsConnection \[▶ 412\]](#) type exposes the following members.

Events

	Name	Description
 	AdsNotification [▶ 652]	Occurs when the ADS device sends a notification to the client.
	AdsNotificationError [▶ 654]	Occurs when a exception has occurred during notification management.
 	AdsNotificationEx [▶ 654]	Occurs when the ADS devices sends a notification to the client.
	AdsStateChanged [▶ 655]	Occurs when ADS State has been changed.
 	AdsSumNotification [▶ 656]	Occurs when Notifications are send (bundled notifications)
	AdsSymbolVersionC hanged [▶ 657]	Occurs when the symbol version has been changed.
 	ConnectionStateCha nged [▶ 658]	Occurs when connection status of the AdsConnection [▶ 412] has been changed.
	RouterStateChange d [▶ 659]	(Local) Router state changed event.

Reference

[AdsConnection Class \[▶ 412\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.5.3.1 AdsConnection.AdsNotification Event

Occurs when the ADS device sends a notification to the client.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<AdsNotificationEventArgs> AdsNotification
```

Value

Type: [System.EventHandler.AdsNotificationEventArgs](#) [► 695].

Implements

[IAdsNotifications.AdsNotification](#) [► 991]

Exceptions

Exception	Condition
ObjectDisposedException	

Remarks

The Event Argument contains the raw data value of the notification, not marshalled to .NET types.

Examples

Example of receiving [AdsNotification](#) [► 991] events.

Trigger on changed values by ADS Notifications

```
private async Task RegisterNotificationsAsync()
{
    CancellationToken cancel = CancellationToken.None;

    using (AdsClient client = new AdsClient())
    {
        // Add the Notification event handler
        client.AdsNotification += Client_AdsNotification2;

        // Connect to target
        client.Connect(AmsNetId.Local, 851);
        uint notificationHandle = 0;

        // Notification to a DINT Type (UINT32)
        // Check for change every 200 ms

        //byte[] notificationBuffer = new byte[sizeof(UInt32)];
        int size = sizeof(UInt32);

        ResultHandle result = await client.AddDeviceNotificationAsync("MAIN.nCounter", size, new NotificationSettings(AdsTransMode.OnChange, 200, 0), null, cancel);

        if (result.Succeeded)
        {
            notificationHandle = result.Handle;
            await Task.Delay(5000); // Wait asynchronously without blocking the UI Thread.
            // Unregister the Event / Handle
            ResultAds result2 = await client.DeleteDeviceNotificationAsync(notificationHandle, cancel);
        }
        client.AdsNotification -= Client_AdsNotification2;
    }
}

private void Client_AdsNotification2(object sender, AdsNotificationEventArgs e)
{
    // Or here we know about UDINT type --> can be marshalled as UINT32
    uint nCounter = BinaryPrimitives.ReadUInt32LittleEndian(e.Data.Span);

    // If Synchronization is needed (e.g. in Windows.Forms or WPF applications)
    // we could synchronize via SynchronizationContext into the UI Thread

    /*SynchronizationContext syncContext = SynchronizationContext.Current;
    _context.Post(status => someLabel.Text = nCounter.ToString(), null); // Non-blocking post */
}
```

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.5.3.2 **AdsConnection.AdsNotificationError Event**

Occurs when an exception has occurred during notification management.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<AdsNotificationErrorEventArgs> AdsNotificationError
```

Value

Type: [System.EventHandler.AdsNotificationErrorEventArgs \[► 694\]](#).

Implements

[IAdsNotifications.AdsNotificationError \[► 993\]](#)

Remarks

The occurrence of this event can have two different reasons:

1. Indicates an internal error occurred during Notification management.
2. The registered notification becomes invalid on the server, eg. after a PLC Download / Online Change. If the ADS Server detects that the (still registered) Notification Sender is getting invalid, it sends an error notification so that the client will be informed about detached notifications. The event arguments contains the [AdsInvalidNotificationException \[► 690\]](#) which describes the invalid notification handle by its [Handle \[► 692\]](#) property.

Reference

[AdsConnection Class \[► 412\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

[TwinCAT.Ads.AdsInvalidNotificationException \[► 690\]](#)

[AdsConnection.AdsNotification \[► 652\]](#)

[AdsConnection.AdsNotificationEx \[► 654\]](#)

6.2.5.3.3 **AdsConnection.AdsNotificationEx Event**

Occurs when the ADS device sends a notification to the client.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<AdsNotificationExEventArgs> AdsNotificationEx
```

Value

Type: [System.EventHandler.AdsNotificationExEventArgs \[▶ 699\]](#).

Implements

[IAdsNotifications.AdsNotificationEx \[▶ 993\]](#)

Exceptions

Exception	Condition
ObjectDisposedException	

Remarks

The Notification event arguments marshals the data value automatically to the specified .NET Type with ANY_TYPE marshallers.

Examples

Example of receiving [AdsNotificationEx \[▶ 993\]](#) events.

Trigger on changed values by ADS Notifications

```
CancellationToken cancel = CancellationToken.None;

using (AdsClient client = new AdsClient())
{
    client.AdsNotificationEx += Client_AdsNotificationEx;
    client.Connect(AmsNetId.Local, 851);

    // Add UDINT
    ResultHandle resultHandle = await client.AddDeviceNotificationExAsync("MAIN.udint", new NotificationSettings(AdsTransMode.OnChange, 200, 200), null, typeof(uint), null, cancel);
    await Task.Delay(5000, cancel); // Wait ...
    ResultAds resultHandleDelete = await client.DeleteDeviceNotificationAsync(resultHandle.Handle, cancel); // Unregister Event
}
```

Reference

[AdsConnection Class \[▶ 412\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.5.3.4 AdsConnection.AdsStateChanged Event

Occurs when ADS State has been changed.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<AdsStateChangedEventArgs> AdsStateChanged
```

Value

Type: [System.EventHandler.AdsStateChangedEventArgs \[▶ 729\]](#).

Implements

[IAdsStateProvider.AdsStateChanged](#) [► 1061]

Exceptions

Exception	Condition
ObjectDisposedException	

Remarks

This event occurs asynchronously if the synchronized flag is not set.

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.5.3.5 AdsConnection.AdsSumNotification Event

Occurs when Notifications are send (bundled notifications)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<AdsSumNotificationEventArgs> AdsSumNotification
```

Value

Type: [System.EventHandler.AdsSumNotificationEventArgs](#).

Implements

[IAdsNotifications.AdsSumNotification](#) [► 994]

Exceptions

Exception	Condition
ObjectDisposedException	

Remarks

As an optimization, this event receives all ADS Notifications that occurred at one point in time together. As consequence, the overhead of handler code is reduced, what can be important if notifications are triggered in a high frequency and the event has to be synchronized to the UI thread context. Because multiple notifications are bound together, less thread synchronization is necessary. The [AdsNotification](#) [► 991] and [AdsNotificationEx](#) [► 993] events shouldn't be used when SumNotifications are registered, because they have an performance side effect to this [AdsSumNotification](#) [► 994] event. The full performance is reached only, when all notifications are handled on this event.

Examples

Example of receiving [AdsSumNotification](#) [► 994] events.

Trigger on changed values by ADS Notifications

```
private async Task RegisterSumNotificationsAsync()
{
    CancellationToken cancel = CancellationToken.None;

    using (AdsClient client = new AdsClient())
    {
        // Add the Notification event handler
        client.AdsSumNotification += Client_SumNotification;

        // Connect to target
        client.Connect(AmsNetId.Local, 851);
        uint notificationHandle = 0;

        // Notification to a DINT Type (UINT32)
        // Check for change every 200 ms

        //byte[] notificationBuffer = new byte[sizeof(UInt32)];

        ResultHandle result = await client.AddDeviceNotificationAsync("MAIN.nCounter", sizeof(UInt32), new NotificationSettings(AdsTransMode.OnChange, 200, 0), null, cancel);

        if (result.Succeeded)
        {
            notificationHandle = result.Handle;
            await Task.Delay(5000); // Wait asynchronously without blocking the UI Thread.
            // Unregister the Event / Handle
            ResultAds result2 = await client.DeleteDeviceNotificationAsync(notificationHandle, cancel);
        }
        client.AdsNotification -= Client_AdsNotification2;
    }
}

private void Client_SumNotification(object sender, AdsSumNotificationEventArgs e)
{
    // Timestamp of the Notification List
    DateTimeOffset dateTime = e.TimeStamp;

    // List of Raw ADS Notifications
    IList<Notification> notifications = e.Notifications;

    foreach(Notification notification in notifications)
    {
        // Notifications can be handled more efficiently, because they occur together
        // handler and can be transformed/
        synchronized in one step compared to AdsClient.AdsNotification events.
    }
}
```

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsNotifications.AdsNotification](#) [► 991]

6.2.5.3.6 AdsConnection.AdsSymbolVersionChanged Event

Occurs when the symbol version has been changed.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<AdsSymbolVersionChangedEventArgs> AdsSymbolVersionChanged
```

Value

Type: [System.EventHandler.AdsSymbolVersionChangedEventArgs](#) [▶ 739].

Implements

[IAdsSymbolChangedProvider.AdsSymbolVersionChanged](#) [▶ 1065]

Remarks

This is the case when the connected ADS server restarts. This invalidates all actual opened symbol handles. The SymbolVersion counter doesn't trigger, when an online change is made on the PLC (ports 801, ..., 851 ...)

Reference

[AdsConnection Class](#) [▶ 412]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.5.3.7 AdsConnection.ConnectionStateChanged Event

Occurs when connection status of the [AdsConnection](#) [▶ 412] has been changed.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public event EventHandler<ConnectionStateChangedEventArgs> ConnectionStateChanged
```

Value

Type: [System.EventHandler.ConnectionStateChangedEventArgs](#) [▶ 74].

Implements

[IConnectionStateProvider.ConnectionStateChanged](#) [▶ 91]

Remarks

The Connection state changes only if the [IConnection](#) [▶ 79] is established / shut down or active communication is triggered by the User of the [IConnection](#) [▶ 79] object.

Examples

The following sample shows how to keep the [ConnectionState](#) [▶ 438] updated by triggering ADS Communication.

Trigger ConnectionState changes in WPF Applications

```
private DispatcherTimer _timer = null;
private AdsSession _session = null;
//private AdsConnection _connection = null;

private void Window_Loaded(object sender, RoutedEventArgs e)
{
    _session = new AdsSession(AmsNetId.Local, 10000);
    IConnection connection = _session.Connect();
    tbConnectionState.Text = connection.ConnectionState.ToString();
}
```

```

    _session.ConnectionStateChanged += _session_ConnectionStateChanged;

    _timer = new DispatcherTimer();
    _timer.Interval = TimeSpan.FromMilliseconds(200);
    _timer.Tick += TimerOnTick;

    _timer.Start();
}

private void Window_Unloaded(object sender, RoutedEventArgs e)
{
    _timer.Stop();
    _session.Dispose();
}

private void _session_ConnectionStateChanged(object sender, TwinCAT.ConnectionStateChangedEventArgs e)
{
    // ConnectionStateChanged will be triggered by communication Invokes
    tbConnectionState.Text = e.NewState.ToString();
}

private void TimerOnTick(object sender, EventArgs eventArgs)
{
    // The Timer Event will occur here in the UI thread because its an DispatcherTimer event!
    // An active ADS request will trigger Connection State periodically!
    StateInfo stateInfo;
    if (_session.Connection.TryReadState(out stateInfo) == AdsErrorCode.NoError)
    {
        tbAdsState.Text = stateInfo.AdsState.ToString();
    }
    else
    {
        tbAdsState.Text = "Invalid";
    }
}
}

```

Reference

[AdsConnection Class \[► 412\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

[AdsConnection.ConnectionState \[► 438\]](#)

6.2.5.3.8 AdsConnection.RouterStateChanged Event

(Local) Router state changed event.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<AmsRouterNotificationEventArgs> RouterStateChanged
```

Value

Type: [System.EventHandler.AmsRouterNotificationEventArgs \[► 798\]](#).

Remarks

This event indicates, that a changed event is received from the Local AmsRouter independant of the connected target address. A remote system RouterStateChanged event cannot be received at another system - it cannot traverse TwinCAT systems.

Reference

[AdsConnection Class](#) [► 412]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.6 AdsDataTypeArrayInfo Class

Array definition for a single dimension.

Inheritance Hierarchy

[System.Object](#)

TwinCAT.Ads.AdsDataTypeArrayInfo

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229



Syntax

C#









```
public class AdsDataTypeArrayInfo
```

The [AdsDataTypeArrayInfo](#) type exposes the following members.

Properties

	Name	Description
	Elements [► 661]	Gets the number of elements.
	LowerBound [► 661]	Gets the lower bound.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	Marshal [► 662]	Marshals the content of the AdsDataTypeArrayInfo .
	MarshalSize [► 663]	Gets the marshal size of the AdsDataTypeArrayInfo .
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)



Reference

[TwinCAT.Ads Namespace](#) [► 179]

6.2.6.1 AdsDataTypeArrayInfo Properties

The [AdsDataTypeArrayInfo](#) [► 660] type exposes the following members.

Properties

	Name	Description
	Elements [▶ 661]	Gets the number of elements.
	LowerBound [▶ 661]	Gets the lower bound.

Reference

[AdsDataTypeArrayInfo Class](#) [[▶ 660](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.6.1.1 AdsDataTypeArrayInfo.Elements Property

Gets the number of elements.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Elements { get; }
```

Property Value

Type: [Int32](#)

Reference

[AdsDataTypeArrayInfo Class](#) [[▶ 660](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.6.1.2 AdsDataTypeArrayInfo.LowerBound Property

Gets the lower bound.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int LowerBound { get; }
```

Property Value

Type: [Int32](#)

Reference









[AdsDataTypeArrayInfo Class](#) [[▶ 660](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.6.2 AdsDataTypeArrayInfo Methods

The [AdsDataTypeArrayInfo](#) [► 660] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	Marshal [► 662]	Marshals the content of the AdsDataTypeArrayInfo [► 660].
	MarshalSize [► 663]	Gets the marshal size of the AdsDataTypeArrayInfo [► 660].
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[AdsDataTypeArrayInfo Class](#) [► 660]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.6.2.1 AdsDataTypeArrayInfo.Marshal Method

Marshals the content of the [AdsDataTypeArrayInfo](#) [► 660].

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Marshal(
    IStringMarshaler marshaler,
    Span<byte> buffer
)
```

Parameters

marshaler	Type: TwinCAT.TypeSystem.IStringMarshaler [► 2650] The marshaler.
buffer	Type: System.Span.Byte . The buffer.

Return Value

Type: [Int32](#)
System.Int32.

Reference

[AdsDataTypeArrayInfo Class](#) [► 660]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.6.2.2 AdsDataTypeArrayInfo.MarshalSize Method

Gets the marshal size of the [AdsDataTypeArrayInfo](#) [▶ 660].

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int MarshalSize(
    IStringMarshaler marshaler
)
```

Parameters

marshaler	Type: TwinCAT.TypeSystem.IStringMarshaler [▶ 2650] The symbol string marshaler
-----------	---

Return Value

Type: [Int32](#)
System.Int32.

Reference

[AdsDataTypeArrayInfo Class](#) [▶ 660]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.7 AdsDataTypeId Enumeration

ADS data types.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum AdsDataTypeId
```

Members

	Member name	Value	Description
	ADST_VOID	0	Empty Type (0)
	ADST_INT8	16	Integer 8 Bit (16)
	ADST_UINT8	17	Unsigned integer 8 Bit (17)
	ADST_INT16	2	Integer 16 Bit (2)
	ADST_UINT16	18	Unsigned integer 16 Bit (18)
	ADST_INT32	3	Integer 32 Bit (3)
	ADST_UINT32	19	Unsigned Integer 32 Bit (19)
	ADST_INT64	20	LONG Integer 64 Bit (20)
	ADST_UINT64	21	Unsigned Long integer 64 Bit (21)
	ADST_REAL32	4	Real (32 Bit) (4)

	Member name	Value	Description
	ADST_REAL64	5	Real 64 Bit (5)
	ADST_BIGTYPE	65	Blob (65)
	ADST_STRING	30	STRING (30)
	ADST_WSTRING	31	WSTRING (31)
	ADST_REAL80	32	ADS REAL80 (32)
	ADST_BIT	33	ADS BIT (33)
	ADST_VARIANT	12	ADS VARIANT (12)

Reference

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.8 AdsErrorCode Enumeration

ADS request return codes.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum AdsErrorCode
```

Members

	Member name	Value	Description
	None	-1	None / Uninitialized. Category: Global This indicates an uninitialized value.
	NoError	0	No Error / Succeeded. Error code: 0 (0x000). Category: Global C++: ERR_NOERROR The ADS request succeeded.
	InternalError	1	Internal Error. Error code: 1 (0x001). Category: Global C++: ERR_INTERNAL This error is very generic and there is no further information available about the problem.
	NoRTime	2	No Realtime available. Error code: 2 (0x002). Category: Global C++: ERR_NORTIME
	LockedMemoryError	3	Allocation locked memory error. Error code: 3 (0x003). Category: Global C++: ERR_ALLOCLOCKEDMEM
	MailBoxError	4	Mailbox error.

	Member name	Value	Description
			Error code: 4 (0x004). Category: Global C++: ERR_INSERTMAILBOX The Mailbox is full – the ADS message could not be sent. Reducing the number of ADS messages per cycle may help.
	WrongHMsg	5	Wrong receive HMSG. Error code: 5 (0x005). Category: Global C++: ERR_WRONGRECEIVEHMSG
	TargetPortNotFound	6	The Target port is not found. Error code: 6 (0x006). Category: Global C++: ERR_TARGETPORTNOTFOUND The ADS server is not started or is not reachable. The target machine could be found, but the Port is not available.
	TargetMachineNotFound	7	The Target machine is not found. Error code: 7 (0x007). Category: Global C++: ERR_TARGETMACHINENOTFOUND The Target computer / AMS route was not found. This Error typically occurs when the Route is not registered at the requesting system.
	UnknownCommandID	8	Unknown command ID. Error code: 8 (0x008). Category: Global C++: ERR_UNKNOWNCMDID
	BadTaskID	9	Bad task ID. Error code: 9 (0x009). Category: Global C++: ERR_BADTASKID
	NoIO	10	No IO. Error code: 10 (0x00A). Category: Global C++: ERR_NOIO
	UnknownAmsCommand	11	Unknown AMS command. Error code: 11 (0x00B). Category: Global C++: ERR_UNKNOWNAMSCMD
	Win32Error	12	Win 32 error. Error code: 12 (0x00C). Category: Global

	Member name	Value	Description
			C++: ERR_WIN32ERROR
	PortNotConnected	13	The Port is not connected. Error code: 13 (0x00D). Category: Global C++: ERR_PORTNOTCONNECTED
	InvalidAmsLength	14	Invalid AMS length. Error code: 14 (0x00E). Category: Global C++: ERR_INVALIDAMSLength
	InvalidAmsNetID	15	Invalid AMS Net ID. Error code: 15 (0x00F). Category: Global C++: ERR_INVALIDAMSNETID
	LowInstallLevel	16	Low Installation level. Error code: 16 (0x010). Category: Global C++: ERR_LOWINSTLEVEL Installation level is too low – TwinCAT 2 license error.
	NoDebug	17	No debug available. Error code: 17 (0x011). Category: Global C++: ERR_NODEBUGINTAVAILABLE
	PortDisabled	18	Port disabled. Error code: 18 (0x012). Category: Global C++: ERR_PORTDISABLED The port is disabled when the TwinCAT system service not in an active state (config, running). For the time of a Reconfig/Restart the included TwinCAT Router disables all RouterPorts.
	PortConnected	19	The Port is already connected. Error code: 19 (0x013). Category: Global C++: ERR_PORTALREADYCONNECTED
	AmsSyncWin32Error	20	AMS Sync Win32 error. Error code: 20 (0x014). C++: ERR_AMSSYNC_W32ERROR
	SyncTimeOut	21	AMS Sync timeout. Error code: 21 (0x015). Category: Global C++: ERR_AMSSYNC_TIMEOUT
	AmsSyncAmsError	22	AMS Sync AMS error Error code: 22 (0x016).

	Member name	Value	Description
			Category: Global C++: ERR_AMSSYNC_AMSERROR
	AmsSyncNoIndexMap	23	No index map for AMS Sync available. Error code: 23 (0x017). Category: Global C++: ERR_AMSSYNC_NOINDEXINMAP
	InvalidAmsPort	24	Invalid AMS port. Error code: 24 (0x018). Category: Global C++: ERR_INVALIDAMSPORT
	NoMemory	25	No memory. Error code: 25 (0x019). Category: Global C++: ERR_NOMEMORY
	TCPSendError	26	TCP send error. Error code: 26 (0x01A). Category: Global C++: ERR_TCPSEND
	HostUnreachable	27	The Host is unreachable. Error code: 27 (0x1B). Category: Global C++: , ERR_HOSTUNREACHABLE
	AmsInvalidFragment	28	Invalid AMS fragment. Error code: 28 (0x1C). Category: Global C++: ERR_INVALIDAMSFAGMENT
	TlsSendError	29	TLS send error – secure ADS connection failed. Error code: 29 (0x1D) Category: Global C++: ERR_TLSEND
	AccessDenied	30	Access denied – secure ADS access denied. Error code: 30 (0x1E) Category: Global C++: ERR_ACCESSDENIED
	TcpConnectionRefused	31	TCP Connection refused Error code: 31 (0x1F) Category: Global C++: ERR_TCP_CONN_REFUSED
	NoLockedMemory	1280	Router: no locked memory. Error code: 1280 (0x500). Category: Router

	Member name	Value	Description
			C++: ROUTERERR_NOLOCKEDMEMORY Locked memory cannot be allocated.
	ResizeMemory	1281	Router: The size of the router memory could not be changed. Error code: 1281 (0x501). Category: Router C++: ROUTERERR_RESIZEMEMORY The router memory size could not be changed.
	MailboxFull	1282	Router: mailbox full. Error code: 1282 (0x502). Category: Router C++: ROUTERERR_MAILBOXFULL The mailbox has reached the maximum number of possible messages.
	DebugBoxFull	1283	Router: The mailbox has reached the maximum number of possible messages. Error code: 1284 (0x504). Category: Router C++: ROUTERERR_DEBUGBOXFULL The Debug mailbox has reached the maximum number of possible messages.
	UnknownPortType	1284	Router: Unknown Port Type Error code: 1284 (0x504). Category: Router C++: ROUTERERR_UNKNOWNPORTTYPE The port type is unknown.
	RouterNotInitialized	1285	Router: Router is not initialized. Error code: 1285 (0x505). Category: Router C++: ROUTERERR_NOTINITIALIZED The router is not initialized.
	PortAlreadyInUse	1286	Router: The desired port number is already assigned. Error code: 1286 (0x506). Category: Router C++: ROUTERERR_PORTALREADYINUSE The port number is already assigned.

	Member name	Value	Description
	PortNotRegistered	1287	Router: Port not registered. Error code: 1287 (0x507). Category: Router C++: ROUTERERR_NOTREGISTERED The port is not registered.
	NoMoreQueues	1288	Router: The maximum number of Ports reached. Error code: 1288 (0x508). Category: Router C++: ROUTERERR_NOMOREQUEUES The maximum number of ports has been reached.
	InvalidPort	1289	Router: The port is invalid. Error code: 1289 (0x509). Category: Router C++: ROUTERERR_INVALIDPORT The port is invalid.
	RouterNotActive	1290	Router: TwinCAT Router not active. Error code: 1290 (0x50A). Category: Router C++: ROUTERERR_NOTACTIVATED The router is not active.
	RouterFragmentBoxFull	1291	Router: The mailbox has reached the maximum number for fragmented messages. Error code: 1291 (0x50B). Category: Router C++: ROUTERERR_FRAGMENTBOXFULL The mailbox has reached the maximum number for fragmented messages.
	RouterFragmentTimeout	1292	Router: A fragment timeout has occurred. Error code: 1292 (0x50C). Category: Router C++: ROUTERERR_FRAGMENTTIMEOUT A fragment timeout has occurred.
	RouterPortToBeRemoved	1293	Router: The port is removed.. Error code: 1293 (0x50D). Category: Router C++: ROUTERERR_TOBEREMOVED

	Member name	Value	Description
			The port is removed..
	DeviceError	1792	error class <device error"> Error code: 1792 (0x700). Category: Device C++: ADSERR_DEVICE_ERROR
	DeviceServiceNotSupported	1793	The requested Service is not supported by the server. Error code: 1793 (0x701). Category: Device C++: ADSERR_DEVICE_SRVNOTSUPP This will be returned for example when an not known IndexGroup or IndexOffset is accessed.
	DeviceInvalidGroup	1794	Invalid index group. Error code: 1794 (0x702). Category: Device C++: ADSERR_DEVICE_INVALIDGRP
	DeviceInvalidOffset	1795	Invalid index offset. Error code: 1795 (0x703). Category: Device C++: ADSERR_DEVICE_INVALIDOFFSE T
	DeviceInvalidAccess	1796	Reading/writing not permitted. Error code: 1796 (0x704). Category: Device C++: ADSERR_DEVICE_INVALIDACCES S
	DeviceInvalidSize	1797	Parameter size not correct. Error code: 1797 (0x705). Category: Device C++: ADSERR_DEVICE_INVALIDSIZE Usually the size of the requested return data (by Read or ReadWrite-request) doesn't have the correct size and doesn't fit in the reserved read memory buffer.
	DeviceInvalidData	1798	Invalid parameter value(s). Error code: 1798 (0x706). Category: Device C++: ADSERR_DEVICE_INVALIDDATA
	DeviceNotReady	1799	Device is not in a ready state. Error code: 1799 (0x707). Category: Device

	Member name	Value	Description
			C++: ADSERR_DEVICE_NOTREADY
	DeviceBusy	1800	Device is busy. Error code: 1800 (0x708). Category: Device C++: ADSERR_DEVICE_BUSY This return code indicates that the addressed (ADS) device is actually busy. Try the request later on again.
	DeviceInvalidContext	1801	Invalid context (must be in Windows). Error code: 1801 (0x709). Category: Device C++: ADSERR_DEVICE_INVALIDCONTE XT
	DeviceNoMemory	1802	The device ran out of memory. Error code: 1802 (0x70a). Category: Device C++: ADSERR_DEVICE_NOMEMORY
	DeviceInvalidParam	1803	Invalid parameter value(s). Error code: 1803 (0x70b). Category: Device C++: ADSERR_DEVICE_INVALIDPARM
	DeviceNotFound	1804	Not found(files, ...). Error code: 1804 (0x70c). Category: Device C++: ADSERR_DEVICE_NOTFOUND
	DeviceSyntaxError	1805	Syntax error in command or file. Error code: 1805 (0x70d). Category: Device C++: ADSERR_DEVICE_SYNTAX
	DeviceIncompatible	1806	Objects do not match. Error code: 1806 (0x70e). Category: Device C++: ADSERR_DEVICE_INCOMPATIBLE
	DeviceExists	1807	Object already exists on device. Error code: 1807 (0x70f). Category: Device C++: ADSERR_DEVICE_EXISTS
	DeviceSymbolNotFound	1808	Symbol not found. Error code: 1808 (0x7010). Category: Device C++: ADSERR_DEVICE_SYMBOLNOTF OUND

	Member name	Value	Description
			The requested Symbol - usually requested by instance path - is not found or not existent.
	DeviceSymbolVersionInvalid	1809	Symbol version is invalid. Error code: 1809 (0x711). Category: Device C++: ADSERR_DEVICE_SYMBOLVERSIONINVALID This ErrorCode can indicate that a variable handle is invalid.
	DeviceInvalidState	1810	Server is not in a valid state. Error code: 1810 (0x712). Category: Device C++: ADSERR_DEVICE_INVALIDSTATE
	DeviceTransModeNotSupported	1811	ADS transmode is not supported. Error code: 1811 (0x713). Category: Device C++: ADSERR_DEVICE_TRANSMODENOTSUPP
	DeviceNotifyHandleInvalid	1812	Notification handle is invalid. Error code: 1812 (0x714). Category: Device C++: ADSERR_DEVICE_NOTIFYHANDINVALID
	DeviceClientUnknown	1813	Notification client not registered. Error code: 1813 (0x715). Category: Device C++: ADSERR_DEVICE_CLIENTUNKNOWN
	DeviceNoMoreHandles	1814	No more notification handles. Error code: 1814 (0x716). Category: Device C++: ADSERR_DEVICE_NOMOREHDLS
	DeviceInvalidWatchsize	1815	Size for watch to big. Error code: 1815 (0x717). Category: Device C++: ADSERR_DEVICE_INVALIDWATCHSIZE
	DeviceNotInitialized	1816	Device is not initialized. Error code: 1816 (0x718). Category: Device C++: ADSERR_DEVICE_NOTINIT
	DeviceTimeOut	1817	Device has a timeout.

	Member name	Value	Description
			Error code: 1817 (0x719). Category: Device C++: ADSERR_DEVICE_TIMEOUT
	DeviceNoInterface	1818	Query interface has failed. Error code: 1818 (0x71A). Category: Device C++: ADSERR_DEVICE_NOINTERFACE
	DeviceInvalidInterface	1819	Wrong interface required. Error code: 1819 (0x71B). Category: Device C++: ADSERR_DEVICE_INVALIDINTERF ACE
	DeviceInvalidCLSID	1820	Class ID is invalid. Error code: 1820 (0x71C). Category: Device C++: ADSERR_DEVICE_INVALIDCLSID
	DeviceInvalidObjectID	1821	Object ID is invalid. Category: Device Error code: 1821 (0x71D). C++: ADSERR_DEVICE_INVALIDOBJID
	DeviceRequestIsPending	1822	Device: A Request is Pending. Error code: 1822 (0x71E). Category: Device C++: ADSERR_DEVICE_PENDING This return code indicates that the result of an request is not available yet. Try later to access the request result.
	DeviceRequestIsAborted	1823	Device: Request is Aborted. Category: Device Error code: 1823 (0x71F). C++: ADSERR_DEVICE_ABORTED
	DeviceSignalWarning	1824	Device: Signal warning. Error code: 1824 (0x720). Category: Device C++: ADSERR_DEVICE_WARNING
	DeviceInvalidArrayIndex	1825	Device: Invalid Array Index (ADSERR_DEVICE_INVALIDARRAYIDX) Error code: 1825 (0x721). Category: Device C++: ADSERR_DEVICE_INVALIDARRAY IDX
	DeviceSymbolNotActive	1826	Device: Symbol not Active

	Member name	Value	Description
			Error code: 1826 (0x722). Category: Device C++: ADSERR_DEVICE_SYMBOLNOTACTIVE
	DeviceAccessDenied	1827	Device: Access denied. Error code: 1827 (0x723). Category: Device C++: ADSERR_DEVICE_ACCESSDENIED
	DeviceLicenseNotFound	1828	Device: license not found Error code: 1828 (0x724). Category: Device C++: ADSERR_DEVICE_LICENSENOTFOUND
	DeviceLicenseExpired	1829	Device: license expired Error code: 1829 (0x725). Category: Device C++: ADSERR_DEVICE_LICENSEEXPIRED
	DeviceLicenseExceeded	1830	Device: license exceeded Error code: 1830 (0x726). Category: Device C++: ADSERR_DEVICE_LICENSEEXCEEDED
	DeviceLicenseInvalid	1831	Device: license invalid Error code: 1831 (0x727). Category: Device C++: ADSERR_DEVICE_LICENSEINVALID
	DeviceLicenseSystemId	1832	Device: license invalid system id Error code: 1832 (0x728). Category: Device C++: ADSERR_DEVICE_LICENSESYSTEMID
	DeviceLicenseNoTimeLimit	1833	Device: license not time limited Error code: 1833 (0x729). Category: Device C++: ADSERR_DEVICE_LICENSENOTIMELIMIT
	DeviceLicenseFutureIssue	1834	Device: license issue time in the future Error code: 1834 (0x72A).

	Member name	Value	Description
			Category: Device C++: ADSERR_DEVICE_LICENSEFUTUREISSUE
	DeviceLicenseTimeToLong	1835	Device: license time period to long Error code: 1835 (0x72B). Category: Device C++: ADSERR_DEVICE_LICENSETIMETO LONG
	DeviceException	1836	Device: Exception in device specific code Error code: 1836 (0x72C). Category: Device C++: ADSERR_DEVICE_EXCEPTION
	DeviceLicenseDuplicated	1837	Device: License file read twice Error code: 1837 (0x72D). Category: Device C++: ADSERR_DEVICE_LICENSEDUPLICATED
	DeviceSignatureInvalid	1838	Device: invalid signature Error code: 1838 (0x72E). Category: Device C++: ADSERR_DEVICE_SIGNATUREINVALID
	DeviceCertificateInvalid	1839	Device: public key certificate Error code: 1839 (0x72F). Category: Device C++: ADSERR_DEVICE_CERTIFICATEINVALID
	DeviceLicenseOemNotFound	1840	Device: public key of OEM unknown Error code: 1840 (0x730). Category: Device C++: ADSERR_DEVICE_LICENSEOEMNOTFOUND
	DeviceLicenseRestricted	1841	Device: license not valid for this system id type Error code: 1841 (0x731). Category: Device C++: ADSERR_DEVICE_LICENSERESTRICTED
	DeviceLicenseDemoDenied	1842	Device: trial license denied Error code: 1842 (0x732). Category: Device

	Member name	Value	Description
			C++: ADSERR_DEVICE_LICENSEDEMO DENIED
	DeviceInvalidFunctionId	1843	Device: function id is invalid Error code: 1843 (0x733). Category: Device C++: ADSERR_DEVICE_INVALIDFNCID
	DeviceOutOfRange	1844	Device: a parameter, an index, an iterator, ... is out of range Error code: 1844 (0x734). Category: Device C++: ADSERR_DEVICE_OUTOFRANGE
	DeviceInvalidAlignment	1845	Device: invalid alignment Error code: 1845 (0x735). Category: Device C++: ADSERR_DEVICE_INVALIDALIGN MENT
	DeviceLicensePlatform	1846	Device: license invalid platform level Error code: 1846 (0x736). Category: Device C++: ADSERR_DEVICE_LICENSEPLATF ORM
	DeviceContextForwardPassive Level	1847	Device: wrong context - forward to passive level (port 12) Error code: 1847 (0x737). Category: Device C++: ADSERR_DEVICE_FORWARD_PL
	DeviceContextForwardDispatch Level	1848	Device: wrong context - forward to dispatch level (port 10) Error code: 1848 (0x738). Category: Device C++: ADSERR_DEVICE_FORWARD_DL
	DeviceContextForwardRealTim e	1849	Device: wrong context - forward to real time (port 11) Error code: 1849 (0x739). Category: Device C++: ADSERR_DEVICE_FORWARD_RT
	DeviceCertificateEntrust	1850	Device: OEM certificate not in trusted list Error code: 1850 (0x73A). Category: Device

	Member name	Value	Description
			C++: ADSERR_DEVICE_CERTIFICATEN TRUST
	ClientError	1856	Error class <client error> Error code: 1856 (0x740). Category: Client C++: ADSERR_CLIENT_ERROR
	ClientInvalidParameter	1857	Parameter at service is invalid. Error code: 1857 (0x741). Category: Client C++: ADSERR_CLIENT_INVALIDPARM
	ClientListEmpty	1858	Polling list is empty. Error code: 1858 (0x742). Category: Client C++: ADSERR_CLIENT_LISTEMPTY
	ClientVariableInUse	1859	Variable connection is already in use. Error code: 1859 (0x743). Category: Client C++: ADSERR_CLIENT_VARUSED
	ClientDuplicateInvokeID	1860	Invoke ID already in use. Error code: 1860 (0x744). Category: Client C++: ADSERR_CLIENT_DUPLINVOKEID
	ClientSyncTimeOut	1861	Timeout has elapsed. Error code: 1861 (0x745). Category: Client C++: ADSERR_CLIENT_SYNCTIMEOUT This error occurs when the communication endpoint don't answer within the configured timeout timespan. The most likely reasons for this return code is a not available client (system shutdown), network problems or a busy device that runs the client.
	ClientW32Error	1862	Error in win32 subsystem. Category: Client Error code: 1862 (0x746). C++: ADSERR_CLIENT_W32ERROR
	ClientTimeoutInvalid	1863	Timeout value is invalid. Error code: 1863 (0x747). Category: Client C++: ADSERR_CLIENT_TIMEOUTINVALI D

	Member name	Value	Description
	ClientPortNotOpen	1864	ADS port is not opened. Error code: 1864 (0x748). Category: Client C++: ADSERR_CLIENT_PORTNOTOPEN
	ClientNoAmsAddr	1865	No AMS Address. Error code: 1865 (0x749). Category: Client C++: ADSERR_CLIENT_NOAMSADDR
	ClientSyncInternal	1872	An internal in ADS sync has occurred. Error code: 1872 (0x750). Category: Client C++: ADSERR_CLIENT_SYNCINTERNAL
	ClientAddHash	1873	Hash table overflow. Error code: 1873(0x751). Category: Client C++: ADSERR_CLIENT_ADDHASH
	ClientRemoveHash	1874	There are no more symbols in the hash table. Error code: 1874 (0x752). Category: Client C++: ADSERR_CLIENT_REMOVEHASH
	ClientNoMoreSymbols	1875	There are no more symbols in cache. Error code: 1875 (0x753). Category: Client C++: ADSERR_CLIENT_NOMORESYP
	ClientResponseInvalid	1876	An invalid response has been received. Error code: 1876 (0x754). Category: Client C++ ADSERR_CLIENT_SYNCRESINVAL ID This error occurs when the client receives invalid response frame for an ADS request. The frame is ignored and this error code is returned to the caller.
	ClientPortLocked	1877	The synchronous ADS Port is locked. Error code: 1877 (0x755). Category: Client C++: ADSERR_CLIENT_SYNCPORTLOCKED

	Member name	Value	Description
	ClientRequestCancelled	1878	<p>The client Request was cancelled (asynchronously)</p> <p>Error code: 1878 (0x756).</p> <p>Category: Client</p> <p>C++: ADSERR_CLIENT_REQCANCEL</p>
	ClientQueueFull	32768	<p>The ADS Client queue is full</p> <p>Error code: 32768 (0x8000).</p> <p>Category: Client</p> <p>C++: ADSERR_CLIENT_SYNCPORTLOCKED</p>
	WSA_ConnAborted	10053	<p>Software caused connection abort (0x2745, 10053)</p> <p>An established connection was aborted by the software in your host computer, possibly due to a data transmission time-out or protocol error.</p> <p>Error code: 10053 (0x2745).</p> <p>Category: Windows Sockets</p> <p>C++: WSAECONNABORTED</p>
	WSA_ConnReset	10054	<p>Connection reset by peer. (0x2746, 10054)</p> <p>An existing connection was forcibly closed by the remote host. This normally results if the peer application on the remote host is suddenly stopped, the host is rebooted, the host or remote network interface is disabled, or the remote host uses a hard close (see setsockopt for more information on the SO_LINGER option on the remote socket). This error may also result if a connection was broken due to keep-alive activity detecting a failure while one or more operations are in progress. Operations that were in progress fail with WSAENETRESET. Subsequent operations fail with WSAECONNRESET.</p> <p>Error code: 10054 (0x2746).</p> <p>Category: Windows Sockets</p> <p>C++: WSAECONNRESET</p>
	WSA_ConnRefused	10061	<p>Windows sockets connection refused (0x274d, 10061)</p> <p>No connection could be made because the target computer actively refused it. This usually results from trying to connect to a service that is inactive on the foreign host—that is, one with no server application running.</p>

	Member name	Value	Description
			Error code: 10061 (0x274d). Category: Windows Sockets C++: WSAECONNREFUSED

Remarks

These ADS Error codes are used within the AmsHeader to indicate ReturnCodes or Error states in ADS Request Responses. For example the NoError (value 0) indicates a success. Further codes like TargetPortNotFound (the ADSServer port is not found) or ClientSyncTimeOut (the client request wasn't answered within the configured global timeout) indicate communication errors. Also ApplicationReturn codes (e.g. ClientNoMoreSymbols (the end of the Server Symbols list is reached) are encoded in this AdsErrorCode.

For more information please see the [AMS Header structure](#) described on the Beckhoff Information System.

Reference

[TwinCAT.Ads Namespace](#) [► 179]

6.2.9 AdsErrorCodeExtensions Class

Class AdsErrorCodeExtensions.

Inheritance Hierarchy

[System.Object](#)

TwinCAT.Ads.AdsErrorCodeExtensions

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax

C#

```
public static class AdsErrorCodeExtensions
```

The AdsErrorCodeExtensions type exposes the following members.

Methods

	Name	Description
	Failed [► 681]	Indicates, that the communication failed with AdsErrorCode [► 664].
	Succeeded [► 681]	Indicates, that the communication / AdsErrorCode [► 664] doesn't show an error.

Remarks

This class extends the [AdsErrorCode](#) [► 664] by [Succeeded\(AdsErrorCode\)](#) [► 681] and [Failed\(AdsErrorCode\)](#) [► 681] methods.





Reference

[TwinCAT.Ads Namespace](#) [► 179]

6.2.9.1 AdsErrorCodeExtensions Methods

The [AdsErrorCodeExtensions](#) [▶ 680] type exposes the following members.

Methods

	Name	Description
	Failed [▶ 681]	Indicates, that the communication failed with AdsErrorCode [▶ 664].
		
	Succeeded [▶ 681]	Indicates, that the communication / AdsErrorCode [▶ 664] doesn't show an error.
		

Reference

[AdsErrorCodeExtensions Class](#) [▶ 680]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.9.1.1 AdsErrorCodeExtensions.Failed Method

Indicates, that the communication failed with [AdsErrorCode](#) [▶ 664].

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool Failed(
    this AdsErrorCode errorCode
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
-----------	---

Return Value

Type: [Boolean](#)

true if the [AdsErrorCode](#) [▶ 664] indicates an Error/Failure, false otherwise.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [AdsErrorCode](#) [▶ 664]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AdsErrorCodeExtensions Class](#) [▶ 680]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.9.1.2 AdsErrorCodeExtensions.Succeeded Method

Indicates, that the communication / [AdsErrorCode](#) [▶ 664] doesn't show an error.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool Succeeded(
    this AdsErrorCode errorCode
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
-----------	---

Return Value

Type: [Boolean](#)

true if the [AdsErrorCode](#) [[▶ 664](#)] is [NoError](#) [[▶ 664](#)], false otherwise.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [AdsErrorCode](#) [[▶ 664](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AdsErrorCodeExtensions Class](#) [[▶ 680](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.10 AdsErrorException Class

The exception that is thrown when an ADS error occurs.

Inheritance Hierarchy

[System.Object](#)

[System.Exception](#)

[TwinCAT.AdsException](#) [[▶ 61](#)]

[TwinCAT.Ads.AdsErrorException](#)

[TwinCAT.Ads.AdsSumCommandException](#) [[▶ 735](#)]

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229


Syntax



C#

```
[SerializableAttribute]
public class AdsErrorException : AdsException
```










The AdsErrorException type exposes the following members.

Constructors










	Name	Description
	AdsErrorException [▶ 684]	Initializes a new Instance of the AdsErrorException class.




	Name	Description
	AdsErrorException(SerializationInfo, StreamingContext) [▶ 685]	Initializes a new instance of the AdsErrorException class.
	AdsErrorException(String, AdsErrorCode) [▶ 685]	Initializes a new Instance of the AdsErrorException class.

Properties


	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	ErrorCode [▶ 686]	Gets the error code of the Exception.
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Methods

	Name	Description
 	Create(AdsErrorCode) [▶ 688]	Creates the AdsErrorException
 	Create(String, AdsErrorCode) [▶ 688]	Creates the AdsErrorException
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData [▶ 689]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Overrides Exception.GetObjectData(SerializationInfo, StreamingContext) .)

	Name	Description
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Events




	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

Reference

[TwinCAT.Ads Namespace](#) [► 179]

6.2.10.1 AdsErrorException Constructor

Overload List

	Name	Description
	AdsErrorException [► 684]	Initializes a new Instance of the AdsErrorException class.
	AdsErrorException(SerializationInfo, StreamingContext) [► 685]	Initializes a new instance of the AdsErrorException [► 682] class.
	AdsErrorException(String, AdsErrorCode) [► 685]	Initializes a new Instance of the AdsErrorException class.

Reference

[AdsErrorException Class](#) [► 682]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.10.1.1 AdsErrorException Constructor

Initializes a new Instance of the [AdsErrorException](#) class.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorException()
```

Reference[AdsErrorException Class \[▶ 682\]](#)[AdsErrorException Overload \[▶ 684\]](#)[TwinCAT.Ads Namespace \[▶ 179\]](#)**6.2.10.1.2 AdsErrorException Constructor (SerializationInfo, StreamingContext)**

Initializes a new instance of the [AdsErrorException \[▶ 682\]](#) class.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
protected AdsErrorException(
    SerializationInfo info,
    StreamingContext streamingContext
)
```

Parameters

info	Type: System.Runtime.Serialization.SerializationInfo The information.
streamingContext	Type: System.Runtime.Serialization.StreamingContext The streaming context.

Reference[AdsErrorException Class \[▶ 682\]](#)[AdsErrorException Overload \[▶ 684\]](#)[TwinCAT.Ads Namespace \[▶ 179\]](#)**6.2.10.1.3 AdsErrorException Constructor (String, AdsErrorCode)**

Initializes a new Instance of the AdsErrorException class.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public AdsErrorException(
    string? message,
    AdsErrorCode errorCode
)
```

Parameters

message	Type: System.String The message.
errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.

Reference

[AdsErrorException Class \[▶ 682\]](#)










[AdsErrorException Overload \[▶ 684\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.10.2 AdsErrorException Properties

The [AdsErrorException \[▶ 682\]](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	ErrorCode [▶ 686]	Gets the error code of the Exception.
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Reference

[AdsErrorException Class \[▶ 682\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.10.2.1 AdsErrorException.ErrorCode Property

Gets the error code of the Exception.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode ErrorCode { get; }
```

Property Value

Type: [AdsErrorCode \[▶ 664\]](#)

The error code.

Reference














[AdsErrorException Class \[▶ 682\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.10.3 AdsErrorException Methods

The [AdsErrorException \[▶ 682\]](#) type exposes the following members.

Methods

	Name	Description
 	Create(AdsErrorCod e) [▶ 688]	Creates the AdsErrorException
 	Create(String, AdsErrorCode) [▶ 688]	Creates the AdsErrorException
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData [▶ 689]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Overrides Exception.GetObjectData(SerializationInfo, StreamingContext) .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
 	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)





Reference

[AdsErrorException Class \[▶ 682\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.10.3.1 AdsErrorException.Create Method

Overload List

	Name	Description
 	Create(AdsErrorCod e) [▶ 688]	Creates the AdsErrorException
 	Create(String, AdsErrorCode) [▶ 688]	Creates the AdsErrorException

Reference[AdsErrorException Class](#) [▶ 682][TwinCAT.Ads Namespace](#) [▶ 179]**6.2.10.3.1.1 AdsErrorException.Create Method (AdsErrorCode)**

Creates the AdsErrorException

Namespace: [TwinCAT.Ads](#) [▶ 179]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static AdsErrorException Create(
    AdsErrorCode adsErrorCode
)
```

Parameters

adsErrorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The ads error code.
--------------	---

Return ValueType: [AdsErrorException](#) [▶ 682]

AdsErrorException.

Reference[AdsErrorException Class](#) [▶ 682][Create Overload](#) [▶ 687][TwinCAT.Ads Namespace](#) [▶ 179]**6.2.10.3.1.2 AdsErrorException.Create Method (String, AdsErrorCode)**

Creates the AdsErrorException

Namespace: [TwinCAT.Ads](#) [▶ 179]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static AdsErrorException Create(
    string message,
    AdsErrorCode adsErrorCode
)
```

Parameters

message	Type: System.String The message.
adsErrorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The ads error code.

Return Value

Type: [AdsErrorException](#) [▶ 682]
 AdsErrorException.

Reference

[AdsErrorException Class](#) [▶ 682]

[Create Overload](#) [▶ 687]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.10.3 AdsErrorException.GetObjectData Method

When overridden in a derived class, sets the [SerializationInfo](#) with information about the exception.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public override void GetObjectData(
    SerializationInfo info,
    StreamingContext context
)
```

Parameters

info	Type: System.Runtime.Serialization.SerializationInfo The SerializationInfo that holds the serialized object data about the exception being thrown.
context	Type: System.Runtime.Serialization.StreamingContext The StreamingContext that contains contextual information about the source or destination.

Implements

[ISerializable.GetObjectData\(SerializationInfo, StreamingContext\)](#)

Exceptions

Exception	Condition
ArgumentNullException	info

Reference


[AdsErrorException Class](#) [▶ 682]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.10.4 AdsErrorException Events

The [AdsErrorException](#) [▶ 682] type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

Reference

[AdsErrorException Class](#) [► 682]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.11 AdsInvalidNotificationException Class

This `AdsInvalidNotificationException` is created if the length of the notification data is 0. This indicates that the notification handle is not valid any more. This exception is passed to the `AdsNotificationErrorEvent`.

Inheritance Hierarchy

[System.Object](#)

[System.Exception](#)

[TwinCAT.AdsException](#) [► 61]

`TwinCAT.Ads.AdsInvalidNotificationException`

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: `TwinCAT.Ads` (in `TwinCAT.Ads.dll`) Version: 6.0.328+39e3229










Syntax


C#

```
[SerializableAttribute]
public sealed class AdsInvalidNotificationException : AdsException
```







The `AdsInvalidNotificationException` type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	Handle [► 692]	Handle of the notification.
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

	Name	Description
	TimeStamp [▶ 692]	Gets the Time stamp as long

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData [▶ 693]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Overrides Exception.GetObjectData(SerializationInfo, StreamingContext) .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)











Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.11.1 AdsInvalidNotificationException Properties

The [AdsInvalidNotificationException](#) [[▶ 690](#)] type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	Handle [▶ 692]	Handle of the notification.
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)
	TimeStamp [▶ 692]	Gets the Time stamp as long

Reference

[AdsInvalidNotificationException Class](#) [[▶ 690](#)]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.11.1.1 **AdsInvalidNotificationException.Handle Property**

Handle of the notification.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint Handle { get; }
```

Property Value

Type: [UInt32](#)

The handle.

Reference

[AdsInvalidNotificationException Class](#) [▶ 690]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.11.1.2 **AdsInvalidNotificationException.TimeStamp Property**

Gets the Time stamp as long

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTimeOffset TimeStamp { get; }
```

Property Value

Type: [DateTimeOffset](#)

The time stamp.

Reference


[AdsInvalidNotificationException Class](#) [▶ 690]






[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.11.2 **AdsInvalidNotificationException Methods**

The [AdsInvalidNotificationException](#) [▶ 690] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)

	Name	Description
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData [▶ 693]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Overrides Exception.GetObjectData(SerializationInfo, StreamingContext) .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Reference

[AdsInvalidNotificationException Class](#) [▶ 690]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.11.2.1 AdsInvalidNotificationException.GetObjectData Method

When overridden in a derived class, sets the [SerializationInfo](#) with information about the exception.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override void GetObjectData(
    SerializationInfo info,
    StreamingContext context
)
```

Parameters

info	Type: System.Runtime.Serialization.SerializationInfo The SerializationInfo that holds the serialized object data about the exception being thrown.
context	Type: System.Runtime.Serialization.StreamingContext The StreamingContext that contains contextual information about the source or destination.

Implements

[ISerializable.GetObjectData\(SerializationInfo, StreamingContext\)](#)

Exceptions

Exception	Condition
ArgumentNullException	info

Reference

[AdsInvalidNotificationException Class](#) [▶ 690]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.12 AdsNotificationEventArgs Class

Arguments for the [AdsNotificationError](#) [▸ 993] events.

Inheritance Hierarchy

[System.Object](#)

[System.EventArgs](#)

[TwinCAT.Ads.AdsNotificationEventArgs](#)

Namespace: [TwinCAT.Ads](#) [▸ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#





```
public sealed class AdsNotificationEventArgs : EventArgs
```

The [AdsNotificationEventArgs](#) type exposes the following members.

Properties

	Name	Description
	Exception [▸ 694]	Exception that was caught while handling notifications.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)


Reference

[TwinCAT.Ads Namespace](#) [▸ 179]

6.2.12.1 AdsNotificationEventArgs Properties

The [AdsNotificationEventArgs](#) [▸ 694] type exposes the following members.

Properties

	Name	Description
	Exception [▸ 694]	Exception that was caught while handling notifications.

Reference

[AdsNotificationEventArgs Class](#) [▸ 694]

[TwinCAT.Ads Namespace](#) [▸ 179]

6.2.12.1.1 AdsNotificationEventArgs.Exception Property

Exception that was caught while handling notifications.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Exception Exception { get; }
```

Property Value

Type: [Exception](#)

Reference





[AdsNotificationEventArgs Class](#) [► 694]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.12.2 AdsNotificationEventArgs Methods

The [AdsNotificationEventArgs](#) [► 694] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[AdsNotificationEventArgs Class](#) [► 694]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.13 AdsNotificationEventArgs Class

Event argument class for [AdsNotification](#) [► 991] events.

Inheritance Hierarchy

[System.Object](#)

[System.EventArgs](#)

[TwinCAT.Ads.AdsNotificationEventArgs](#)

[TwinCAT.Ads.AdsNotificationEventArgs](#) [► 699]

[TwinCAT.Ads.ValueNotificationEventArgs.T.](#) [► 1213]

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229





Syntax

C#







```
public class AdsNotificationEventArgs : EventArgs,
    INotification
```

The `AdsNotificationEventArgs` type exposes the following members.

Properties

	Name	Description
	Data [▶ 697]	Memory object holding the Notification Data/Value.
	Handle [▶ 697]	Gets the Notification handle.
	TimeStamp [▶ 698]	Gets the time stamp of this Notification as <code>DateTimeOffset</code> .
	UserData [▶ 698]	Gets the user object. This object is passed by to <code>AddDeviceNotification</code> and can be used to store data.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from <code>Object</code> .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from <code>Object</code> .)
	GetHashCode	Serves as the default hash function. (Inherited from <code>Object</code> .)
	GetType	Gets the <code>Type</code> of the current instance. (Inherited from <code>Object</code> .)
	MemberwiseClone	Creates a shallow copy of the current <code>Object</code> . (Inherited from <code>Object</code> .)
	ToString	Returns a string that represents the current object. (Inherited from <code>Object</code> .)





Reference

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.13.1 AdsNotificationEventArgs Properties

The `AdsNotificationEventArgs` [▶ 695] type exposes the following members.

Properties

	Name	Description
	Data [▶ 697]	Memory object holding the Notification Data/Value.
	Handle [▶ 697]	Gets the Notification handle.
	TimeStamp [▶ 698]	Gets the time stamp of this Notification as <code>DateTimeOffset</code> .
	UserData [▶ 698]	Gets the user object. This object is passed by to <code>AddDeviceNotification</code> and can be used to store data.

Reference

[AdsNotificationEventArgs Class \[▶ 695\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.13.1.1 AdsNotificationEventArgs.Data Property

Memory object holding the Notification Data/Value.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyMemory<byte> Data { get; }
```

Property Value

Type: [ReadOnlyMemory.Byte](#).

Implements

[INotification.Data](#) [[▶ 1097](#)]

Remarks

This Memory object can be seen as binary 'View' to the value object. It represents exactly the data that corresponds to the [Handle](#) [[▶ 697](#)].

Reference

[AdsNotificationEventArgs Class](#) [[▶ 695](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.13.1.2 AdsNotificationEventArgs.Handle Property

Gets the Notification handle.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint Handle { get; }
```

Property Value

Type: [UInt32](#)

Implements

[INotification.Handle](#) [[▶ 1097](#)]

Remarks


The Notification Handle is the handle that is created during ADS Notification registration ([AddDeviceNotificationAsync Overload](#) [[▶ 973](#)], [AddDeviceNotification Overload](#) [[▶ 971](#)]) and used for deregistration ([DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [[▶ 984](#)], [DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [[▶ 984](#)]).

Reference

[AdsNotificationEventArgs Class \[► 695\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

Also see about this

 [IAdsNotifications.DeleteDeviceNotification Method \[► 983\]](#)

6.2.13.1.3 AdsNotificationEventArgs.TimeStamp Property

Gets the time stamp of this Notification as [DateTimeOffset](#).

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTimeOffset TimeStamp { get; }
```

Property Value

Type: [DateTimeOffset](#)

Implements

[INotification.TimeStamp \[► 1097\]](#)

Reference

[AdsNotificationEventArgs Class \[► 695\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.13.1.4 AdsNotificationEventArgs.UserData Property

Gets the user object. This object is passed by to AddDeviceNotification and can be used to store data.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object? UserData { get; }
```

Property Value

Type: [Object](#)

Implements

[INotification.UserData \[► 1098\]](#)

Reference







[AdsNotificationEventArgs Class \[► 695\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.13.2 AdsNotificationEventArgs Methods

The [AdsNotificationEventArgs \[▶ 695\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[AdsNotificationEventArgs Class \[▶ 695\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.14 AdsNotificationExEventArgs Class

Arguments for [AdsNotificationEx \[▶ 993\]](#) events.

Inheritance Hierarchy

[System.Object](#)
[System.EventArgs](#)
[TwinCAT.Ads.AdsNotificationEventArgs \[▶ 695\]](#)
[TwinCAT.Ads.AdsNotificationExEventArgs](#)

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229




Syntax



C#

```
public sealed class AdsNotificationExEventArgs : AdsNotificationEventArgs
```





The [AdsNotificationExEventArgs](#) type exposes the following members.

Properties

	Name	Description
	Data [▶ 697]	Memory object holding the Notification Data/Value. (Inherited from AdsNotificationEventArgs [▶ 695] .)
	Handle [▶ 697]	Gets the Notification handle. (Inherited from AdsNotificationEventArgs [▶ 695] .)
	TimeStamp [▶ 698]	Gets the time stamp of this Notification as DateTimeOffset . (Inherited from AdsNotificationEventArgs [▶ 695] .)

	Name	Description
	UserData [▶ 698]	Gets the user object. This object is passed by to AddDeviceNotification and can be used to store data. (Inherited from AdsNotificationEventArgs [▶ 695].)
	Value [▶ 700]	Value of the ADS Notification.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)






Reference

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.14.1 AdsNotificationExEventArgs Properties

The [AdsNotificationExEventArgs \[▶ 699\]](#) type exposes the following members.

Properties

	Name	Description
	Data [▶ 697]	Memory object holding the Notification Data/Value. (Inherited from AdsNotificationEventArgs [▶ 695].)
	Handle [▶ 697]	Gets the Notification handle. (Inherited from AdsNotificationEventArgs [▶ 695].)
	TimeStamp [▶ 698]	Gets the time stamp of this Notification as DateTimeOffset . (Inherited from AdsNotificationEventArgs [▶ 695].)
	UserData [▶ 698]	Gets the user object. This object is passed by to AddDeviceNotification and can be used to store data. (Inherited from AdsNotificationEventArgs [▶ 695].)
	Value [▶ 700]	Value of the ADS Notification.

Reference

[AdsNotificationExEventArgs Class \[▶ 699\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.14.1.1 AdsNotificationExEventArgs.Value Property

Value of the ADS Notification.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object Value { get; }
```

Property Value

Type: [Object](#)

Reference





[AdsNotificationExEventArgs Class \[▶ 699\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.14.2 AdsNotificationExEventArgs Methods

The [AdsNotificationExEventArgs \[▶ 699\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[AdsNotificationExEventArgs Class \[▶ 699\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.15 AdsSession Class

AdsSession class

Inheritance Hierarchy

- [System.Object](#)
- [TwinCAT.Session \[▶ 107\]](#)
- [TwinCAT.Ads.AdsSessionBase \[▶ 716\]](#)
- TwinCAT.Ads.AdsSession

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229











Syntax

C#


```
public class AdsSession : AdsSessionBase
```















The AdsSession type exposes the following members.

Constructors










	Name	Description
	AdsSession(AmsAddress) [▶ 708]	Initializes a new instance of the AdsSession class.
	AdsSession(AmsAddress, ILogger) [▶ 711]	Initializes a new instance of the AdsSession class.
	AdsSession(AmsAddress, SessionSettings) [▶ 709]	Initializes a new instance of the AdsSession class.
	AdsSession(AmsNetId, Int32) [▶ 709]	Initializes a new instance of the AdsSession class.
	AdsSession(AmsAddress, SessionSettings, ILogger) [▶ 711]	Initializes a new instance of the AdsSession class.
	AdsSession(AmsAddress, SessionSettings, Object) [▶ 710]	Initializes a new instance of the AdsSession class.
	AdsSession(AmsNetId, Int32, ILogger) [▶ 712]	Initializes a new instance of the AdsSession class.
	AdsSession(AmsNetId, Int32, SessionSettings) [▶ 710]	Initializes a new instance of the AdsSession class.
	AdsSession(AmsAddress, SessionSettings, ILogger, Object) [▶ 712]	Initializes a new instance of the AdsSession class.
	AdsSession(AmsNetId, Int32, SessionSettings, ILogger) [▶ 713]	Initializes a new instance of the AdsSession class.




Properties

	Name	Description
	Address [▶ 720]	Gets the target address of the AdsSessionBase [▶ 716] (Inherited from AdsSessionBase [▶ 716].)



	Name	Description
	AddressSpecifier [▶ 109]	Gets the communication endpoint address string representation. (Inherited from Session [▶ 107].)
	Connection [▶ 720]	Gets the connection. (Inherited from AdsSessionBase [▶ 716].)
 	ConnectionState [▶ 110]	Gets the current Connection state of the Session [▶ 107] (Inherited from Session [▶ 107].)
	Disposed [▶ 111]	Gets a value indicating whether this Session [▶ 107] is disposed. (Inherited from Session [▶ 107].)
	EstablishedAt [▶ 112]	Gets the UTC time when the session was established. (Inherited from Session [▶ 107].)
	Id [▶ 112]	Gets the Session Identifier (Inherited from Session [▶ 107].)
	IsConnected [▶ 113]	Gets a value indicating whether this instance is connected. (Inherited from Session [▶ 107].)
	Logger [▶ 723]	Gets the logger interface or null. (Inherited from AdsSessionBase [▶ 716].)
	Name [▶ 113]	Gets the name of the session (Inherited from Session [▶ 107].)
	NetId [▶ 721]	Gets the NetId of the Session (Inherited from AdsSessionBase [▶ 716].)
	Owner [▶ 721]	Gets the Session owner. (Inherited from AdsSessionBase [▶ 716].)
	Port [▶ 722]	Gets the Ams Port of the Session (Inherited from AdsSessionBase [▶ 716].)
	Settings [▶ 722]	Gets the settings of the Session/Connection. (Inherited from AdsSessionBase [▶ 716].)
	Statistics [▶ 722]	Gets the Communication / Session statistics. (Inherited from AdsSessionBase [▶ 716].)
	SymbolServer [▶ 113]	Gets the symbol server. (Inherited from Session [▶ 107].)

Methods

	Name	Description
	Close [▶ 115]	Closes this ISession [▶ 92] (Inherited from Session [▶ 107].)
	Connect [▶ 116]	Connects the session. (Inherited from Session [▶ 107].)
	ConnectAsync [▶ 122]	Connects the session. (Inherited from Session [▶ 107].)
	Disconnect [▶ 116]	Disconnects the session from the target. (Inherited from Session [▶ 107].)
	Dispose. [▶ 117]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from Session [▶ 107].)
	Dispose(Boolean) [▶ 725]	Releases unmanaged and - optionally - managed resources. (Inherited from AdsSessionBase [▶ 716].)
	EnsureConnection [▶ 728]	Ensures, that the ISession [▶ 92] is connected and returns the IConnection [▶ 79] object. (Inherited from AdsSessionBase [▶ 716].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize [▶ 725]	Finalizes an instance of the AdsSessionBase [▶ 716] class. (Inherited from AdsSessionBase [▶ 716].)

	Name	Description
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetSessionName [▶ 725]	Gets the name/string identifier of the session. (Inherited from AdsSessionBase [▶ 716].)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnConnect [▶ 726]	Handler function connecting the Session. (Inherited from AdsSessionBase [▶ 716].)
	OnConnectAsync [▶ 123]	Handler function connecting the Session. (Inherited from Session [▶ 107].)
	OnConnectionState Changed [▶ 121]	Handles the [E:ConnectionStateChanged] event. (Inherited from Session [▶ 107].)
	OnCreateSymbolSer ver [▶ 726]	Handler function creating the symbol server object. (Inherited from AdsSessionBase [▶ 716].)
	OnDisconnect [▶ 727]	Handler function disconnecting the session. (Inherited from AdsSessionBase [▶ 716].)
	OnGetAddress [▶ 727]	Handler function getting the address of the session. (Inherited from AdsSessionBase [▶ 716].)
	ToString [▶ 121]	Returns a String that represents this instance. (Inherited from Session [▶ 107].)

Events

	Name	Description
 	ConnectionStateCha nged [▶ 124]	Occurs when connection status of the IConnectionStateProvider [▶ 89] has been changed. (Inherited from Session [▶ 107].)

Remarks

On top of the well known [AdsClient](#) [▶ 183] class that is used traditionally for ADS communication, the [AdsSession](#) class provides the following additionally abilities out of the box: These are used to provide more stable connections to ADS Servers than the [AdsClient](#) [▶ 183] can provide. The main issues are Resurrection / Self-Healing after communication timeouts, faster and less error prone reaction to communication errors (not necessarily waiting for communication timeouts) und enhanced communication diagnosis. These enhanced features are provided by the following additions to the TwinCAT.Ads API:

- [AdsConnection](#) [▶ 412] class.
- Enhanced diagnosis in form of communication statistics [Statistics](#) [▶ 722]
- (semi-automatic) Resurrectable client communication with [AdsConnection](#) [▶ 412] objects.
- Symbol caching [SymbolServer](#) [▶ 113]
- Fail fast handler for connection stabilization [IFailFastHandler](#)

The [AdsConnection](#) [▶ 412] is established by calling the [Connect](#). [▶ 116] method. The returned [AdsConnection](#) [▶ 412] can be used as long the [AdsSessionBase](#) [▶ 716] exists.

Examples

The following sample shows a simple use of the [AdsSessionBase](#) [▶ 716] object. The [AdsSession](#) object (and the dynamic [SymbolLoader](#) features) are only available from .NET 4 and upwards.

Use Session (async)

```

using System;
using System.Diagnostics;
using System.Threading;

using TwinCAT;
using TwinCAT.Ads;
using TwinCAT.Ads.TypeSystem;
using TwinCAT.TypeSystem;
using TwinCAT.TypeSystem.Generic;

namespace Sample
{
    class SessionAsync
    {
        /// <summary>
        /// Defines the entry point of the application.
        /// </summary>
        /// <param name="args">The arguments.</param>
        static async void Main(string[] args)
        {
            AmsAddress address = ArgParser.Parse(args);

            CancellationTokenSource cancelSource = new CancellationTokenSource();
            CancellationToken cancel = cancelSource.Token;
            SessionSettings settings = SessionSettings.Default; // Default settings are Async access with
            // Timeout 5 sec

            // Async access is necessary for Console applications!

            using (AdsSession session = new AdsSession(address, settings))
            {
                AdsConnection connection = (AdsConnection) session.Connect(); // Establish the connection
                connection.ConnectionStateChanged += Connection_ConnectionStateChanged;

                ConnectionState connectionState = connection.ConnectionState; // The actual connection state

                // Read the identification and version number of the device
                var result = await connection.ReadDeviceInfoAsync(cancel);

                result.ThrowOnError(); // Throws exception if failed.

                DeviceInfo deviceInfo = result.DeviceInfo;
                Version version = deviceInfo.Version.ConvertToStandard();
                Console.WriteLine(string.Format("DeviceName: {0}", deviceInfo.Name));
                Console.WriteLine(string.Format("DeviceVersion: {0}", version.ToString(3)));

                /// Read the state of the device
                var resultReadState = await connection.ReadStateAsync(cancel);
                resultReadState.ThrowOnError();

                StateInfo stateInfo = resultReadState.State;
                AdsState adsState = stateInfo.AdsState;
                short deviceState = stateInfo.DeviceState;

                Console.WriteLine(string.Format("DeviceState: {0}", deviceState));
                Console.WriteLine(string.Format("AdsState : {0}", adsState));

                // Other ADS methods (as formerly used on AdsClient) can be used also on connection object:

                // connection.ReadAsync(...)
                // connection.WriteAsync(...)
                // connection.AddDeviceNotificationEx += ...

                // Session communication Diagnostic:

                int resurrectionTries = connection.TotalResurrectingTries;
                int succeededResurrections = connection.TotalResurrections;

                AdsCommunicationStatistics statistics = session.Statistics; // The communication statistics

                // Symbol access:
                // The Session holds and Caches the Symbolic data information
                var resultDataTypes = await session.SymbolServer.GetDataTypesAsync(cancel);
                var resultSymbols = await session.SymbolServer.GetSymbolsAsync(cancel);

                if (resultDataTypes.Succeeded && resultSymbols.Succeeded) // Check for succeed
                {
                    IDataValueCollection<IDataType> types = resultDataTypes.DataTypes;

```

```

        ISymbolCollection<ISymbol> symbols = resultSymbols.Symbols;

        Symbol projectNameSymbol = (Symbol)symbols["TwinCAT_SystemInfoVarList._AppInfo.ProjectName"];

        var resultReadProjectName = await projectNameSymbol.ReadValueAsync(cancel);
        string projectName = (string)resultReadProjectName.Value;

        // Or use dynamic objects
        dynamic appInfo = symbols["TwinCAT_SystemInfoVarList._AppInfo"];
        string projectName2 = appInfo.ProjectName; // Property dynamically created (synchronous call)!
    }
    Console.WriteLine("");
    Console.WriteLine("Press [Enter] for leave:");
    Console.ReadLine();
}

private static void Connection_ConnectionStateChanged(object sender, ConnectionStateChangedEventArgs e)
{
    Console.WriteLine("Connection State changed (NewState: {0}, OldState: {1})",e.NewState,e.OldState);
}

```

Use Session (sync)

```

using System;
using System.Diagnostics;
using System.Threading;

using TwinCAT;
using TwinCAT.Ads;
using TwinCAT.Ads.TypeSystem;
using TwinCAT.TypeSystem;
using TwinCAT.TypeSystem.Generic;

namespace Sample
{
    class Session
    {
        /// <summary>
        /// Defines the entry point of the application.
        /// </summary>
        /// <param name="args">The arguments.</param>
        static void Main(string[] args)
        {
            AmsAddress address = ArgParser.Parse(args);
            SessionSettings settings = SessionSettings.Default; // Default settings are Async access with Timeout 5 sec

            // Async access is necessary for Console applications!

            using (AdsSession session = new AdsSession(address, settings))
            {
                AdsConnection connection = (AdsConnection) session.Connect(); // Establish the connection
                connection.ConnectionStateChanged += Connection_ConnectionStateChanged;

                ConnectionState connectionState = connection.ConnectionState; // The actual connection state

                // Read the identification and version number of the device
                DeviceInfo deviceInfo = connection.ReadDeviceInfo();
                Version version = deviceInfo.Version.ConvertToStandard();
                Console.WriteLine(string.Format("DeviceName: {0}", deviceInfo.Name));
                Console.WriteLine(string.Format("DeviceVersion: {0}", version.ToString(3)));

                // Read the state of the device
                StateInfo stateInfo = connection.ReadState();
                AdsState adsState = stateInfo.AdsState;

                short deviceState = stateInfo.DeviceState;
                Console.WriteLine(string.Format("DeviceState: {0}", deviceState));
                Console.WriteLine(string.Format("AdsState : {0}", adsState));

                // Other ADS methods (as formerly used on AdsClient) can be used also on connection object:

                // connection.Read(...)
                // connection.Write(...)
                // connection.AddDeviceNotificationEx += ...
            }
        }
    }
}

```

```

// Session communication Diagnostic:

int resurrectionTries = connection.TotalResurrectingTries;
int succeededResurrections = connection.TotalResurrections;

AdsCommunicationStatistics statistics = session.Statistics; // The communication statistics

// Symbol access:
// The Session holds and Caches the Symbolic data information
IDataTypeCollection<IDataType> types = session.SymbolServer.DataTypes;
ISymbolCollection<ISymbol> symbols = session.SymbolServer.Symbols;

Symbol projectNameSymbol = (Symbol)symbols["TwinCAT_SystemInfoVarList._AppInfo.ProjectName"];
;
string projectName = (string) projectNameSymbol.ReadValue();

// Or use dynamic objects
dynamic appInfo = symbols["TwinCAT_SystemInfoVarList._AppInfo"];
string projectName2 = appInfo.ProjectName;

Console.WriteLine("");
Console.WriteLine("Press [Enter] for leave:");
Console.ReadLine();
}

private static void Connection_ConnectionStateChanged(object sender, ConnectionStateChangedEventArgs e)
{
    Console.WriteLine("Connection State changed (NewState: {0}, OldState: {1})",e.NewState,e.OldState);
}

```

Reference

[TwinCAT.Ads Namespace](#) [▶ 179]





[TwinCAT.Session](#) [▶ 107]


[TwinCAT.Ads.IAdsSession](#) [▶ 1033]

[IInterceptionFactory](#)

6.2.15.1 AdsSession Constructor

Overload List

	Name	Description
	AdsSession(AmsAddress) [▶ 708]	Initializes a new instance of the AdsSession [▶ 701] class.
	AdsSession(AmsAddress, ILogger) [▶ 711]	Initializes a new instance of the AdsSession [▶ 701] class.
	AdsSession(AmsAddress, SessionSettings) [▶ 709]	Initializes a new instance of the AdsSession [▶ 701] class.
	AdsSession(AmsNetId, Int32) [▶ 709]	Initializes a new instance of the AdsSession [▶ 701] class.

	Name	Description
	AdsSession(AmsAddress, SessionSettings, ILogger) [▶ 711]	Initializes a new instance of the AdsSession [▶ 701] class.
	AdsSession(AmsAddress, SessionSettings, Object) [▶ 710]	Initializes a new instance of the AdsSession [▶ 701] class.
	AdsSession(AmsNetId, Int32, ILogger) [▶ 712]	Initializes a new instance of the AdsSession [▶ 701] class.
	AdsSession(AmsNetId, Int32, SessionSettings) [▶ 710]	Initializes a new instance of the AdsSession [▶ 701] class.
	AdsSession(AmsAddress, SessionSettings, ILogger, Object) [▶ 712]	Initializes a new instance of the AdsSession [▶ 701] class.
	AdsSession(AmsNetId, Int32, SessionSettings, ILogger) [▶ 713]	Initializes a new instance of the AdsSession [▶ 701] class.

Reference

[AdsSession Class](#) [[▶ 701](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.15.1.1 AdsSession Constructor (AmsAddress)

Initializes a new instance of the [AdsSession](#) [[▶ 701](#)] class.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsSession(
    AmsAddress address
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [▶ 752] The address.
---------	--

Reference[AdsSession Class \[► 701\]](#)[AdsSession Overload \[► 707\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.15.1.2 AdsSession Constructor (AmsAddress, SessionSettings)**Initializes a new instance of the [AdsSession \[► 701\]](#) class.**Namespace:** [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public AdsSession(
    AmsAddress address,
    SessionSettings settings
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [► 752] The address.
settings	Type: TwinCAT.Ads.SessionSettings [► 1193] The settings.

Reference[AdsSession Class \[► 701\]](#)[AdsSession Overload \[► 707\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.15.1.3 AdsSession Constructor (AmsNetId, Int32)**Initializes a new instance of the [AdsSession \[► 701\]](#) class.**Namespace:** [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public AdsSession(
    AmsNetId netId,
    int port
)
```

Parameters

netId	Type: TwinCAT.Ads.AmsNetId [► 767] The net identifier.
port	Type: System.Int32 The port.

Reference[AdsSession Class \[► 701\]](#)[AdsSession Overload \[► 707\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.15.1.4 AdsSession Constructor (AmsAddress, SessionSettings, Object)**Initializes a new instance of the [AdsSession \[► 701\]](#) class.**Namespace:** [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public AdsSession(
    AmsAddress address,
    SessionSettings settings,
    Object? owner
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [► 752] The address.
settings	Type: TwinCAT.Ads.SessionSettings [► 1193] The settings.
owner	Type: System.Object The session owner

Reference[AdsSession Class \[► 701\]](#)[AdsSession Overload \[► 707\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.15.1.5 AdsSession Constructor (AmsNetId, Int32, SessionSettings)**Initializes a new instance of the [AdsSession \[► 701\]](#) class.**Namespace:** [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public AdsSession(
    AmsNetId netId,
    int port,
    SessionSettings settings
)
```

Parameters

netId	Type: TwinCAT.Ads.AmsNetId [► 767] The net identifier.
-------	---

port	Type: System.Int32 The port.
settings	Type: TwinCAT.Ads.SessionSettings [▶ 1193] The settings.

Reference

[AdsSession Class](#) [[▶ 701](#)]

[AdsSession Overload](#) [[▶ 707](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.15.1.6 AdsSession Constructor (AmsAddress, ILogger)

Initializes a new instance of the [AdsSession](#) [[▶ 701](#)] class.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsSession(
    AmsAddress address,
    ILogger? logger
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [▶ 752] The address.
logger	Type: Microsoft.Extensions.Logging.ILogger The logger interface.

Reference

[AdsSession Class](#) [[▶ 701](#)]

[AdsSession Overload](#) [[▶ 707](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.15.1.7 AdsSession Constructor (AmsAddress, SessionSettings, ILogger)

Initializes a new instance of the [AdsSession](#) [[▶ 701](#)] class.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsSession(
    AmsAddress address,
    SessionSettings settings,
    ILogger? logger
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [▶ 752] The address.
settings	Type: TwinCAT.Ads.SessionSettings [▶ 1193] The settings.
logger	Type: Microsoft.Extensions.Logging.ILogger The logger interface.

Reference

[AdsSession Class](#) [▶ 701]

[AdsSession Overload](#) [▶ 707]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.15.1.8 AdsSession Constructor (AmsNetId, Int32, ILogger)

Initializes a new instance of the [AdsSession](#) [▶ 701] class.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public AdsSession(
    AmsNetId netId,
    int port,
    ILogger? logger
)
```

Parameters

netId	Type: TwinCAT.Ads.AmsNetId [▶ 767] The net identifier.
port	Type: System.Int32 The port.
logger	Type: Microsoft.Extensions.Logging.ILogger The logger interface.

Reference

[AdsSession Class](#) [▶ 701]

[AdsSession Overload](#) [▶ 707]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.15.1.9 AdsSession Constructor (AmsAddress, SessionSettings, ILogger, Object)

Initializes a new instance of the [AdsSession](#) [▶ 701] class.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsSession(
    AmsAddress address,
    SessionSettings settings,
    ILogger? logger,
    Object? owner
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [▶ 752] The address.
settings	Type: TwinCAT.Ads.SessionSettings [▶ 1193] The settings.
logger	Type: Microsoft.Extensions.Logging.ILogger The logger interface.
owner	Type: System.Object The session owner

Reference

[AdsSession Class](#) [▶ 701]

[AdsSession Overload](#) [▶ 707]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.15.1.10 AdsSession Constructor (AmsNetId, Int32, SessionSettings, ILogger)

Initializes a new instance of the [AdsSession](#) [▶ 701] class.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsSession(
    AmsNetId netId,
    int port,
    SessionSettings settings,
    ILogger? logger
)
```

Parameters

netId	Type: TwinCAT.Ads.AmsNetId [▶ 767] The net identifier.
port	Type: System.Int32 The port.
settings	Type: TwinCAT.Ads.SessionSettings [▶ 1193] The settings.
logger	Type: Microsoft.Extensions.Logging.ILogger The logger interface.

Reference

[AdsSession Class \[► 701\]](#)


















[AdsSession Overload \[► 707\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.15.2 AdsSession Properties

The [AdsSession \[► 701\]](#) type exposes the following members.

Properties

	Name	Description
	Address [► 720]	Gets the target address of the AdsSessionBase [► 716] (Inherited from AdsSessionBase [► 716] .)
	AddressSpecifier [► 109]	Gets the communication endpoint address string representation. (Inherited from Session [► 107] .)
	Connection [► 720]	Gets the connection. (Inherited from AdsSessionBase [► 716] .)
 	ConnectionState [► 110]	Gets the current Connection state of the Session [► 107] (Inherited from Session [► 107] .)
	Disposed [► 111]	Gets a value indicating whether this Session [► 107] is disposed. (Inherited from Session [► 107] .)
	EstablishedAt [► 112]	Gets the UTC time when the session was established. (Inherited from Session [► 107] .)
	Id [► 112]	Gets the Session Identifier (Inherited from Session [► 107] .)
	IsConnected [► 113]	Gets a value indicating whether this instance is connected. (Inherited from Session [► 107] .)
	Logger [► 723]	Gets the logger interface or null. (Inherited from AdsSessionBase [► 716] .)
	Name [► 113]	Gets the name of the session (Inherited from Session [► 107] .)
	NetId [► 721]	Gets the NetId of the Session (Inherited from AdsSessionBase [► 716] .)
	Owner [► 721]	Gets the Session owner. (Inherited from AdsSessionBase [► 716] .)
	Port [► 722]	Gets the Ams Port of the Session (Inherited from AdsSessionBase [► 716] .)
	Settings [► 722]	Gets the settings of the Session/Connection. (Inherited from AdsSessionBase [► 716] .)
	Statistics [► 722]	Gets the Communication / Session statistics. (Inherited from AdsSessionBase [► 716] .)
	SymbolServer [► 113]	Gets the symbol server. (Inherited from Session [► 107] .)

Reference

[AdsSession Class \[► 701\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.15.3 AdsSession Methods

The [AdsSession](#) [▶ 701] type exposes the following members.

Methods

	Name	Description
	Close [▶ 115]	Closes this ISession [▶ 92] (Inherited from Session [▶ 107].)
	Connect [▶ 116]	Connects the session. (Inherited from Session [▶ 107].)
	ConnectAsync [▶ 122]	Connects the session. (Inherited from Session [▶ 107].)
	Disconnect [▶ 116]	Disconnects the session from the target. (Inherited from Session [▶ 107].)
	Dispose. [▶ 117]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from Session [▶ 107].)
	Dispose(Boolean) [▶ 725]	Releases unmanaged and - optionally - managed resources. (Inherited from AdsSessionBase [▶ 716].)
	EnsureConnection [▶ 728]	Ensures, that the ISession [▶ 92] is connected and returns the IConnection [▶ 79] object. (Inherited from AdsSessionBase [▶ 716].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize [▶ 725]	Finalizes an instance of the AdsSessionBase [▶ 716] class. (Inherited from AdsSessionBase [▶ 716].)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetSessionName [▶ 725]	Gets the name/string identifier of the session. (Inherited from AdsSessionBase [▶ 716].)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnConnect [▶ 726]	Handler function connecting the Session. (Inherited from AdsSessionBase [▶ 716].)
	OnConnectAsync [▶ 123]	Handler function connecting the Session. (Inherited from Session [▶ 107].)
	OnConnectionStateChanged [▶ 121]	Handles the [E:ConnectionStateChanged] event. (Inherited from Session [▶ 107].)
	OnCreateSymbolServer [▶ 726]	Handler function creating the symbol server object. (Inherited from AdsSessionBase [▶ 716].)
	OnDisconnect [▶ 727]	Handler function disconnecting the session. (Inherited from AdsSessionBase [▶ 716].)
	OnGetAddress [▶ 727]	Handler function getting the address of the session. (Inherited from AdsSessionBase [▶ 716].)
	ToString [▶ 121]	Returns a String that represents this instance. (Inherited from Session [▶ 107].)

Reference



[AdsSession Class](#) [▶ 701]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.15.4 AdsSession Events

The [AdsSession](#) [[▶ 701](#)] type exposes the following members.

Events

	Name	Description
 	ConnectionStateChanged [▶ 124]	Occurs when connection status of the IConnectionStateProvider [▶ 89] has been changed. (Inherited from Session [▶ 107].)

Reference

[AdsSession Class](#) [[▶ 701](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.16 AdsSessionBase Class

Abstract base class for ADS Sessions.

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Session](#) [[▶ 107](#)]

[TwinCAT.Ads.AdsSessionBase](#)

[TwinCAT.Ads.AdsSession](#) [[▶ 701](#)]

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: [TwinCAT.Ads](#) (in [TwinCAT.Ads.dll](#)) Version: 6.0.328+39e3229



Syntax

C#






```
public abstract class AdsSessionBase : Session,
    IAdsSession, ISession, IConnectionStateProvider, ISymbolServerProvider
```













The [AdsSessionBase](#) type exposes the following members.

Constructors














	Name	Description
 	AdsSessionBase [▶ 718]	Initializes a new instance of the AdsSessionBase class.








Properties

	Name	Description
	Address [▶ 720]	Gets the target address of the AdsSessionBase
	AddressSpecifier [▶ 109]	Gets the communication endpoint address string representation. (Inherited from Session [▶ 107].)
	Connection [▶ 720]	Gets the connection.
 	ConnectionState [▶ 110]	Gets the current Connection state of the Session [▶ 107] (Inherited from Session [▶ 107].)



	Name	Description
	Disposed [▶ 111]	Gets a value indicating whether this Session [▶ 107] is disposed. (Inherited from Session [▶ 107].)
	EstablishedAt [▶ 112]	Gets the UTC time when the session was established. (Inherited from Session [▶ 107].)
	Id [▶ 112]	Gets the Session Identifier (Inherited from Session [▶ 107].)
	IsConnected [▶ 113]	Gets a value indicating whether this instance is connected. (Inherited from Session [▶ 107].)
	Logger [▶ 723]	Gets the logger interface or null.
	Name [▶ 113]	Gets the name of the session (Inherited from Session [▶ 107].)
	NetId [▶ 721]	Gets the NetId of the Session
	Owner [▶ 721]	Gets the Session owner.
	Port [▶ 722]	Gets the Ams Port of the Session
	Settings [▶ 722]	Gets the settings of the Session/Connection.
	Statistics [▶ 722]	Gets the Communication / Session statistics.
	SymbolServer [▶ 113]	Gets the symbol server. (Inherited from Session [▶ 107].)

Methods

	Name	Description
	Close [▶ 115]	Closes this ISession [▶ 92] (Inherited from Session [▶ 107].)
	Connect [▶ 116]	Connects the session. (Inherited from Session [▶ 107].)
	ConnectAsync [▶ 122]	Connects the session. (Inherited from Session [▶ 107].)
	Disconnect [▶ 116]	Disconnects the session from the target. (Inherited from Session [▶ 107].)
	Dispose. [▶ 117]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from Session [▶ 107].)
	Dispose(Boolean) [▶ 725]	Releases unmanaged and - optionally - managed resources. (Overrides Session.Dispose(Boolean) [▶ 118].)
	EnsureConnection [▶ 728]	Ensures, that the ISession [▶ 92] is connected and returns the IConnection [▶ 79] object.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize [▶ 725]	Finalizes an instance of the AdsSessionBase class. (Overrides Object.Finalize.)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetSessionName [▶ 725]	Gets the name/string identifier of the session. (Overrides Session.GetSessionName. [▶ 118].)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)

	Name	Description
	OnConnect [▶ 726]	Handler function connecting the Session. (Overrides Session.OnConnect(Boolean) [▶ 119].)
	OnConnectAsync [▶ 123]	Handler function connecting the Session. (Inherited from Session [▶ 107].)
	OnConnectionState Changed [▶ 121]	Handles the [E:ConnectionStateChanged] event. (Inherited from Session [▶ 107].)
	OnCreateSymbolSer ver [▶ 726]	Handler function creating the symbol server object. (Overrides Session.OnCreateSymbolServer . [▶ 119].)
	OnDisconnect [▶ 727]	Handler function disconnecting the session. (Overrides Session.OnDisconnect . [▶ 120].)
	OnGetAddress [▶ 727]	Handler function getting the address of the session. (Overrides Session.OnGetAddress . [▶ 120].)
	ToString [▶ 121]	Returns a String that represents this instance. (Inherited from Session [▶ 107].)

Events

	Name	Description
 	ConnectionStateCha nged [▶ 124]	Occurs when connection status of the IConnectionStateProvider [▶ 89] has been changed. (Inherited from Session [▶ 107].)

Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.16.1 AdsSessionBase Constructor

Initializes a new instance of the [AdsSessionBase](#) [[▶ 716](#)] class.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected AdsSessionBase(
    AmsAddress address,
    SessionSettings settings,
    IAdsClientFactory factory,
    ILogger? logger,
    Object? owner
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [▶ 752] The address.
settings	Type: TwinCAT.Ads.SessionSettings [▶ 1193] The settings.
factory	Type: IAdsClientFactory The client factory

logger	Type: Microsoft.Extensions.Logging.ILogger The logger.
owner	Type: System.Object The session owner

Exceptions

Exception	Condition
ArgumentNullException	address

Reference

















[AdsSessionBase Class \[▶ 716\]](#)


[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.16.2 AdsSessionBase Properties

The [AdsSessionBase \[▶ 716\]](#) type exposes the following members.

Properties

	Name	Description
	Address [▶ 720]	Gets the target address of the AdsSessionBase [▶ 716]
	AddressSpecifier [▶ 109]	Gets the communication endpoint address string representation. (Inherited from Session [▶ 107] .)
	Connection [▶ 720]	Gets the connection.
 	ConnectionState [▶ 110]	Gets the current Connection state of the Session [▶ 107] (Inherited from Session [▶ 107] .)
	Disposed [▶ 111]	Gets a value indicating whether this Session [▶ 107] is disposed. (Inherited from Session [▶ 107] .)
	EstablishedAt [▶ 112]	Gets the UTC time when the session was established. (Inherited from Session [▶ 107] .)
	Id [▶ 112]	Gets the Session Identifier (Inherited from Session [▶ 107] .)
	IsConnected [▶ 113]	Gets a value indicating whether this instance is connected. (Inherited from Session [▶ 107] .)
	Logger [▶ 723]	Gets the logger interface or null.
	Name [▶ 113]	Gets the name of the session (Inherited from Session [▶ 107] .)
	NetId [▶ 721]	Gets the NetId of the Session
	Owner [▶ 721]	Gets the Session owner.
	Port [▶ 722]	Gets the Ams Port of the Session
	Settings [▶ 722]	Gets the settings of the Session/Connection.
	Statistics [▶ 722]	Gets the Communication / Session statistics.

	Name	Description
	SymbolServer [▶ 113]	Gets the symbol server. (Inherited from Session [▶ 107].)

Reference

[AdsSessionBase Class](#) [▶ 716]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.16.2.1 AdsSessionBase.Address Property

Gets the target address of the [AdsSessionBase](#) [▶ 716]

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsAddress Address { get; }
```

Property Value

Type: [AmsAddress](#) [▶ 752]

The address.

Implements

[IAdsSession.Address](#) [▶ 1035]

Reference

[AdsSessionBase Class](#) [▶ 716]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.16.2.2 AdsSessionBase.Connection Property

Gets the connection.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsConnection? Connection { get; protected set; }
```

Property Value

Type: [AdsConnection](#) [▶ 412]

The connection.

Reference

[AdsSessionBase Class](#) [▶ 716]

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.16.2.3 **AdsSessionBase.NetId Property**

Gets the NetId of the Session

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsNetId NetId { get; }
```

Property Value

Type: [AmsNetId \[► 767\]](#)

The net identifier.

Implements

[IAdsSession.NetId \[► 1036\]](#)

Reference

[AdsSessionBase Class \[► 716\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.16.2.4 **AdsSessionBase.Owner Property**

Gets the Session owner.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object? Owner { get; }
```

Property Value

Type: [Object](#)

The owner or NULL

Implements

[IAdsSession.Owner \[► 1036\]](#)

Reference

[AdsSessionBase Class \[► 716\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.16.2.5 AdsSessionBase.Port Property

Gets the Ams Port of the Session

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Port { get; }
```

Property Value

Type: [Int32](#)

The port.

Implements

[IAdsSession.Port](#) [[▶ 1036](#)]

Reference

[AdsSessionBase Class](#) [[▶ 716](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.16.2.6 AdsSessionBase.Settings Property

Gets the settings of the Session/Connection.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SessionSettings Settings { get; }
```

Property Value

Type: [SessionSettings](#) [[▶ 1193](#)]

The settings.

Reference

[AdsSessionBase Class](#) [[▶ 716](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.16.2.7 AdsSessionBase.Statistics Property

Gets the Communication / Session statistics.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsCommunicationStatistics Statistics { get; }
```

Property Value

Type: [AdsCommunicationStatistics](#) [▶ 160]
 The communication / Session statistics.

Reference

[AdsSessionBase Class](#) [▶ 716]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.16.2.8 AdsSessionBase.Logger Property

Gets the logger interface or null.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ILogger? Logger { get; }
```

Property Value

Type: [ILogger](#)
 The logger.

Reference







[AdsSessionBase Class](#) [▶ 716]















[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.16.3 AdsSessionBase Methods

The [AdsSessionBase](#) [▶ 716] type exposes the following members.

Methods

	Name	Description
	Close [▶ 115]	Closes this ISession [▶ 92] (Inherited from Session [▶ 107].)
	Connect [▶ 116]	Connects the session. (Inherited from Session [▶ 107].)
	ConnectAsync [▶ 122]	Connects the session. (Inherited from Session [▶ 107].)
	Disconnect [▶ 116]	Disconnects the session from the target. (Inherited from Session [▶ 107].)
	Dispose . [▶ 117]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from Session [▶ 107].)
	Dispose(Boolean) [▶ 725]	Releases unmanaged and - optionally - managed resources. (Overrides Session.Dispose(Boolean) [▶ 118].)

	Name	Description
	EnsureConnection [▶ 728]	Ensures, that the Session [▶ 92] is connected and returns the Connection [▶ 79] object.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize [▶ 725]	Finalizes an instance of the AdsSessionBase [▶ 716] class. (Overrides Object.Finalize .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetSessionName [▶ 725]	Gets the name/string identifier of the session. (Overrides Session.GetSessionName . [▶ 118].)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnConnect [▶ 726]	Handler function connecting the Session. (Overrides Session.OnConnect(Boolean) [▶ 119].)
	OnConnectAsync [▶ 123]	Handler function connecting the Session. (Inherited from Session [▶ 107].)
	OnConnectionStateChanged [▶ 121]	Handles the [E:ConnectionStateChanged] event. (Inherited from Session [▶ 107].)
	OnCreateSymbolServer [▶ 726]	Handler function creating the symbol server object. (Overrides Session.OnCreateSymbolServer . [▶ 119].)
	OnDisconnect [▶ 727]	Handler function disconnecting the session. (Overrides Session.OnDisconnect . [▶ 120].)
	OnGetAddress [▶ 727]	Handler function getting the address of the session. (Overrides Session.OnGetAddress . [▶ 120].)
	ToString [▶ 121]	Returns a String that represents this instance. (Inherited from Session [▶ 107].)



Reference

[AdsSessionBase Class](#) [[▶ 716](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.16.3.1 AdsSessionBase.Dispose Method

Overload List

	Name	Description
	Dispose . [▶ 117]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from Session [▶ 107].)
	Dispose(Boolean) [▶ 725]	Releases unmanaged and - optionally - managed resources. (Overrides Session.Dispose(Boolean) [▶ 118].)

Reference

[AdsSessionBase Class](#) [[▶ 716](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.16.3.1 AdsSessionBase.Dispose Method (Boolean)

Releases unmanaged and - optionally - managed resources.

Namespace: [TwinCAT.Ads](#) [[▶](#) [179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected override void Dispose(  
    bool disposing  
)
```

Parameters

disposing	Type: System.Boolean true to release both managed and unmanaged resources; false to release only unmanaged resources.
-----------	--

Reference

[AdsSessionBase Class](#) [[▶](#) [716](#)]

[Dispose Overload](#) [[▶](#) [724](#)]

[TwinCAT.Ads Namespace](#) [[▶](#) [179](#)]

6.2.16.3.2 AdsSessionBase.Finalize Method

Finalizes an instance of the [AdsSessionBase](#) [[▶](#) [716](#)] class.

Namespace: [TwinCAT.Ads](#) [[▶](#) [179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected override void Finalize()
```

Implements

[Object.Finalize](#).

Reference

[AdsSessionBase Class](#) [[▶](#) [716](#)]

[TwinCAT.Ads Namespace](#) [[▶](#) [179](#)]

6.2.16.3.3 AdsSessionBase.GetSessionName Method

Gets the name/string identifier of the session.

Namespace: [TwinCAT.Ads](#) [[▶](#) [179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected override string GetSessionName()
```

Return Value

Type: [String](#)
System.String.

Reference

[AdsSessionBase Class](#) [▸ 716]

[TwinCAT.Ads Namespace](#) [▸ 179]

6.2.16.3.4 AdsSessionBase.OnConnect Method

Handler function connecting the Session.

Namespace: [TwinCAT.Ads](#) [▸ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected override IConnection? OnConnect (
    bool reconnect
)
```

Parameters

reconnect	Type: System.Boolean
-----------	--------------------------------------

Return Value

Type: [IConnection](#) [▸ 79]
IConnection.

Reference

[AdsSessionBase Class](#) [▸ 716]

[TwinCAT.Ads Namespace](#) [▸ 179]

6.2.16.3.5 AdsSessionBase.OnCreateSymbolServer Method

Handler function creating the symbol server object.

Namespace: [TwinCAT.Ads](#) [▸ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected override ISymbolServer OnCreateSymbolServer()
```

Return Value

Type: [ISymbolServer](#) [[▶ 2719](#)]
[ISymbolServer](#).

Exceptions

Exception	Condition
SessionNotConnectedException [▶ 137]	The connection is not established!

Reference

[AdsSessionBase Class](#) [[▶ 716](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.16.3.6 **AdsSessionBase.OnDisconnect Method**

Handler function disconnecting the session.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected override bool OnDisconnect()
```

Return Value

Type: [Boolean](#)

true if the [Session](#) [[▶ 107](#)] is disconnected, false if the [Session](#) [[▶ 107](#)] was disconnected before.

Reference

[AdsSessionBase Class](#) [[▶ 716](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.16.3.7 **AdsSessionBase.OnGetAddress Method**

Handler function getting the address of the session.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected override string OnGetAddress()
```

Return Value

Type: [String](#)
[System.String](#).

Reference[AdsSessionBase Class \[▶ 716\]](#)[TwinCAT.Ads Namespace \[▶ 179\]](#)**6.2.16.3.8 AdsSessionBase.EnsureConnection Method**

Ensures, that the [ISession \[▶ 92\]](#) is connected and returns the [IConnection \[▶ 79\]](#) object.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public AdsConnection EnsureConnection()
```

Return ValueType: [AdsConnection \[▶ 412\]](#)

IConnection.

Exceptions

Exception	Condition
NotImplementedException	



Remarks

If the session is actually not connected an exception will be thrown.

Reference[AdsSessionBase Class \[▶ 716\]](#)[TwinCAT.Ads Namespace \[▶ 179\]](#)**6.2.16.4 AdsSessionBase Events**

The [AdsSessionBase \[▶ 716\]](#) type exposes the following members.

Events

	Name	Description
 	ConnectionStateChanged [▶ 124]	Occurs when connection status of the IConnectionStateProvider [▶ 89] has been changed. (Inherited from Session [▶ 107] .)

Reference[AdsSessionBase Class \[▶ 716\]](#)[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.17 AdsState Enumeration

Describes the AdsState.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum AdsState
```

Members

	Member name	Value	Description
	Invalid	0	Ads State is Invalid / Uninitialized
	Idle	1	Idle
	Reset	2	Reset
	Init	3	Initialize
	Start	4	Start
	Run	5	Run
	Stop	6	Stop
	SaveConfig	7	Save Configuration
	LoadConfig	8	Load Configuration
	PowerFailure	9	Power failure
	PowerGood	10	Power Good
	Error	11	Error
	Shutdown	12	Shutdown
	Suspend	13	Suspend
	Resume	14	Resume
	Config	15	Config (System is in config mode)
	Reconfig	16	Reconfig (System should restart in config mode)
	Stopping	17	Stopping
	Incompatible	18	Incompatible
	Exception	19	Exception

Reference

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.18 AdsStateChangedEventArgs Class

Arguments for the [AdsStateChanged \[▸ 1061\]](#) event.

Inheritance Hierarchy

[System.Object](#)

[System.EventArgs](#)

[TwinCAT.Ads.AdsStateChangedEventArgs](#)

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#


```
public sealed class AdsStateChangedEventArgs : EventArgs
```

The AdsStateChangedEventArgs type exposes the following members.





Constructors

	Name	Description
	AdsStateChangedEventArgs [▶ 730]	Initializes a new instance of the AdsStateChangedEventArgs class.

Properties

	Name	Description
	State [▶ 731]	Current state of the ADS device.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.18.1 AdsStateChangedEventArgs Constructor

Initializes a new instance of the AdsStateChangedEventArgs class.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsStateChangedEventArgs (
    StateInfo value
)
```

Parameters

value	Type: TwinCAT.Ads.StateInfo [▶ 1200] Current state of the ADS device.
-------	--

Reference


[AdsStateChangedEventArgs Class](#) [[▶ 729](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.18.2 AdsStateChangedEventArgs Properties

The [AdsStateChangedEventArgs](#) [▶ 729] type exposes the following members.

Properties

	Name	Description
	State [▶ 731]	Current state of the ADS device.

Reference

[AdsStateChangedEventArgs Class](#) [▶ 729]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.18.2.1 AdsStateChangedEventArgs.State Property

Current state of the ADS device.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public StateInfo State { get; }
```

Property Value

Type: [StateInfo](#) [▶ 1200]

The state.

Reference





[AdsStateChangedEventArgs Class](#) [▶ 729]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.18.3 AdsStateChangedEventArgs Methods

The [AdsStateChangedEventArgs](#) [▶ 729] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[AdsStateChangedEventArgs Class](#) [▶ 729]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.19 AdsStateChangedEventArgs2 Class

Event Arguments for AdsStateChanged events.

Inheritance Hierarchy

[System.Object](#)

[System.EventArgs](#)

[TwinCAT.Ads.AdsStateChangedEventArgs2](#)

Namespace: [TwinCAT.Ads](#) [[▶](#) 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229





Syntax

C#





```
public sealed class AdsStateChangedEventArgs2 : EventArgs
```

The AdsStateChangedEventArgs2 type exposes the following members.

Properties

	Name	Description
	Connection [▶ 733]	Gets the connection.
	NewState [▶ 733]	The new state
	OldState [▶ 734]	The old state
	Session [▶ 734]	The session

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference



[TwinCAT.Ads Namespace](#) [[▶](#) 179]



[System.EventArgs](#)

6.2.19.1 AdsStateChangedEventArgs2 Properties

The [AdsStateChangedEventArgs2](#) [[▶](#) 732] type exposes the following members.

Properties

	Name	Description
	Connection [▶ 733]	Gets the connection.
	NewState [▶ 733]	The new state

	Name	Description
	OldState [▶ 734]	The old state
	Session [▶ 734]	The session

Reference

[AdsStateChangedEventArgs2 Class](#) [[▶ 732](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.19.1.1 **AdsStateChangedEventArgs2.Connection Property**

Gets the connection.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IConnection? Connection { get; }
```

Property Value

Type: [IConnection](#) [[▶ 79](#)]

The connection.

Reference

[AdsStateChangedEventArgs2 Class](#) [[▶ 732](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.19.1.2 **AdsStateChangedEventArgs2.NewState Property**

The new state

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public StateInfo NewState { get; }
```

Property Value

Type: [StateInfo](#) [[▶ 1200](#)]

Reference

[AdsStateChangedEventArgs2 Class](#) [[▶ 732](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.19.1.3 AdsStateChangedEventArgs2.OldState Property

The old state

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public StateInfo OldState { get; }
```

Property Value

Type: [StateInfo](#) [[▶ 1200](#)]

Reference

[AdsStateChangedEventArgs2 Class](#) [[▶ 732](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.19.1.4 AdsStateChangedEventArgs2.Session Property

The session

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISession? Session { get; }
```

Property Value

Type: [ISession](#) [[▶ 92](#)]

Reference





[AdsStateChangedEventArgs2 Class](#) [[▶ 732](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.19.2 AdsStateChangedEventArgs2 Methods

The [AdsStateChangedEventArgs2](#) [[▶ 732](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[AdsStateChangedEventArgs2 Class \[▶ 732\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.20 AdsSumCommandException Class

The exception that is thrown when an ADS SumCommandBase error occurs.

Inheritance Hierarchy

System.Object
 System.Exception
 TwinCAT.AdsException [▶ 61]
 TwinCAT.Ads.AdsErrorException [▶ 682]
 TwinCAT.Ads.AdsSumCommandException

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229


Syntax

C#










```
[SerializableAttribute]
public sealed class AdsSumCommandException : AdsErrorException
```


The AdsSumCommandException type exposes the following members.

Constructors







	Name	Description
	AdsSumCommandException [▶ 736]	Initializes a new Instance of the AdsErrorException class.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	ErrorCode [▶ 686]	Gets the error code of the Exception. (Inherited from AdsErrorException [▶ 682] .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	SumCommand [▶ 737]	Gets the sum command.

	Name	Description
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData [▶ 738]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Overrides AdsErrorException.GetObjectData(SerializationInfo, StreamingContext) [▶ 689].)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Reference

[TwinCAT.Ads Namespace](#) [▶ [179](#)]

6.2.20.1 AdsSumCommandException Constructor

Initializes a new Instance of the [AdsErrorException](#) class.

Namespace: [TwinCAT.Ads](#) [▶ [179](#)]

Assembly: [TwinCAT.Ads](#) (in [TwinCAT.Ads.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsSumCommandException(
    string message,
    ISumCommand command
)
```

Parameters

message	Type: System.String The message.
command	Type: TwinCAT.Ads.SumCommand.ISumCommand [▶ 1527] The command.

Reference











[AdsSumCommandException Class](#) [▶ [735](#)]

[TwinCAT.Ads Namespace](#) [▶ [179](#)]

6.2.20.2 AdsSumCommandException Properties

The [AdsSumCommandException](#) [▶ [735](#)] type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	ErrorCode [▶ 686]	Gets the error code of the Exception. (Inherited from AdsErrorException [▶ 682].)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	SumCommand [▶ 737]	Gets the sum command.
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Reference

[AdsSumCommandException Class](#) [[▶ 735](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.20.2.1 AdsSumCommandException.SumCommand Property

Gets the sum command.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: [TwinCAT.Ads](#) (in [TwinCAT.Ads.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public ISumCommand SumCommand { get; }
```

Property Value

Type: [ISumCommand](#) [[▶ 1527](#)]

The sum command.

Reference







[AdsSumCommandException Class](#) [[▶ 735](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.20.3 AdsSumCommandException Methods

The [AdsSumCommandException](#) [[▶ 735](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData [▶ 738]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Overrides AdsErrorException.GetObjectData(SerializationInfo, StreamingContext) [▶ 689].)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Reference

[AdsSumCommandException Class](#) [▶ 735]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.20.3.1 AdsSumCommandException.GetObjectData Method

When overridden in a derived class, sets the [SerializationInfo](#) with information about the exception.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public override void GetObjectData(
    SerializationInfo info,
    StreamingContext context
)
```

Parameters

info	Type: System.Runtime.Serialization.SerializationInfo The SerializationInfo that holds the serialized object data about the exception being thrown.
context	Type: System.Runtime.Serialization.StreamingContext The StreamingContext that contains contextual information about the source or destination.

Implements

[ISerializable.GetObjectData\(SerializationInfo, StreamingContext\)](#)

Exceptions

Exception	Condition
ArgumentNullException	info

Reference

[AdsSumCommandException Class \[▶ 735\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.21 AdsSymbolVersionChangedEventArgs Class

Arguments for the [AdsSymbolVersionChanged \[▶ 1065\]](#) event.

Inheritance Hierarchy

System.Object
 System.EventArgs
 TwinCAT.Ads.AdsSymbolVersionChangedEventArgs

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#





```
public sealed class AdsSymbolVersionChangedEventArgs : EventArgs
```

The AdsSymbolVersionChangedEventArgs type exposes the following members.

Properties

	Name	Description
	SymbolVersion [▶ 740]	Current symbol version device.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)


Reference

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.21.1 AdsSymbolVersionChangedEventArgs Properties

The [AdsSymbolVersionChangedEventArgs \[▶ 739\]](#) type exposes the following members.

Properties

	Name	Description
	SymbolVersion [▶ 740]	Current symbol version device.

Reference

[AdsSymbolVersionChangedEventArgs Class \[► 739\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.21.1.1 AdsSymbolVersionChangedEventArgs.SymbolVersion Property

Current symbol version device.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public short SymbolVersion { get; }
```

Property Value

Type: [Int16](#)

The symbol version.

Reference





[AdsSymbolVersionChangedEventArgs Class \[► 739\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.21.2 AdsSymbolVersionChangedEventArgs Methods

The [AdsSymbolVersionChangedEventArgs \[► 739\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[AdsSymbolVersionChangedEventArgs Class \[► 739\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.22 AdsTransMode Enumeration

ADS Transmission Mode for ADS Notifications.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum AdsTransMode
```

Members

	Member name	Value	Description
	None	0	None / Uninitialized transport mode. No AdsNotification [▸ 991] event is fired.
	Cyclic	3	<p>The AdsNotification [▸ 991] event is fired cyclically.</p> <p>The Notification will be registered on the ADS Server side for a cyclical trigger (dependant on time parameter) and is bound to the 'default' task of the addressed target. In case of the PLC target (e.g. Port 851) the default task is the first configured task.</p> <p>Each time the 'default' task has finished its cycle the realtime system will check for the expired cycle time and sends the AdsNotification [▸ 991] message on expiry.</p> <p>The used ContextMask for the 'default' task is 0.</p> <p>Please be aware, that server side 'Change' notifications stress the realtime system and should be handled with care. Therefore, dependent of the cycle time of the task and the capabilities of the system only a limited set of Cyclic Notifications should be used!</p>
	OnChange	4	<p>On-Change AdsNotification [▸ 991] event.</p> <p>The Notification will be registered on the ADS Server side for an on-change and optional cyclical trigger (dependant on parameters) and is bound to the 'default' task of the addressed target. In case of the PLC target (e.g. Port 851) the default task is the first configured task.</p> <p>Each time this task has finished its cycle the realtime system will check for the changed value and an optional expired cycle time and sends the AdsNotification [▸ 991] message on change or expiry.</p> <p>The used ContextMask for the 'default' task is 0.</p> <p>Please be aware, that server side 'OnChange' notifications stress the realtime system / the default task with value comparisons.</p>

	Member name	Value	Description
			Therefore, dependent of the cycle time of the task and the capabilities of the system a higher amount of notification registrations should be handled with care !
	CyclicInContext	5	<p>The AdsNotification [▶ 991] event is fired cyclically within the given task context.</p> <p>A Value of parameter is interpreted as task context number ContextMask [▶ 1765]. This can be important, if the notifications have to be synchronous with specific tasks, but should not be used in the default case.</p> <p>The Notification will be registered on the ADS Server side for a cyclical trigger (dependant on time parameter) and is bound to the task specified by the ContextMask of the addressed target. In case of the PLC target (e.g. Port 851) the ContextMask is the Index of the global TASKINFOARRAY - 1.</p> <p>Each time this task has finished its cycle the realtime system will check for the expired cycle time and sends the AdsNotification [▶ 991] message on expiry.</p>
	OnChangeInContext	6	<p>The AdsNotification [▶ 991] event is fired when the data changes within the given task context.</p> <p>A Value of parameter is interpreted as task context number ContextMask [▶ 1765]. This can be important, if the notifications have to be synchronously with specific tasks, but should not be used in the default case.</p> <p>The Notification will be registered on the ADS Server side for an on-change and optional cyclical trigger (dependant on parameters) and is bound to the task specified by the ContextMask of the addressed target. In case of the PLC target (e.g. Port 851) the ContextMask is the Index of the global TASKINFOARRAY - 1. Each time this task has finished its cycle the realtime system will check for the changed value and an optional expired cycle time and sends the AdsNotification [▶ 991] message on change or expiry.</p>

	Member name	Value	Description
			<p>Please be aware, that server side 'OnChange' notifications stress the realtime system / the default task with value comparisons. Therefore, dependent of the cycle time of the task and the capabilities of the system only a limited set of OnChange Notifications should be used!</p>

Remarks

The AdsTransMode configures the registration of the [AdsNotification](#) [► 991] at the server system and how the parameters of the [AddDeviceNotification\(String, Int32, NotificationSettings, Object\)](#) [► 971] are interpreted. The following general scenarios are addressed:

- Cyclic notifications.
- Notifications on value change.
- Server side and Client side notifications.
- Binding of notifications to specific tasks.

In the default case the OnChange or the Cyclic (Server cycle) should be used. All other modes are side cases for special purposes.

More about the AdsNotifications: [ADS Notification concept](#) [► 26].

Reference

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsNotifications.AdsNotification](#) [► 991]

[IAdsNotifications.AdsNotificationEx](#) [► 993]

[AddDeviceNotification Overload](#) [► 971]

[AddDeviceNotificationEx Overload](#) [► 976]

6.2.23 AdsVersion Class

The structure contains the version number, revision number and build number.

Inheritance Hierarchy

[System.Object](#)

 TwinCAT.Ads.AdsVersion

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229






Syntax

C#











```
public class AdsVersion
```

The AdsVersion type exposes the following members.

Properties

	Name	Description
	Build [▶ 747]	Gets or sets the build number.
	Empty [▶ 747]	Get the Empty/Uninitialized Version (0,0,0)
	IsEmpty [▶ 747]	Gets a value indicating whether this instance is empty / uninitialized.
	Revision [▶ 748]	Gets or sets the revision number.
	Version [▶ 748]	Gets or sets the version number.

Methods

	Name	Description
	ConvertToStandard [▶ 749]	Converts this AdsVersion to a .NET Framework Version [▶ 748] object.
	Create(Version) [▶ 750]	Creates the AdsVersion object form Version .
	Create(Int32, Int32) [▶ 750]	Creates a new AdsVersion object.
	Create(Int32, Int32, Int32) [▶ 751]	Creates a new AdsVersion object.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)


Remarks


This version structure is used to indicate Version numbers of TwinCAT.AdsServers.

Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.23.1 AdsVersion Constructor**Overload List**

	Name	Description
	AdsVersion(Byte.) [▶ 745]	Initializes a new instance of the AdsVersion [▶ 743] class.

	Name	Description
	AdsVersion(Int32, Int32, Int32) [▶ 745]	Initializes a new instance of the AdsVersion [▶ 743] struct.

Reference

[AdsVersion Class](#) [[▶ 743](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.23.1.1 AdsVersion Constructor (.Byte.)

Initializes a new instance of the [AdsVersion](#) [[▶ 743](#)] class.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public AdsVersion(
    byte[] bytes
)
```

Parameters

bytes Type: [.System.Byte](#).
The bytes.

Exceptions

Exception	Condition
ArgumentNullException	bytes
ArgumentException	bytes

Reference

[AdsVersion Class](#) [[▶ 743](#)]

[AdsVersion Overload](#) [[▶ 744](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.23.1.2 AdsVersion Constructor (Int32, Int32, Int32)

Initializes a new instance of the [AdsVersion](#) [[▶ 743](#)] struct.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public AdsVersion(
    int version,
    int revision,
    int build
)
```

Parameters

version	Type: System.Int32 The version.
revision	Type: System.Int32 The revision.
build	Type: System.Int32 The build.

Exceptions

Exception	Condition
ArgumentOutOfRangeException	version or revision

Reference

[AdsVersion Class](#) [► 743]







[AdsVersion Overload](#) [► 744]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.23.2 AdsVersion Properties

The [AdsVersion](#) [► 743] type exposes the following members.

Properties

	Name	Description
	Build [► 747]	Gets or sets the build number.
 	Empty [► 747]	Get the Empty/Uninitialized Version (0,0,0)
	IsEmpty [► 747]	Gets a value indicating whether this instance is empty / uninitialized.
	Revision [► 748]	Gets or sets the revision number.
	Version [► 748]	Gets or sets the version number.

Reference

[AdsVersion Class](#) [► 743]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.23.2.1 **AdsVersion.Build Property**

Gets or sets the build number.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public short Build { get; set; }
```

Property Value

Type: [Int16](#)

Reference

[AdsVersion Class \[▸ 743\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.23.2.2 **AdsVersion.Empty Property**

Get the Empty/Uninitialized Version (0,0,0)

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AdsVersion Empty { get; }
```

Property Value

Type: [AdsVersion \[▸ 743\]](#)

The empty.

Reference

[AdsVersion Class \[▸ 743\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.23.2.3 **AdsVersion.IsEmpty Property**

Gets a value indicating whether this instance is empty / uninitialized.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsEmpty { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is empty; otherwise, false.

Reference

[AdsVersion Class](#) [► 743]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.23.2.4 AdsVersion.Revision Property

Gets or sets the revision number.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public byte Revision { get; set; }
```

Property Value

Type: [Byte](#)

Reference

[AdsVersion Class](#) [► 743]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.23.2.5 AdsVersion.Version Property

Gets or sets the version number.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public byte Version { get; set; }
```

Property Value

Type: [Byte](#)

Reference











[AdsVersion Class](#) [► 743]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.23.3 AdsVersion Methods

The [AdsVersion](#) [► 743] type exposes the following members.

Methods

	Name	Description
	ConvertToStandard [▶ 749]	Converts this AdsVersion [▶ 743] to a .NET Framework Version [▶ 748] object.
	Create(Version) [▶ 750]	Creates the AdsVersion [▶ 743] object form Version .
	Create(Int32, Int32) [▶ 750]	Creates a new AdsVersion [▶ 743] object.
	Create(Int32, Int32, Int32) [▶ 751]	Creates a new AdsVersion [▶ 743] object.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[AdsVersion Class](#) [▶ 743]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.23.3.1 **AdsVersion.ConvertToStandard Method**

Converts this [AdsVersion](#) [▶ 743] to a .NET Framework [Version](#) [▶ 748] object.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Version ConvertToStandard()
```

Return Value

Type: [Version](#)
Version.

Reference

[AdsVersion Class](#) [▶ 743]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.23.3.2 AdsVersion.Create Method

Overload List

	Name	Description
	Create(Version) [▶ 750]	Creates the AdsVersion [▶ 743] object form Version .
	Create(Int32, Int32) [▶ 750]	Creates a new AdsVersion [▶ 743] object.
	Create(Int32, Int32, Int32) [▶ 751]	Creates a new AdsVersion [▶ 743] object.

Reference

[AdsVersion Class](#) [\[▶ 743\]](#)

[TwinCAT.Ads Namespace](#) [\[▶ 179\]](#)

6.2.23.3.2.1 AdsVersion.Create Method (Version)

Creates the [AdsVersion](#) [\[▶ 743\]](#) object form [Version](#).

Namespace: [TwinCAT.Ads](#) [\[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AdsVersion Create(
    Version version
)
```

Parameters

version	Type: System.Version The version.
---------	--

Return Value

Type: [AdsVersion](#) [\[▶ 743\]](#)

AdsVersion.

Reference

[AdsVersion Class](#) [\[▶ 743\]](#)

[Create Overload](#) [\[▶ 750\]](#)

[TwinCAT.Ads Namespace](#) [\[▶ 179\]](#)

6.2.23.3.2.2 AdsVersion.Create Method (Int32, Int32)

Creates a new [AdsVersion](#) [\[▶ 743\]](#) object.

Namespace: [TwinCAT.Ads](#) [\[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AdsVersion Create(  
    int major,  
    int minor  
)
```

Parameters

major	Type: System.Int32 The major.
minor	Type: System.Int32 The minor.

Return Value

Type: [AdsVersion](#) [[▶ 743](#)]
AdsVersion.

Reference

[AdsVersion Class](#) [[▶ 743](#)]

[Create Overload](#) [[▶ 750](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.23.3.2.3 AdsVersion.Create Method (Int32, Int32, Int32)

Creates a new [AdsVersion](#) [[▶ 743](#)] object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AdsVersion Create(  
    int major,  
    int minor,  
    int build  
)
```

Parameters

major	Type: System.Int32 The major.
minor	Type: System.Int32 The minor.
build	Type: System.Int32 The build.

Return Value

Type: [AdsVersion](#) [[▶ 743](#)]
AdsVersion.

Reference[AdsVersion Class \[► 743\]](#)[Create Overload \[► 750\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.24 AmsAddress Class**

Ams/Ads Address

Inheritance Hierarchy[System.Object](#)











TwinCAT.Ads.AmsAddress


Namespace: [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public class AmsAddress
```





The AmsAddress type exposes the following members.

Constructors














	Name	Description
	AmsAddress. [► 754]	Protected constructor
	AmsAddress(AmsAddress) [► 756]	Copy constructor
	AmsAddress(AmsPort) [► 756]	Constructor
	AmsAddress(Int32) [► 755]	Constructor
	AmsAddress(String) [► 755]	Initializes a new instance of the AmsAddress class.
	AmsAddress(AmsNetId, Int32) [► 759]	Constructor
	AmsAddress(AmsNetId, AmsPort) [► 759]	Constructor
	AmsAddress(Byte, Int32) [► 757]	Constructor
	AmsAddress(Byte, AmsPort) [► 757]	Constructor
	AmsAddress(String, Int32) [► 758]	Constructor

	Name	Description
	AmsAddress(String, AmsPort) [▶ 758]	Constructor





Properties

	Name	Description
 	Empty [▶ 760]	Gets an Empty Address.
	NetId [▶ 760]	Gets the NetId
	Port [▶ 761]	Gets the Port number

Methods

	Name	Description
	Clone [▶ 762]	Clones this instance.
	Equals [▶ 762]	Equals (Overrides Object.Equals(Object) .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode [▶ 763]	Gets the HashCode of the Address (Overrides Object.GetHashCode() .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
 	Parse [▶ 763]	Parses a string to an AmsAddress object.
	SetNetId [▶ 764]	Sets the net identifier.
	SetPort [▶ 764]	Sets the port.
	ToString [▶ 765]	Converts the Address to String 'NetId:Port' (Overrides Object.ToString() .)
 	TryParse [▶ 765]	Tries to parse the AmsAddress from string.

Operators

	Name	Description
 	Equality [▶ 766]	Operator==
 	Inequality [▶ 766]	Implements the != operator.












Remarks

The `AmsAddress` consists of `NetId` [▶ 760] and `Port` [▶ 761] information and once it is constructed is immutable.

Reference

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.24.1 AmsAddress Constructor**Overload List**

	Name	Description
	<code>AmsAddress.</code> [▶ 754]	Protected constructor
	<code>AmsAddress(AmsAddress)</code> [▶ 756]	Copy constructor
	<code>AmsAddress(AmsPort)</code> [▶ 756]	Constructor
	<code>AmsAddress(Int32)</code> [▶ 755]	Constructor
	<code>AmsAddress(String)</code> [▶ 755]	Initializes a new instance of the <code>AmsAddress</code> [▶ 752] class.
	<code>AmsAddress(AmsNetId, Int32)</code> [▶ 759]	Constructor
	<code>AmsAddress(AmsNetId, AmsPort)</code> [▶ 759]	Constructor
	<code>AmsAddress(Byte, Int32)</code> [▶ 757]	Constructor
	<code>AmsAddress(Byte, AmsPort)</code> [▶ 757]	Constructor
	<code>AmsAddress(String, Int32)</code> [▶ 758]	Constructor
	<code>AmsAddress(String, AmsPort)</code> [▶ 758]	Constructor

Reference

[AmsAddress Class](#) [▶ 752]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.24.1.1 AmsAddress Constructor

Protected constructor

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected AmsAddress ()
```

Reference

[AmsAddress Class](#) [► 752]

[AmsAddress Overload](#) [► 754]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.24.1.2 AmsAddress Constructor (Int32)

Constructor

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsAddress(  
    int port  
)
```

Parameters

port	Type: System.Int32 The port.
------	---

Reference

[AmsAddress Class](#) [► 752]

[AmsAddress Overload](#) [► 754]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.24.1.3 AmsAddress Constructor (String)

Initializes a new instance of the [AmsAddress](#) [► 752] class.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsAddress(  
    string str  
)
```

Parameters

str	Type: System.String The address coded as string (Format NetId:Port, 1.2.3.4.5.6:Port)
-----	--

Reference[AmsAddress Class \[► 752\]](#)[AmsAddress Overload \[► 754\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.24.1.4 AmsAddress Constructor (AmsAddress)**

Copy constructor

Namespace: [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public AmsAddress(
    AmsAddress address
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [► 752] The address.
---------	--

Reference[AmsAddress Class \[► 752\]](#)[AmsAddress Overload \[► 754\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.24.1.5 AmsAddress Constructor (AmsPort)**

Constructor

Namespace: [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public AmsAddress(
    AmsPort port
)
```

Parameters

port	Type: TwinCAT.Ads.AmsPort [► 795] The port.
------	--

Reference[AmsAddress Class \[► 752\]](#)[AmsAddress Overload \[► 754\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.24.1.6 AmsAddress Constructor (.Byte., Int32)

Constructor

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsAddress(  
    byte[] netId,  
    int port  
)
```

Parameters

netId	Type: .System.Byte. The net identifier.
port	Type: System.Int32 The port.

Reference

[AmsAddress Class](#) [[▶ 752](#)]

[AmsAddress Overload](#) [[▶ 754](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.24.1.7 AmsAddress Constructor (.Byte., AmsPort)

Constructor

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsAddress(  
    byte[] netId,  
    AmsPort port  
)
```

Parameters

netId	Type: .System.Byte. The net identifier.
port	Type: TwinCAT.Ads.AmsPort [▶ 795] The port.

Reference

[AmsAddress Class](#) [[▶ 752](#)]

[AmsAddress Overload](#) [[▶ 754](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.24.1.8 AmsAddress Constructor (String, Int32)

Constructor

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsAddress(
    string netId,
    int port
)
```

Parameters

netId	Type: System.String The net identifier.
port	Type: System.Int32 The port.

Reference

[AmsAddress Class](#) [[▶ 752](#)]

[AmsAddress Overload](#) [[▶ 754](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.24.1.9 AmsAddress Constructor (String, AmsPort)

Constructor

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsAddress(
    string netId,
    AmsPort port
)
```

Parameters

netId	Type: System.String The net identifier.
port	Type: TwinCAT.Ads.AmsPort [▶ 795] The port.

Reference

[AmsAddress Class](#) [[▶ 752](#)]

[AmsAddress Overload](#) [[▶ 754](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.24.1.10 AmsAddress Constructor (AmsNetId, Int32)

Constructor

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsAddress(
    AmsNetId netId,
    int port
)
```

Parameters

netId	Type: TwinCAT.Ads.AmsNetId [► 767] Net Id
port	Type: System.Int32 Port

Reference

[AmsAddress Class](#) [► 752]

[AmsAddress Overload](#) [► 754]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.24.1.11 AmsAddress Constructor (AmsNetId, AmsPort)

Constructor

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsAddress(
    AmsNetId netId,
    AmsPort port
)
```

Parameters

netId	Type: TwinCAT.Ads.AmsNetId [► 767] The net identifier.
port	Type: TwinCAT.Ads.AmsPort [► 795] The port.

Reference

[AmsAddress Class](#) [► 752]




[AmsAddress Overload](#) [► 754]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.24.2 AmsAddress Properties

The [AmsAddress](#) [[▶ 752](#)] type exposes the following members.

Properties

	Name	Description
	Empty [▶ 760]	Gets an Empty Address.
	NetId [▶ 760]	Gets the NetId
	Port [▶ 761]	Gets the Port number

Reference

[AmsAddress Class](#) [[▶ 752](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.24.2.1 AmsAddress.Empty Property

Gets an Empty Address.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AmsAddress Empty { get; }
```

Property Value

Type: [AmsAddress](#) [[▶ 752](#)]

The empty.

Reference

[AmsAddress Class](#) [[▶ 752](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.24.2.2 AmsAddress.NetId Property

Gets the NetId

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsNetId NetId { get; }
```


Property Value

Type: [AmsNetId](#) [[▶ 767](#)]
 The net identifier.

Reference

[AmsAddress Class](#) [[▶ 752](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.24.2.3 AmsAddress.Port Property

Gets the Port number

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Port { get; }
```

Property Value

Type: [Int32](#)
 The port.

Reference









[AmsAddress Class](#) [[▶ 752](#)]





[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.24.3 AmsAddress Methods

The [AmsAddress](#) [[▶ 752](#)] type exposes the following members.

Methods

	Name	Description
	Clone [▶ 762]	Clones this instance.
	Equals [▶ 762]	Equals (Overrides Object.Equals(Object) .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode [▶ 763]	Gets the HashCode of the Address (Overrides Object.GetHashCode() .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Parse [▶ 763]	Parses a string to an AmsAddress [▶ 752] object.
	SetNetId [▶ 764]	Sets the net identifier.

	Name	Description
	SetPort [▶ 764]	Sets the port.
	ToString [▶ 765]	Converts the Address to String 'NetId:Port' (Overrides Object.ToString..)
	TryParse [▶ 765]	Tries to parse the AmsAddress [▶ 752] from string.
		

Reference

[AmsAddress Class](#) [[▶ 752](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.24.3.1 AmsAddress.Clone Method

Clones this instance.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsAddress Clone()
```

Return Value

Type: [AmsAddress](#) [[▶ 752](#)]

AmsAddress.

Reference

[AmsAddress Class](#) [[▶ 752](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.24.3.2 AmsAddress.Equals Method

Equals

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool Equals(
    Object? obj
)
```

Parameters

obj	Type: System.Object The object to compare with the current object.
-----	---

Return Value

Type: [Boolean](#)

true if the specified [Object](#) is equal to this instance; otherwise, false.

Reference

[AmsAddress Class](#) [[▶ 752](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.24.3.3 AmsAddress.GetHashCode Method

Gets the HashCode of the Address

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override int GetHashCode()
```

Return Value

Type: [Int32](#)

A hash code for this instance, suitable for use in hashing algorithms and data structures like a hash table.

Reference

[AmsAddress Class](#) [[▶ 752](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.24.3.4 AmsAddress.Parse Method

Parses a string to an [AmsAddress](#) [[▶ 752](#)] object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AmsAddress Parse(  
    string str  
)
```

Parameters

str	Type: System.String The string.
-----	--

Return Value

Type: [AmsAddress](#) [[▶ 752](#)]

AmsAddress.

Exceptions

Exception	Condition
FormatException	

Reference

[AmsAddress Class](#) [▶ 752]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.24.3.5 AmsAddress.SetNetId Method

Sets the net identifier.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
protected void SetNetId(
    AmsNetId netId
)
```

Parameters

netId	Type: TwinCAT.Ads.AmsNetId [▶ 767] The net identifier.
-------	---

Reference

[AmsAddress Class](#) [▶ 752]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.24.3.6 AmsAddress.SetPort Method

Sets the port.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
protected void SetPort(
    int port
)
```

Parameters

port	Type: System.Int32 The port.
------	---

Reference

[AmsAddress Class](#) [▶ 752]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.24.3.7 AmsAddress.ToString Method

Converts the Address to String 'NetId:Port'

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```

Return Value

Type: [String](#)

A [String](#) that represents this instance.

Reference

[AmsAddress Class](#) [► 752]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.24.3.8 AmsAddress.TryParse Method

Tries to parse the [AmsAddress](#) [► 752] from string.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool TryParse(  
    string str,  
    out AmsAddress?? address  
)
```

Parameters

str	Type: System.String The STR.
address	Type: TwinCAT.Ads.AmsAddress [► 752]. The address.

Return Value

Type: [Boolean](#)

true if the address is successfully parsed, false otherwise.

Reference





[AmsAddress Class](#) [► 752]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.24.4 AmsAddress Operators

The [AmsAddress](#) [► 752] type exposes the following members.

Operators

	Name	Description
 	Equality [▶ 766]	Operator==
 	Inequality [▶ 766]	Implements the != operator.

Reference

[AmsAddress Class](#) [[▶ 752](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.24.4.1 AmsAddress.Equality Operator

Operator==

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool operator ==(
    AmsAddress? o1,
    AmsAddress? o2
)
```

Parameters

o1	Type: TwinCAT.Ads.AmsAddress [▶ 752] The o1.
o2	Type: TwinCAT.Ads.AmsAddress [▶ 752] The o2.

Return Value

Type: [Boolean](#)

The result of the operator.

Reference

[AmsAddress Class](#) [[▶ 752](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.24.4.2 AmsAddress.Inequality Operator

Implements the != operator.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool operator !=(
    AmsAddress? o1,
    AmsAddress? o2
)
```

Parameters

o1	Type: TwinCAT.Ads.AmsAddress [▶ 752] The o1.
o2	Type: TwinCAT.Ads.AmsAddress [▶ 752] The o2.

Return Value

Type: [Boolean](#)
The result of the operator.

Reference

[AmsAddress Class](#) [▶ 752]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.25 AmsNetId Class

AMS/ADS Net ID

Inheritance Hierarchy

[System.Object](#)
TwinCAT.Ads.AmsNetId

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229





Syntax

C#















```
[SerializableAttribute]
public class AmsNetId : IComparable<AmsNetId>,
    IComparable
```

The AmsNetId type exposes the following members.












Constructors

	Name	Description
	AmsNetId(AmsNetId) [▶ 771]	Copy Constructor
	AmsNetId(Byte) [▶ 770]	Constructor
	AmsNetId(ReadOnlySpan.Byte) [▶ 772]	Initializes a new instance of the AmsNetId class from Memory.
	AmsNetId(String) [▶ 771]	Constructor

Properties

	Name	Description
 	Broadcast [▶ 773]	Gets the broadcast address (255.255.255.255.255.255)
 	Empty [▶ 774]	Creates an empty NetId ("0.0.0.0.0.0")
	IsBroadcast [▶ 774]	Gets a value indicating whether this AmsNetId is the broadcast address (255.255.255.255.255.255)
	IsEmpty [▶ 774]	Gets a value indicating whether this instance is empty / Uninitialized (AmsNetId: 0.0.0.0.0.0)
	IsLocal [▶ 775]	Determines, whether the AmsNetId is the Address of the local system (or the LocalHost [▶ 777] address).
	IsLoopback [▶ 775]	Indicates, that this AmsNetId indicates a Loopback (ID: 127.0.0.1.1.1)
	IsSubAddress [▶ 776]	Gets a value indicating whether this AmsNetId indicates a SubNet.
	Item [▶ 776]	Gets or sets the Byte with the specified index.
 	Local [▶ 777]	Gets the Local Net ID (System service must be running)
 	LocalHost [▶ 777]	Creates the local NetId ("127.0.0.1.1.1")


Methods

	Name	Description
	AsSpan [▶ 779]	Returns the AmsNetId as byte Span.
	Clone [▶ 779]	Clones the NetId
	CompareTo(AmsNetId) [▶ 781]	Compares the current object with another object of the same type.
	CompareTo(Object) [▶ 780]	Compares the current instance with another object of the same type and returns an integer that indicates whether the current instance precedes, follows, or occurs in the same position in the sort order as the other object.
	Equals(Object) [▶ 782]	Equals (Overrides Object.Equals(Object) .)
 	Equals(AmsNetId, AmsNetId) [▶ 782]	Determines whether the specified AmsNetIds are equal.
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
 	FromBinHexString [▶ 783]	Creates the AmsNetId from bin hex string.
	GetHashCode [▶ 783]	Gets the HashCode of the Address (Overrides Object.GetHashCode() .)

	Name	Description
	GetType	Gets the <u>Type</u> of the current instance. (Inherited from <u>Object</u> .)
 	IsSameTarget(AmsNetId, AmsNetId) [▶ 784]	Determines whether the AmsNetIds refer to the same target.
 	IsSameTarget(AmsNetId, AmsNetId, Boolean) [▶ 785]	Determines whether the AmsNetIds refer to the same target.
	MemberwiseClone	Creates a shallow copy of the current <u>Object</u> . (Inherited from <u>Object</u> .)
	NetIdsEqual(.Byte.) [▶ 786]	Compares the netIds
 	NetIdsEqual(.Byte., .Byte.) [▶ 787]	Compares the NetIds
 	NetIdsEqual(.Byte., .Byte., Boolean) [▶ 787]	Compares the NetIds for SubID equality.
 	Parse [▶ 788]	Converts the string representation of the address to AmsNetId.
	ToBinHex. [▶ 789]	Converts the AmsNetId to a BinHex string.
 	ToBinHex(AmsNetId) [▶ 789]	Converts the specified AmsNetId to a BinHex string.
	ToBytes [▶ 790]	Converts the NetId object to byte array
	ToString. [▶ 791]	Converts the netId to string (Overrides <u>Object.ToString</u> .)
	ToString(String, IFormatProvider) [▶ 791]	Returns a <u>String</u> that represents this instance.
 	TryGetLocalNetId [▶ 793]	Gets the Local NetID
 	TryParse [▶ 792]	Converts the string representation of the address to AmsNetId.
	TryWriteBytes [▶ 792]	Writes the AmsNetId to memory.

Operators

	Name	Description
 	Equality [▶ 794]	Operator==
	Inequality [▶ 794]	Implements the != operator.





	Name	Description
		

Reference

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.25.1 AmsNetId Constructor

Overload List

	Name	Description
	AmsNetId(AmsNetId) [▶ 771]	Copy Constructor
	AmsNetId(.Byte.) [▶ 770]	Constructor
	AmsNetId(ReadOnlySpan.Byte.) [▶ 772]	Initializes a new instance of the AmsNetId [▶ 767] class from Memory.
	AmsNetId(String) [▶ 771]	Constructor

Reference

[AmsNetId Class](#) [▶ 767]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.25.1.1 AmsNetId Constructor (.Byte.)

Constructor

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsNetId(
    byte[] netId
)
```

Parameters

netId	Type: .System.Byte. Net ID in bytes
-------	--

Exceptions

Exception	Condition
ArgumentException	Not a valid NetId;netId

Reference

[AmsNetId Class](#) [▶ 767]

[AmsNetId Overload \[▸ 770\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.25.1.2 AmsNetId Constructor (ReadOnlySpan`1)

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public AmsNetId(  
    ReadOnlySpan data  
)
```

Parameters

data Type: [ReadOnlySpan](#)

Reference

[AmsNetId Class \[▸ 767\]](#)

[AmsNetId Overload \[▸ 770\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.25.1.3 AmsNetId Constructor (String)

Constructor

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsNetId(  
    string netId  
)
```

Parameters

netId	Type: System.String NetID as string
-------	--

Reference

[AmsNetId Class \[▸ 767\]](#)

[AmsNetId Overload \[▸ 770\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.25.1.4 AmsNetId Constructor (AmsNetId)

Copy Constructor

Namespace: [TwinCAT.Ads](#) [▸ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsNetId(
    AmsNetId netId
)
```

Parameters

netId	Type: TwinCAT.Ads.AmsNetId [▸ 767] Net Id.
-------	---

Reference

[AmsNetId Class](#) [▸ 767]

[AmsNetId Overload](#) [▸ 770]

[TwinCAT.Ads Namespace](#) [▸ 179]

6.2.25.1.5 AmsNetId Constructor (ReadOnlySpan.Byte.)

Initializes a new instance of the [AmsNetId](#) [▸ 767] class from Memory.

Namespace: [TwinCAT.Ads](#) [▸ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsNetId(
    ReadOnlySpan<byte> data
)
```

Parameters

data	Type: System.ReadOnlySpan.Byte . The data.
------	---

Exceptions

Exception	Condition
ArgumentException	Not a valid NetId - data

Reference

[AmsNetId Class](#) [▸ 767]











[AmsNetId Overload](#) [▸ 770]

[TwinCAT.Ads Namespace](#) [▸ 179]

6.2.25.2 AmsNetId Properties

The [AmsNetId](#) [▸ 767] type exposes the following members.

Properties

	Name	Description
 S	Broadcast [▶ 773]	Gets the broadcast address (255.255.255.255.255.255)
 S	Empty [▶ 774]	Creates an empty NetId ("0.0.0.0.0.0")
	IsBroadcast [▶ 774]	Gets a value indicating whether this AmsNetId [▶ 767] is the broadcast address (255.255.255.255.255.255)
	IsEmpty [▶ 774]	Gets a value indicating whether this instance is empty / Uninitialized (AmsNetId: 0.0.0.0.0.0)
	IsLocal [▶ 775]	Determines, whether the AmsNetId [▶ 767] is the Address of the local system (or the LocalHost [▶ 777] address).
	IsLoopback [▶ 775]	Indicates, that this AmsNetId [▶ 767] indicates a Loopback (ID: 127.0.0.1.1.1)
	IsSubAddress [▶ 776]	Gets a value indicating whether this AmsNetId [▶ 767] indicates a SubNet.
	Item [▶ 776]	Gets or sets the Byte with the specified index.
 S	Local [▶ 777]	Gets the Local Net ID (System service must be running)
 S	LocalHost [▶ 777]	Creates the local NetId ("127.0.0.1.1.1")

Reference

[AmsNetId Class](#) [[▶ 767](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.25.2.1 AmsNetId.Broadcast Property

Gets the broadcast address (255.255.255.255.255.255)

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AmsNetId Broadcast { get; }
```

Property Value

Type: [AmsNetId](#) [[▶ 767](#)]

The broadcast.

Reference

[AmsNetId Class](#) [[▶ 767](#)]

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.25.2.2 AmsNetId.Empty Property

Creates an empty NetId ("0.0.0.0.0.0")

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AmsNetId Empty { get; }
```

Property Value

Type: [AmsNetId \[► 767\]](#)

The empty.

Reference

[AmsNetId Class \[► 767\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.25.2.3 AmsNetId.IsBroadcast Property

Gets a value indicating whether this [AmsNetId \[► 767\]](#) is the broadcast address (255.255.255.255.255.255)

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsBroadcast { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is broadcast; otherwise, false.

Reference

[AmsNetId Class \[► 767\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.25.2.4 AmsNetId.IsEmpty Property

Gets a value indicating whether this instance is empty / Uninitialized (AmsNetId: 0.0.0.0.0.0)

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsEmpty { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is empty; otherwise, false.

Reference

[AmsNetId Class](#) [► 767]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.25.2.5 AmsNetId.IsLocal Property

Determines, whether the [AmsNetId](#) [► 767] is the Address of the local system (or the [LocalHost](#) [► 777] address).

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsLocal { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is local; otherwise, false.

Remarks

True is also returned for the local host AmsNetId. Comparing the [AmsNetId](#) [► 767] doesn't include the SubNetIds (Byte-Indices 4 and 5), what means SubNetIds are ignored.

Reference

[AmsNetId Class](#) [► 767]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.25.2.6 AmsNetId.IsLoopback Property

Indicates, that this [AmsNetId](#) [► 767] indicates a Loopback (ID: 127.0.0.1.1.1)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsLoopback { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is loop back; otherwise, false.

Reference

[AmsNetId Class](#) [► 767]

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.25.2.7 AmsNetId.IsSubAddress Property

Gets a value indicating whether this [AmsNetId \[► 767\]](#) indicates a SubNet.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsSubAddress { get; }
```

Property Value

Type: [Boolean](#)

true if this instance indicates a subnet; otherwise, false.

Remarks

A Subnet is indicated, when the [AmsNetId \[► 767\]](#) doesn't end with **X.X.X.X.1.1**

Reference

[AmsNetId Class \[► 767\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.25.2.8 AmsNetId.Item Property

Gets or sets the [Byte](#) with the specified index.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public byte this[  
    int i  
] { get; set; }
```

Parameters

i	Type: System.Int32 The index.
---	--

Return Value

Type: [Byte](#)
System.Byte.

Reference

[AmsNetId Class \[► 767\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.25.2.9 AmsNetId.Local Property

Gets the Local Net ID (System service must be running)

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AmsNetId Local { get; }
```

Property Value

Type: [AmsNetId](#) [[▶ 767](#)]

The local.

Remarks

The system service must be running

Reference

[AmsNetId Class](#) [[▶ 767](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.25.2.10 AmsNetId.LocalHost Property

Creates the local NetId ("127.0.0.1.1.1")

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AmsNetId LocalHost { get; }
```

Property Value

Type: [AmsNetId](#) [[▶ 767](#)]

The local host.

Reference



[AmsNetId Class](#) [[▶ 767](#)]









[TwinCAT.Ads Namespace](#) [[▶ 179](#)]







6.2.25.3 AmsNetId Methods

The [AmsNetId](#) [[▶ 767](#)] type exposes the following members.

Methods

	Name	Description
	AsSpan [▶ 779]	Returns the AmsNetId [▶ 767] as byte Span.
	Clone [▶ 779]	Clones the NetId

	Name	Description
	CompareTo(AmsNetId) [▶ 781]	Compares the current object with another object of the same type.
	CompareTo(Object) [▶ 780]	Compares the current instance with another object of the same type and returns an integer that indicates whether the current instance precedes, follows, or occurs in the same position in the sort order as the other object.
	Equals(Object) [▶ 782]	Equals (Overrides Object.Equals(Object) .)
 	Equals(AmsNetId, AmsNetId) [▶ 782]	Determines whether the specified AmsNetId [▶ 767]s are equal.
 	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
 	FromBinHexString [▶ 783]	Creates the AmsNetId [▶ 767] from bin hex string.
	GetHashCode [▶ 783]	Gets the GetHashCode of the Address (Overrides Object.GetHashCode .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
 	IsSameTarget(AmsNetId, AmsNetId) [▶ 784]	Determines whether the AmsNetId [▶ 767]s refer to the same target.
 	IsSameTarget(AmsNetId, AmsNetId, Boolean) [▶ 785]	Determines whether the AmsNetId [▶ 767]s refer to the same target.
 	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	NetIdsEqual(.Byte.) [▶ 786]	Compares the netIds
 	NetIdsEqual(.Byte., .Byte.) [▶ 787]	Compares the NetIds
 	NetIdsEqual(.Byte., .Byte., Boolean) [▶ 787]	Compares the NetIds for SubID equality.
 	Parse [▶ 788]	Converts the string representation of the address to AmsNetId [▶ 767].
	ToBinHex . [▶ 789]	Converts the AmsNetId [▶ 767] to a BinHex string.
 	ToBinHex(AmsNetId) [▶ 789]	Converts the specified AmsNetId [▶ 767] to a BinHex string.
	ToBytes [▶ 790]	Converts the NetId object to byte array
	ToString . [▶ 791]	Converts the netId to string (Overrides Object.ToString .)

	Name	Description
	ToString(String, IFormatProvider) [▶ 791]	Returns a String that represents this instance.
 	TryGetLocalNetId [▶ 793]	Gets the Local NetID
 	TryParse [▶ 792]	Converts the string representation of the address to AmsNetId [▶ 767].
	TryWriteBytes [▶ 792]	Writes the AmsNetId [▶ 767] to memory.

Reference

[AmsNetId Class](#) [▶ 767]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.25.3.1 AmsNetId.AsSpan Method

Returns the [AmsNetId](#) [▶ 767] as byte Span.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlySpan<byte> AsSpan()
```

Return Value

Type: [ReadOnlySpan.Byte](#).

[ReadOnlySpan<System.Byte>](#).

Reference

[AmsNetId Class](#) [▶ 767]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.25.3.2 AmsNetId.Clone Method

Clones the NetId

Namespace: [TwinCAT.Ads](#) [▶ 179]



Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsNetId Clone()
```

Return ValueType: [AmsNetId](#) [[▶ 767](#)]The cloned [AmsNetId](#) [[▶ 767](#)]**Reference**[AmsNetId Class](#) [[▶ 767](#)][TwinCAT.Ads Namespace](#) [[▶ 179](#)]**6.2.25.3.3 AmsNetId.CompareTo Method****Overload List**

	Name	Description
	CompareTo(AmsNetId) [▶ 781]	Compares the current object with another object of the same type.
	CompareTo(Object) [▶ 780]	Compares the current instance with another object of the same type and returns an integer that indicates whether the current instance precedes, follows, or occurs in the same position in the sort order as the other object.

Reference[AmsNetId Class](#) [[▶ 767](#)][TwinCAT.Ads Namespace](#) [[▶ 179](#)]**6.2.25.3.3.1 AmsNetId.CompareTo Method (Object)**

Compares the current instance with another object of the same type and returns an integer that indicates whether the current instance precedes, follows, or occurs in the same position in the sort order as the other object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public int CompareTo(
    Object? obj
)
```

Parameters

obj	Type: System.Object An object to compare with this instance.
-----	---

Return ValueType: [Int32](#)

A value that indicates the relative order of the objects being compared. The return value has these meanings: Value Meaning Less than zero This instance precedes obj in the sort order. Zero This instance occurs in the same position in the sort order as obj. Greater than zero This instance follows obj in the sort order.

Implements[IComparable.CompareTo\(Object\)](#)**Reference**[AmsNetId Class \[► 767\]](#)[CompareTo Overload \[► 780\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.25.3.3.2 AmsNetId.CompareTo Method (AmsNetId)**

Compares the current object with another object of the same type.

Namespace: [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public int CompareTo(
    AmsNetId? other
)
```


Parameters


other	Type: TwinCAT.Ads.AmsNetId [► 767] An object to compare with this object.
-------	--

Return ValueType: [Int32](#)

A value that indicates the relative order of the objects being compared. The return value has the following meanings: Value Meaning Less than zero This object is less than the other parameter.Zero This object is equal to other. Greater than zero This object is greater than other.

Implements[IComparable.T..CompareTo\(T\)](#)**Reference**[AmsNetId Class \[► 767\]](#)[CompareTo Overload \[► 780\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.25.3.4 AmsNetId.Equals Method****Overload List**

	Name	Description
	Equals(Object) [► 782]	Equals (Overrides Object.Equals(Object) .)

	Name	Description
	Equals(AmsNetId, AmsNetId) [▶ 782]	Determines whether the specified AmsNetId [▶ 767]s are equal.

Reference

[AmsNetId Class](#) [[▶ 767](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.25.3.4.1 AmsNetId.Equals Method (Object)

Equals

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool Equals(
    Object? obj
)
```

Parameters

obj	Type: System.Object The object to compare with the current object.
-----	---

Return Value

Type: [Boolean](#)

true if the specified [Object](#) is equal to this instance; otherwise, false.

Reference

[AmsNetId Class](#) [[▶ 767](#)]

[Equals Overload](#) [[▶ 781](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.25.3.4.2 AmsNetId.Equals Method (AmsNetId, AmsNetId)

Determines whether the specified [AmsNetId](#) [[▶ 767](#)]s are equal.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool Equals(
    AmsNetId netIDA,
    AmsNetId netIDB
)
```

Parameters

netIDA	Type: TwinCAT.Ads.AmsNetId [▶ 767] The net IDA.
netIDB	Type: TwinCAT.Ads.AmsNetId [▶ 767] The net IDB.

Return ValueType: [Boolean](#)

true if the specified net IDA is equal; otherwise, false.

Reference[AmsNetId Class](#) [▶ 767][Equals Overload](#) [▶ 781][TwinCAT.Ads Namespace](#) [▶ 179]**6.2.25.3.5 AmsNetId.FromBinHexString Method**Creates the [AmsNetId](#) [▶ 767] from bin hex string.**Namespace:** [TwinCAT.Ads](#) [▶ 179]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static AmsNetId FromBinHexString(
    string str
)
```

Parameters

str	Type: System.String The BinHex string.
-----	---

Return ValueType: [AmsNetId](#) [▶ 767]

AmsNetId.

Reference[AmsNetId Class](#) [▶ 767][TwinCAT.Ads Namespace](#) [▶ 179]**6.2.25.3.6 AmsNetId.GetHashCode Method**

Gets the HashCode of the Address

Namespace: [TwinCAT.Ads](#) [▶ 179]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override int GetHashCode()
```

Return Value

Type: [Int32](#)

A hash code for this instance, suitable for use in hashing algorithms and data structures like a hash table.





Reference

[AmsNetId Class](#) [► 767]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.25.3.7 AmsNetId.IsSameTarget Method

Overload List

	Name	Description
 	IsSameTarget(AmsNetId, AmsNetId) [► 784]	Determines whether the AmsNetId [► 767]s refer to the same target.
 	IsSameTarget(AmsNetId, AmsNetId, Boolean) [► 785]	Determines whether the AmsNetId [► 767]s refer to the same target.

Reference

[AmsNetId Class](#) [► 767]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.25.3.7.1 AmsNetId.IsSameTarget Method (AmsNetId, AmsNetId)

Determines whether the [AmsNetId](#) [► 767]s refer to the same target.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool IsSameTarget(
    AmsNetId netIDA,
    AmsNetId netIDB
)
```

Parameters

netIDA	Type: TwinCAT.Ads.AmsNetId [► 767] NetID of target system A
netIDB	Type: TwinCAT.Ads.AmsNetId [► 767] NetID of target system B

Return Value

Type: [Boolean](#)
 true if the target systems are the same, otherwise false.

Remarks

In comparison to the [Equals\(Object\) \[▸ 782\]](#) or [Equals\(AmsNetId, AmsNetId\) \[▸ 782\]](#) methods, this Method also checks against the LocalHost ID, which means that [LocalHost \[▸ 777\]](#) is the same target as [Local \[▸ 777\]](#)

Reference

- [AmsNetId Class \[▸ 767\]](#)
- [IsSameTarget Overload \[▸ 784\]](#)
- [TwinCAT.Ads Namespace \[▸ 179\]](#)
- [AmsNetId.Equals\(Object\) \[▸ 782\]](#)
- [AmsNetId.Equals\(AmsNetId, AmsNetId\) \[▸ 782\]](#)

6.2.25.3.7.2 AmsNetId.IsSameTarget Method (AmsNetId, AmsNetId, Boolean)

Determines whether the [AmsNetId \[▸ 767\]](#)s refer to the same target.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)
Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool IsSameTarget(
    AmsNetId netIDA,
    AmsNetId netIDB,
    bool ignoreSubId
)
```

Parameters

netIDA	Type: TwinCAT.Ads.AmsNetId [▸ 767] NetID of target system A
netIDB	Type: TwinCAT.Ads.AmsNetId [▸ 767] NetID of target system B
ignoreSubId	Type: System.Boolean Indicates only to check the 4 parts of the address (for SubId check).

Return Value

Type: [Boolean](#)
 true if the target systems are the same, otherwise false.

Exceptions

Exception	Condition
ArgumentNullException	netIDA
ArgumentNullException	netIDB

Remarks

In comparison to the [Equals\(Object\)](#) [▶ 782] or [Equals\(AmsNetId, AmsNetId\)](#) [▶ 782] methods, this Method also checks against the LocalHost ID, which means that [LocalHost](#) [▶ 777] is the same target as [Local](#) [▶ 777]

Reference

[AmsNetId Class](#) [▶ 767]






[IsSameTarget Overload](#) [▶ 784]

[TwinCAT.Ads Namespace](#) [▶ 179]

[AmsNetId.Equals\(Object\)](#) [▶ 782]

[AmsNetId.Equals\(AmsNetId, AmsNetId\)](#) [▶ 782]

6.2.25.3.8 AmsNetId.NetIdsEqual Method**Overload List**

	Name	Description
	NetIdsEqual(.Byte.) [▶ 786]	Compares the netIds
 	NetIdsEqual(.Byte., .Byte.) [▶ 787]	Compares the NetIds
 	NetIdsEqual(.Byte., .Byte., Boolean) [▶ 787]	Compares the NetIds for SubID equality.

Reference

[AmsNetId Class](#) [▶ 767]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.25.3.8.1 AmsNetId.NetIdsEqual Method (.Byte.)

Compares the netIds

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public bool NetIdsEqual(
    byte[] netId
)
```

Parameters

netId	Type: .System.Byte. NetId in bytes.
-------	--

Return Value

Type: [Boolean](#)

true if the NetIds are equal, false otherwise.

Reference

[AmsNetId Class](#) [► 767]

[NetIdsEqual Overload](#) [► 786]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.25.3.8.2 AmsNetId.NetIdsEqual Method (.Byte., .Byte.)

Compares the NetIds

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool NetIdsEqual(  
    byte[] netId1,  
    byte[] netId2  
)
```

Parameters

netId1	Type: .System.Byte. NetID1
netId2	Type: .System.Byte. NetId2

Return Value

Type: [Boolean](#)

true if the NetIds are equal, false otherwise.

Reference

[AmsNetId Class](#) [► 767]

[NetIdsEqual Overload](#) [► 786]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.25.3.8.3 AmsNetId.NetIdsEqual Method (.Byte., .Byte., Boolean)

Compares the NetIds for SubID equality.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool NetIdsEqual(
    byte[] netId1,
    byte[] netId2,
    bool ignoreSubID
)
```

Parameters

netId1	Type: .System.Byte . NetID1
netId2	Type: .System.Byte . NetId2
ignoreSubID	Type: System.Boolean Ignores the SubId part of the NetId (last 2 numbers/bytes).

Return Value

Type: [Boolean](#)
true if the SubIds are equal, false otherwise.

Remarks

The first 4 numbers indicate the global ID of the AmsNetId. The last 2 numbers indicate the (local) submodule.

Reference

[AmsNetId Class](#) [► 767]

[NetIdsEqual Overload](#) [► 786]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.25.3.9 AmsNetId.Parse Method

Converts the string representation of the address to [AmsNetId](#) [► 767].

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AmsNetId Parse(
    string str
)
```

Parameters

str	Type: System.String The string to parse.
-----	---

Return Value

Type: [AmsNetId](#) [► 767]
AmsNetId.

Exceptions




Exception	Condition
FormatException	Format of AmsNetId is not valid!

Reference

[AmsNetId Class](#) [► 767]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.25.3.10 AmsNetId.ToBinHex Method**Overload List**

	Name	Description
	ToBinHex. [► 789]	Converts the AmsNetId [► 767] to a BinHex string.
 	ToBinHex(AmsNetId) [► 789]	Converts the specified AmsNetId [► 767] to a BinHex string.

Reference

[AmsNetId Class](#) [► 767]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.25.3.10.1 AmsNetId.ToBinHex Method

Converts the [AmsNetId](#) [► 767] to a BinHex string.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public string ToBinHex()
```

Return Value

Type: [String](#)
System.String.

Reference

[AmsNetId Class](#) [► 767]

[ToBinHex Overload](#) [► 789]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.25.3.10.2 AmsNetId.ToBinHex Method (AmsNetId)

Converts the specified [AmsNetId](#) [► 767] to a BinHex string.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static string ToBinHex(
    AmsNetId netId
)
```

Parameters

netId	Type: TwinCAT.Ads.AmsNetId [▶ 767] NetId to convert
-------	--

Return Value

Type: [String](#)
System.String.

Reference

[AmsNetId Class](#) [[▶ 767](#)]

[ToBinHex Overload](#) [[▶ 789](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.25.3.11 AmsNetId.ToBytes Method

Converts the NetId object to byte array

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public byte[] ToBytes()
```

Return Value

Type: [.Byte](#)
System.Byte[].



Reference

[AmsNetId Class](#) [[▶ 767](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.25.3.12 AmsNetId.ToString Method

Overload List

	Name	Description
	ToString . [▶ 791]	Converts the netId to string (Overrides Object.ToString ..)
	ToString (String , IFormatProvider) [▶ 791]	Returns a String that represents this instance.

Reference

[AmsNetId Class](#) [► 767]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.25.3.12.1 AmsNetId.ToString Method

Converts the netId to string

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```

Return Value

Type: [String](#)

A [String](#) that represents this instance.

Reference

[AmsNetId Class](#) [► 767]

[ToString Overload](#) [► 790]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.25.3.12.2 AmsNetId.ToString Method (String, IFormatProvider)

Returns a [String](#) that represents this instance.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string ToString(  
    string? format,  
    IFormatProvider? formatProvider  
)
```

Parameters

format	Type: System.String The format.
formatProvider	Type: System.IFormatProvider The format provider.

Return Value

Type: [String](#)

A [String](#) that represents this instance.

Exceptions

Exception	Condition
FormatException	

Remarks

Formatting	Description
g	Standard formatting
x	Formatting as Hexadecimal (small letters)
X	Formatting as Hexadecimal (big letters)

Reference

[AmsNetId Class](#) [► 767]

[ToString Overload](#) [► 790]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.25.3.13 AmsNetId.TryParse Method

Converts the string representation of the address to [AmsNetId](#) [► 767].

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static bool TryParse(
    string str,
    out AmsNetId?? netId
)
```

Parameters

str	Type: System.String The string to parse.
netId	Type: TwinCAT.Ads.AmsNetId [► 767]. The parsed AmsNetId [► 767].

Return Value

Type: [Boolean](#)
true if parsed, false otherwise.

Reference

[AmsNetId Class](#) [► 767]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.25.3.14 AmsNetId.TryWriteBytes Method

Writes the [AmsNetId](#) [► 767] to memory.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryWriteBytes(  
    Span<byte> span  
)
```

Parameters

span	Type: System.Span.Byte . The memory
------	--

Return Value

Type: [Boolean](#)
true if writing succeeded, false otherwise.

Reference

[AmsNetId Class](#) [► 767]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.25.3.15 AmsNetId.TryGetLocalNetId Method

Gets the Local NetID

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool TryGetLocalNetId(  
    out AmsNetId?? netId  
)
```

Parameters

netId	Type: TwinCAT.Ads.AmsNetId [► 767]. The local AmsNetId
-------	---

Return Value

Type: [Boolean](#)
true if the local NetId is determined, false otherwise.

Reference





[AmsNetId Class](#) [► 767]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.25.4 AmsNetId Operators

The [AmsNetId](#) [► 767] type exposes the following members.

Operators

	Name	Description
 	Equality [▶ 794]	Operator==
 	Inequality [▶ 794]	Implements the != operator.

Reference

[AmsNetId Class \[▶ 767\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.25.4.1 AmsNetId.Equality Operator

Operator==

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool operator ==(
    AmsNetId? o1,
    AmsNetId? o2
)
```

Parameters

o1	Type: TwinCAT.Ads.AmsNetId [▶ 767] The o1.
o2	Type: TwinCAT.Ads.AmsNetId [▶ 767] The o2.

Return Value

Type: [Boolean](#)

The result of the operator.

Reference

[AmsNetId Class \[▶ 767\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.25.4.2 AmsNetId.Inequality Operator

Implements the != operator.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool operator !=(
    AmsNetId? o1,
    AmsNetId? o2
)
```

Parameters

o1	Type: TwinCAT.Ads.AmsNetId [▶ 767] The o1.
o2	Type: TwinCAT.Ads.AmsNetId [▶ 767] The o2.

Return Value

Type: [Boolean](#)
The result of the operator.

Reference

[AmsNetId Class](#) [▶ 767]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.26 AmsPort Enumeration

Ams Ports Definitions.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum AmsPort
```

Members

	Member name	Value	Description
	Router	1	AMS Router (Port 1)
	Debugger	2	AMS Debugger (Port 2)
	R0_TComServer	10	The TCom Server. Dpc or passive level.
	R0_TComServerTask	11	TCom Server Task. RT context.
	R0_TComServer_PL	12	TCom Serve Task. Passive level.
	R0_TcDebugger	20	TwinCAT Debugger
	R0_TcDebuggerTask	21	TwinCAT Debugger Task
	R0_LicenseServer	30	The License Server (Port 30)
	Logger	100	Logger (Port 100)
	EventLog	110	Event Logger (Port 110)
	DeviceApplication	120	application for coupler (EK), gateway (EL), etc.
	EventLog_UM	130	Event Logger UM
	EventLog_RT	131	Event Logger RT
	EventLogPublisher	132	Event Logger Publisher

	Member name	Value	Description
	R0_Realtime	200	R0 Realtime (Port 200)
	R0_Trace	290	R0 Trace (Port 290)
	R0_IO	300	R0 IO (Port 300)
	R0_NC	500	NC (R0) (Port 500)
	R0_NCSAF	501	R0 Satzausführung (Port 501)
	R0_NCSVB	511	R0 Satzvorbereitung (Port 511)
	R0_NCINSTANCE	520	Preconfigured Nc2-Nc3-Instance
	R0_ISG	550	R0 ISG (Port 550)
	R0_CNC	600	R0 CNC (Port 600)
	R0_LINE	700	R0 Line (Port 700)
	R0_PLC	800	R0 PLC (Port 800)
	Tc2_Plc1	801	PLC runtime on Port 801 (TwinCAT 2, Runtime System 1)
	PlcRuntime_801	801	PLC runtime on Port 801 (TwinCAT 2, Runtime System 1)
	Tc2_Plc2	811	PLC runtime on Port 811 (TwinCAT 2, Runtime System 2)
	PlcRuntime_811	811	PLC runtime on Port 811 (TwinCAT 2, Runtime System 2)
	Tc2_Plc3	821	PLC runtime on Port 821 (TwinCAT 2, Runtime System 3)
	PlcRuntime_821	821	PLC runtime on Port 821 (TwinCAT 2, Runtime System 3)
	Tc2_Plc4	831	PLC runtime on Port 831 (TwinCAT 2, Runtime System 4)
	PlcRuntime_831	831	PLC runtime on Port 831 (TwinCAT 2, Runtime System 4)
	R0_RTS	850	R0 RTS (Port 850)
	PlcRuntime_851	851	PLC runtime on Port 851 (TwinCAT 3)
	PlcRuntime_852	852	PLC runtime on Port 852 (TwinCAT 3)
	PlcRuntime_853	853	PLC runtime on Port 853 (TwinCAT 3)
	PlcRuntime_854	854	PLC runtime on Port 854 (TwinCAT 3)
	PlcRuntime_855	855	PLC runtime on Port 855 (TwinCAT 3)
	PlcRuntime_856	856	PLC runtime on Port 856 (TwinCAT 3)
	PlcRuntime_857	857	PLC runtime on Port 857 (TwinCAT 3)
	PlcRuntime_858	858	PLC runtime on Port 858 (TwinCAT 3)
	PlcRuntime_859	859	PLC runtime on Port 859 (TwinCAT 3)
	PlcRuntime_860	860	PLC runtime on Port 860 (TwinCAT 3)
	CamshaftController	900	Camshaft Controller (R0) (Port 900)
	R0_CAMTOOL	950	R0 CAM Tool (Port 950)
	R0_USER	2000	R0 User (Port 2000)

	Member name	Value	Description
	SystemService	10000	System Service (AMSPORT_R3_SYSSERV, 10000)
	R3_CTRLPROG	10000	(Port 10000)
	R3_SYSCTRL	10001	(Port 10001)
	R3_SYSSAMPLER	10100	Port 10100
	R3_TCPRAWCONN	10200	Port 10200
	R3_TCPIPSERVER	10201	Port 10201
	R3_SYSMANAGER	10300	Port 10300
	R3_SMSSERVER	10400	Port 10400
	R3_MODBUSSERVER	10500	Port 10500
	R3_AMSLOGGER	10502	Port 10502
	R3_XMLDATASERVER	10600	Port 10600
	R3_AUTOCONFIG	10700	Port 10700
	R3_PLCCONTROL	10800	Port 10800
	R3_FTPCLIENT	10900	Port 10900
	R3_NCCTRL	11000	Port 11000
	R3_NCINTERPRETER	11500	Port 11500
	R3_GSTINTERPRETER	11600	Port 11600
	R3_STRECKECTRL	12000	Port 12000
	R3_CAMCTRL	13000	Port 13000
	R3_SCOPE	14000	Port 14000
	R3_CONDITIONMON	14100	Port 14100
	R3_SINECH1	15000	Port 15000
	R3_CONTROLNET	16000	Port 16000
	R3_OPCTSERVER	17000	Port 17000
	R3_OPCCCLIENT	17500	Port 17500
	R3_MAILSERVER	18000	Port 18000
	R3_EL60XX	19000	Port 19000
	R3_MANAGEMENT	19100	Port 19100
	R3_MIELEHOME	19200	Port 19200
	R3_CPLINK3	19300	Port 19300
	R3_VNSERVICE	19500	Port 19500
	MultiUser	19600	Multiuser (Port 19600)
	R3_NI845X	19700	(AMSPORT_R3_NI845X,19700)
	HmiServer	19800	(AMSPORT_R3_HMISERVER,19800)
	UserModeRuntime	20000	(AMSPORT_R3_UMRUNTIME,20000)
	UNFIXEDPORT	0	UNFIXEDPORT / Uninitialized
	USEDEFAULT	65535	Use default port

Remarks

The AmsPort enumeration defines the pre-occupied port numbers for Beckhoff AmsServers. Additionally the free areas for custom AmsServers are defined here.

Port Range	Description
1 .. 24999 (0x01 .. 0x61A7)	Internally Reserved for Beckhoff AMS Servers.

Port Range	Description
CUSTOMER_FIRST [▶ 1233] (25000, 0x61A8) <= PORT <= CUSTOMER_LAST [▶ 1233] (25999, 658F)	Reserved for Customer use. Don't forget to register your port at Beckhoff to prevent Port conflicts with other released AmsServers.
CUSTOMERPRIVATE_FIRST [▶ 1233] (26000, 0x6590) <= PORT <= CUSTOMERPRIVATE_LAST [▶ 1233] (26999, 6977)	Reserved for Customer private use. Private use means that AMSServers with this port never leave the customer private network and the customer is responsible by himself to prevent port conflicts in this area.
32768 .. PORT_LAST [▶ 1233] (65534) (0x8000 .. 0xFFFFE)	Dynamic ports used for AMSServers defined with UNFIXEDPORT. This is used by AdsClients which are in fact an AMSServer with dynamic port.

Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.27 AmsRouterNotificationEventArgs Class

Arguments for the [IRouterNotificationProvider](#) [[▶ 1099](#)] events.

Inheritance Hierarchy

System.Object
 System.EventArgs
 TwinCAT.Ads.AmsRouterNotificationEventArgs

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#


```
public sealed class AmsRouterNotificationEventArgs : EventArgs
```

The AmsRouterNotificationEventArgs type exposes the following members.


Constructors




	Name	Description
	AmsRouterNotificationEventArgs [▶ 799]	Initializes a new instance of the AmsRouterNotificationEventArgs class.

Properties

	Name	Description
	State [▶ 799]	Current state of the AMS Router.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)

	Name	Description
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[TwinCAT.Ads Namespace](#) |> [179](#)]

6.2.27.1 AmsRouterNotificationEventArgs Constructor

Initializes a new instance of the AmsRouterNotificationEventArgs class.

Namespace: [TwinCAT.Ads](#) |> [179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsRouterNotificationEventArgs(
    AmsRouterState value
)
```

Parameters

value	Type: TwinCAT.Ads.AmsRouterState > 800 Value of the ADS variable.
-------	---

Reference


[AmsRouterNotificationEventArgs Class](#) |> [798](#)]

[TwinCAT.Ads Namespace](#) |> [179](#)]

6.2.27.2 AmsRouterNotificationEventArgs Properties

The [AmsRouterNotificationEventArgs](#) |> [798](#)] type exposes the following members.

Properties

	Name	Description
	State > 799]	Current state of the AMS Router.

Reference

[AmsRouterNotificationEventArgs Class](#) |> [798](#)]

[TwinCAT.Ads Namespace](#) |> [179](#)]

6.2.27.2.1 AmsRouterNotificationEventArgs.State Property

Current state of the AMS Router.

Namespace: [TwinCAT.Ads](#) |> [179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsRouterState State { get; }
```

Property Value

Type: [AmsRouterState](#) [[▶ 800](#)]

Reference





[AmsRouterNotificationEventArgs Class](#) [[▶ 798](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.27.3 AmsRouterNotificationEventArgs Methods

The [AmsRouterNotificationEventArgs](#) [[▶ 798](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[AmsRouterNotificationEventArgs Class](#) [[▶ 798](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.28 AmsRouterState Enumeration

State of the AMS Router.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum AmsRouterState
```

Members

	Member name	Value	Description
	None	0	Unknown Router State (None, Uninitialized)
	Stopped	1	AMS Router is stopped.
	Started	2	AMS Router is started.
	Removed	3	AMS Router has been removed.

Reference

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.29 DeviceInfo Class

The structure contains the name and the version information of the device.

Inheritance Hierarchy

[System.Object](#)

 TwinCAT.Ads.DeviceInfo

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#





```
public class DeviceInfo
```

The DeviceInfo type exposes the following members.







Constructors

	Name	Description
	DeviceInfo [▶ 802]	Initializes a new instance of the DeviceInfo class.

Properties

	Name	Description
	Empty [▶ 802]	Empty / Uninitialized DeviceInfo
	IsEmpty [▶ 803]	Gets a value indicating whether this instance is empty.
	Name [▶ 803]	Gets the name of the device.
	Version [▶ 804]	Gets the version information.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.29.1 DeviceInfo Constructor

Initializes a new instance of the [DeviceInfo](#) [▶ 801] class.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DeviceInfo(
    string name,
    AdsVersion version
)
```

Parameters

name	Type: System.String The name.
version	Type: TwinCAT.Ads.AdsVersion [▶ 743] The version.

Reference






[DeviceInfo Class](#) [▶ 801]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.29.2 DeviceInfo Properties

The [DeviceInfo](#) [▶ 801] type exposes the following members.

Properties

	Name	Description
	Empty [▶ 802]	Empty / Uninitialized DeviceInfo
		
	IsEmpty [▶ 803]	Gets a value indicating whether this instance is empty.
	Name [▶ 803]	Gets the name of the device.
	Version [▶ 804]	Gets the version information.

Reference

[DeviceInfo Class](#) [▶ 801]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.29.2.1 DeviceInfo.Empty Property

Empty / Uninitialized DeviceInfo

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static DeviceInfo Empty { get; }
```

Property Value

Type: [DeviceInfo](#) [[▶ 801](#)]

The empty.

Reference

[DeviceInfo Class](#) [[▶ 801](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.29.2.2 DeviceInfo.IsEmpty Property

Gets a value indicating whether this instance is empty.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsEmpty { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is empty; otherwise, false.

Reference

[DeviceInfo Class](#) [[▶ 801](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.29.2.3 DeviceInfo.Name Property

Gets the name of the device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Name { get; }
```

Property Value

Type: [String](#)







Reference[DeviceInfo Class \[► 801\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.29.2.4 DeviceInfo.Version Property**

Gets the version information.

Namespace: [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public AdsVersion Version { get; }
```

Property ValueType: [AdsVersion \[► 743\]](#)**Reference**[DeviceInfo Class \[► 801\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.29.3 DeviceInfo Methods**The [DeviceInfo \[► 801\]](#) type exposes the following members.**Methods**

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference[DeviceInfo Class \[► 801\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.30 IAdsAnyAccess Interface**

Interface for accessing ADS 'Any' objects.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229













Syntax





C#

```
public interface IAdsAnyAccess
```

Methods

	Name	Description
	ReadAny(UInt32, Type) [▶ 812]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, Type, .Int32.) [▶ 813]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type) [▶ 815]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type, .Int32.) [▶ 815]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32) [▶ 810]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, .Int32.) [▶ 811]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, UInt32) [▶ 812]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, UInt32, .Int32.) [▶ 814]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, Type, CancellationToken) [▶ 819]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, Type, .Int32., CancellationToken) [▶ 821]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, UInt32, Type, CancellationToken) [▶ 823]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, UInt32, Type, .Int32., CancellationToken) [▶ 823]	Reads data asynchronously from an ADS device and writes it to an object.

	Name	Description
	ReadAnyAsync.T. (UInt32, CancellationToken) [▶ 817]	Reads data asynchronously from an ADS device.
	ReadAnyAsync.T. (UInt32, .Int32, CancellationToken) [▶ 818]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T. (UInt32, UInt32, CancellationToken) [▶ 820]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T. (UInt32, UInt32, .Int32, CancellationToken) [▶ 822]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyString(UInt 32, Int32, Encoding) [▶ 825]	Reads a string from the specified symbol/variable.
	ReadAnyString(UInt 32, UInt32, Int32, Encoding) [▶ 825]	Reads as string from a specified address.
	ReadAnyStringAsyn c(UInt32, Int32, Encoding, CancellationToken) [▶ 827]	Reads a string asynchronously from the specified symbol/variable
	ReadAnyStringAsyn c(UInt32, UInt32, Int32, Encoding, CancellationToken) [▶ 827]	Reads a string from a specified address asynchronously.
	WriteAny(UInt32, Object) [▶ 829]	Writes an object synchronously to an ADS device.
	WriteAny(UInt32, Object, .Int32.) [▶ 829]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAny(UInt32, UInt32, Object) [▶ 830]	Writes an object synchronously to an ADS device.
	WriteAny(UInt32, UInt32, Object, .Int32.) [▶ 830]	Writes an object synchronously to an ADS device.









	Name	Description
	WriteAnyAsync(UInt32, Object, CancellationToken) [▶ 832]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAnyAsync(UInt32, Object, .Int32, CancellationToken) [▶ 833]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAnyAsync(UInt32, UInt32, Object, CancellationToken) [▶ 834]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAnyAsync(UInt32, UInt32, Object, .Int32, CancellationToken) [▶ 834]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.












Reference










[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.30.1 IAdsAnyAccess Methods

Methods

	Name	Description
	ReadAny(UInt32, Type) [▶ 812]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, Type, .Int32) [▶ 813]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type) [▶ 815]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type, .Int32) [▶ 815]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32) [▶ 810]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, .Int32) [▶ 811]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, UInt32) [▶ 812]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, UInt32, .Int32) [▶ 814]	Reads data synchronously from an ADS device and writes it to an object.

	Name	Description
	ReadAnyAsync(UInt32, Type, CancellationToken) [▶ 819]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, Type, .Int32., CancellationToken) [▶ 821]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, UInt32, Type, CancellationToken) [▶ 823]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, UInt32, Type, .Int32., CancellationToken) [▶ 823]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T.(UInt32, CancellationToken) [▶ 817]	Reads data asynchronously from an ADS device.
	ReadAnyAsync.T.(UInt32, .Int32., CancellationToken) [▶ 818]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T.(UInt32, UInt32, CancellationToken) [▶ 820]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T.(UInt32, UInt32, .Int32., CancellationToken) [▶ 822]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyString(UInt32, Int32, Encoding) [▶ 825]	Reads a string from the specified symbol/variable.
	ReadAnyString(UInt32, UInt32, Int32, Encoding) [▶ 825]	Reads as string from a specified address.
	ReadAnyStringAsync(UInt32, Int32, Encoding, CancellationToken) [▶ 827]	Reads a string asynchronously from the specified symbol/variable

	Name	Description
	ReadAnyStringAsyn c(UInt32, UInt32, Int32, Encoding, CancellationToken) [▶ 827]	Reads a string from a specified address asynchronously.
	WriteAny(UInt32, Object) [▶ 829]	Writes an object synchronously to an ADS device.
	WriteAny(UInt32, Object, .Int32.) [▶ 829]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAny(UInt32, UInt32, Object) [▶ 830]	Writes an object synchronously to an ADS device.
	WriteAny(UInt32, UInt32, Object, .Int32.) [▶ 830]	Writes an object synchronously to an ADS device.
	WriteAnyAsync(UInt 32, Object, CancellationToken) [▶ 832]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAnyAsync(UInt 32, Object, .Int32., CancellationToken) [▶ 833]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAnyAsync(UInt 32, UInt32, Object, CancellationToken) [▶ 834]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAnyAsync(UInt 32, UInt32, Object, .Int32., CancellationToken) [▶ 834]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.









Reference

[IAdsAnyAccess Interface](#) [▶ 804]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.30.1.1 IAdsAnyAccess.ReadAny Method

Overload List

	Name	Description
	ReadAny.T.(UInt32) [► 810]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, Type) [► 812]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, .Int32.) [► 811]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, UInt32) [► 812]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, Type, .Int32.) [► 813]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type) [► 815]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny.T.(UInt32, UInt32, .Int32.) [► 814]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAny(UInt32, UInt32, Type, .Int32.) [► 815]	Reads data synchronously from an ADS device and writes it to an object.

Reference

[IAdsAnyAccess Interface](#) [► 804]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.30.1.1.1 IAdsAnyAccess.ReadAny.T. Method (UInt32)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
T ReadAny<T>(
    uint variableHandle
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
----------------	--

Type Parameters

T	The type of the value to read.
---	--------------------------------

Return Value

Type: T
The value of the read symbol.

Reference

[IAdsAnyAccess Interface](#) [► 804]

[ReadAny Overload](#) [► 810]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.30.1.1.2 IAdsAnyAccess.ReadAny.T. Method (UInt32, .Int32.)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
T ReadAny<T>(
    uint variableHandle,
    int[]? args
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
args	Type: .System.Int32 . Additional arguments.

Type Parameters

T	The type of the value to read.
---	--------------------------------

Return Value

Type: T
The value of the read symbol.

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [► 2927].
string[]	
Array	

Reference

[IAdsAnyAccess Interface](#) [► 804]

[ReadAny Overload \[▶ 810\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.30.1.1.3 IAdsAnyAccess.ReadAny Method (UInt32, Type)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object ReadAny(
    uint variableHandle,
    Type type
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
type	Type: System.Type Type of the object to be read.

Return Value

Type: [Object](#)
The read object.

Reference

[IAdsAnyAccess Interface \[▶ 804\]](#)

[ReadAny Overload \[▶ 810\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.30.1.1.4 IAdsAnyAccess.ReadAny.T. Method (UInt32, UInt32)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
T ReadAny<T>(
    uint indexGroup,
    uint indexOffset
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.

Type Parameters

T	The type of the object to be read.
---	------------------------------------

Return Value

Type: T
The read value.

Reference

[IAdsAnyAccess Interface \[▶ 804\]](#)

[ReadAny Overload \[▶ 810\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.30.1.1.5 IAdsAnyAccess.ReadAny Method (UInt32, Type, .Int32.)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object ReadAny(
    uint variableHandle,
    Type type,
    int[]? args
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
type	Type: System.Type Type of the object to be read.
args	Type: .System.Int32 . Additional arguments.

Return Value

Type: [Object](#)
The read value.

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927] .
string[]	
Array	

Reference[IAdsAnyAccess Interface \[▸ 804\]](#)[ReadAny Overload \[▸ 810\]](#)[TwinCAT.Ads Namespace \[▸ 179\]](#)**6.2.30.1.1.6 IAdsAnyAccess.ReadAny.T. Method (UInt32, UInt32, .Int32.)**

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
T ReadAny<T>(
    uint indexGroup,
    uint indexOffset,
    int[]? args
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
args	Type: .System.Int32 . Additional arguments.

Type Parameters

T	The type of the object to be read.
---	------------------------------------

Return Value

Type: T
The read value.

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▸ 2927] .
string[]	
Array	

Reference[IAdsAnyAccess Interface \[▸ 804\]](#)[ReadAny Overload \[▸ 810\]](#)

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.30.1.1.7 IAdsAnyAccess.ReadAny Method (UInt32, UInt32, Type)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object ReadAny(
    uint indexGroup,
    uint indexOffset,
    Type type
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
type	Type: System.Type Type of the object to be read.

Return Value

Type: [Object](#)
The read value.

Reference

[IAdsAnyAccess Interface](#) [▶ 804]

[ReadAny Overload](#) [▶ 810]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.30.1.1.8 IAdsAnyAccess.ReadAny Method (UInt32, UInt32, Type, .Int32.)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object ReadAny(
    uint indexGroup,
    uint indexOffset,
    Type type,
    int[]? args
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
------------	---

indexOffset	Type: System.UInt32 Index offset of the ADS variable.
type	Type: System.Type Type of the object to be read.
args	Type: .System.Int32 . Additional arguments.

Return Value

Type: [Object](#)
The read value.

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference





[IAdsAnyAccess Interface](#) [[▶ 804](#)]





[ReadAny Overload](#) [[▶ 810](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.30.1.2 IAdsAnyAccess.ReadAnyAsync Method

Overload List

	Name	Description
	ReadAnyAsync.T. (UInt32 , CancellationToken) [▶ 817]	Reads data asynchronously from an ADS device.
	ReadAnyAsync(UInt32, Type, CancellationToken) [▶ 819]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T. (UInt32 , .Int32 , CancellationToken) [▶ 818]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T. (UInt32 , UInt32 , CancellationToken) [▶ 820]	Reads data asynchronously from an ADS device and writes it to an object.

	Name	Description
	ReadAnyAsync(UInt32, Type, .Int32., CancellationToken) [▶ 821]	Reads data synchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, UInt32, Type, CancellationToken) [▶ 823]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync.T.(UInt32, UInt32, .Int32., CancellationToken) [▶ 822]	Reads data asynchronously from an ADS device and writes it to an object.
	ReadAnyAsync(UInt32, UInt32, Type, .Int32., CancellationToken) [▶ 823]	Reads data asynchronously from an ADS device and writes it to an object.

Reference

[IAdsAnyAccess Interface](#) [▶ 804]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.30.1.2.1 IAdsAnyAccess.ReadAnyAsync.T. Method (UInt32, CancellationToken)

Reads data asynchronously from an ADS device.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultValue<T>> ReadAnyAsync<T>(
    uint variableHandle,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable/symbol handle.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Type Parameters

T	The Type of the value to be read.
---	-----------------------------------

Return Value

Type: [Task.ResultValue](#) [[▸ 1181](#)].T..

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [[▸ 1129](#)] parameter contains the read value ([Value](#) [[▸ 1185](#)]) and the [ErrorCode](#) [[▸ 1120](#)] after execution.

Remarks

As object types only primitive types are supported.

Reference

[IAdsAnyAccess Interface](#) [[▸ 804](#)]

[ReadAnyAsync Overload](#) [[▸ 816](#)]

[TwinCAT.Ads Namespace](#) [[▸ 179](#)]

Also see about this

 [ResultAnyValue.Value Property](#) [[▸ 1131](#)]

6.2.30.1.2.2 IAdsAnyAccess.ReadAnyAsync.T. Method (UInt32, .Int32., CancellationToken)

Reads data asynchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [[▸ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultValue<T>> ReadAnyAsync<T>(
    uint variableHandle,
    int[]? args,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
args	Type: .System.Int32 . Additional arguments.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Type Parameters

T	Type of the object to be read
---	-------------------------------

Return Value

Type: [Task.ResultValue](#) [[▸ 1181](#)].T..

A task that represents the asynchronous read operation. The [ResultValue.TValue.](#) [[▸ 1181](#)] parameter contains the read value ([Value](#) [[▸ 1185](#)]) and the [ErrorCode](#) [[▸ 1120](#)] after execution.

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference

[IAdsAnyAccess Interface](#) [[▶ 804](#)]

[ReadAnyAsync Overload](#) [[▶ 816](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.30.1.2.3 IAdsAnyAccess.ReadAnyAsync Method (UInt32, Type, CancellationToken)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultAnyValue> ReadAnyAsync (
    uint variableHandle,
    Type type,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable/symbol handle.
type	Type: System.Type Type of the object to be read.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAnyValue](#) [[▶ 1129](#)].

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [[▶ 1129](#)] parameter contains the read value ([Value](#) [[▶ 1185](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Remarks

As object types only primitive types are supported.

Reference

[IAdsAnyAccess Interface](#) [[▶ 804](#)]

[ReadAnyAsync Overload \[► 816\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

Also see about this

 [ResultAnyValue.Value Property \[► 1131\]](#)

6.2.30.1.2.4 IAdsAnyAccess.ReadAnyAsync.T. Method (UInt32, UInt32, CancellationToken)

Reads data asynchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultValue<T>> ReadAnyAsync<T>(
    uint indexGroup,
    uint indexOffset,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Type Parameters

T

Return Value

Type: [Task.ResultValue \[► 1181\].T..](#)
The asynchronous result.

Return Value

Type: [Task.ResultValue \[► 1181\].T..](#)

A task that represents the asynchronous read operation. The [ResultValue.TValue. \[► 1181\]](#) parameter contains the read value ([Value \[► 1185\]](#)) and the [ErrorCode \[► 1120\]](#) after execution.

Reference

[IAdsAnyAccess Interface \[► 804\]](#)

[ReadAnyAsync Overload \[► 816\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.30.1.2.5 IAdsAnyAccess.ReadAnyAsync Method (UInt32, Type, .Int32., CancellationToken)

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultAnyValue> ReadAnyAsync (
    uint variableHandle,
    Type type,
    int[]? args,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
type	Type: System.Type Type of the object to be read.
args	Type: .System.Int32 . Additional arguments.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAnyValue](#) [[▶ 1129](#)].

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [[▶ 1129](#)] parameter contains the read value ([Value](#) [[▶ 1185](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference

[IAdsAnyAccess Interface](#) [[▶ 804](#)]

[ReadAnyAsync Overload](#) [[▶ 816](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

- 📖 [ResultAnyValue.Value Property](#) [[▶ 1131](#)]

6.2.30.1.2.6 IAdsAnyAccess.ReadAnyAsync.T. Method (UInt32, UInt32, .Int32., CancellationToken)

Reads data asynchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultValue<T>> ReadAnyAsync<T>(
    uint indexGroup,
    uint indexOffset,
    int[]? args,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
args	Type: .System.Int32 . Additional arguments.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Type Parameters

T	The type of the result value.
---	-------------------------------

Return Value

Type: [Task.ResultValue](#) [[▶ 1181](#)].T..

A task that represents the asynchronous read operation. The [ResultValue.TValue](#). [[▶ 1181](#)] parameter contains the read value ([Value](#) [[▶ 1185](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference

[IAdsAnyAccess Interface](#) [[▶ 804](#)]

[ReadAnyAsync Overload](#) [[▶ 816](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.30.1.2.7 IAdsAnyAccess.ReadAnyAsync Method (UInt32, UInt32, Type, CancellationToken)

Reads data asynchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultAnyValue> ReadAnyAsync (
    uint indexGroup,
    uint indexOffset,
    Type type,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
type	Type: System.Type Type of the object to be read.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAnyValue](#) [[▶ 1129](#)].

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [[▶ 1129](#)] parameter contains the read value ([Value](#) [[▶ 1185](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Reference

[IAdsAnyAccess Interface](#) [[▶ 804](#)]

[ReadAnyAsync Overload](#) [[▶ 816](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

 [ResultAnyValue.Value Property](#) [[▶ 1131](#)]

6.2.30.1.2.8 IAdsAnyAccess.ReadAnyAsync Method (UInt32, UInt32, Type, .Int32., CancellationToken)

Reads data asynchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultAnyValue> ReadAnyAsync (
    uint indexGroup,
    uint indexOffset,
```

```

Type type,
int[]? args,
CancellationTokel cancel
)

```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
type	Type: System.Type Type of the object to be read.
args	Type: System.Int32 . Additional arguments.
cancel	Type: System.Threading.CancellationTokel The cancellation token.

Return Value

Type: [Task.ResultAnyValue](#) [[▶ 1129](#)].

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [[▶ 1129](#)] parameter contains the read value ([Value](#) [[▶ 1185](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference

[IAdsAnyAccess Interface](#) [[▶ 804](#)]

[ReadAnyAsync Overload](#) [[▶ 816](#)]


[TwinCAT.Ads Namespace](#) [[▶ 179](#)]


Also see about this

-  [ResultAnyValue.Value Property](#) [[▶ 1131](#)]

6.2.30.1.3 IAdsAnyAccess.ReadAnyString Method

Overload List

	Name	Description
	ReadAnyString(UInt32, Int32, Encoding) [▶ 825]	Reads a string from the specified symbol/variable.

	Name	Description
	ReadAnyString(UInt32, UInt32, Int32, Encoding) [▶ 825]	Reads as string from a specified address.

Reference

[IAdsAnyAccess Interface](#) [[▶ 804](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.30.1.3.1 IAdsAnyAccess.ReadAnyString Method (UInt32, Int32, Encoding)

Reads a string from the specified symbol/variable.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string ReadAnyString(
    uint variableHandle,
    int len,
    Encoding encoding
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
len	Type: System.Int32 The length.
encoding	Type: System.Text.Encoding The encoding.

Return Value

Type: [String](#)
The string value.

Reference

[IAdsAnyAccess Interface](#) [[▶ 804](#)]

[ReadAnyString Overload](#) [[▶ 824](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.30.1.3.2 IAdsAnyAccess.ReadAnyString Method (UInt32, UInt32, Int32, Encoding)

Reads as string from a specified address.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string ReadAnyString(
    uint indexGroup,
    uint indexOffset,
    int len,
    Encoding encoding
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
len	Type: System.Int32 The string length to be read.
encoding	Type: System.Text.Encoding The encoding.

Return Value

Type: [String](#)
System.String.

Reference



[IAdsAnyAccess Interface](#) [► 804]

[ReadAnyString Overload](#) [► 824]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.30.1.4 IAdsAnyAccess.ReadAnyStringAsync Method

Overload List

	Name	Description
	ReadAnyStringAsync(UInt32, Int32, Encoding, CancellationToken) [► 827]	Reads a string asynchronously from the specified symbol/variable
	ReadAnyStringAsync(UInt32, UInt32, Int32, Encoding, CancellationToken) [► 827]	Reads a string from a specified address asynchronously.

Reference

[IAdsAnyAccess Interface](#) [► 804]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.30.1.4.1 IAdsAnyAccess.ReadAnyStringAsync Method (UInt32, Int32, Encoding, CancellationToken)

Reads a string asynchronously from the specified symbol/variable

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultAnyValue> ReadAnyStringAsync(
    uint variableHandle,
    int len,
    Encoding encoding,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
len	Type: System.Int32 The length.
encoding	Type: System.Text.Encoding The encoding.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAnyValue](#) [[▶ 1129](#)].

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [[▶ 1129](#)] parameter contains the read string ([Value](#) [[▶ 1185](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Reference

[IAdsAnyAccess Interface](#) [[▶ 804](#)]

[ReadAnyStringAsync Overload](#) [[▶ 826](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

 [ResultAnyValue.Value Property](#) [[▶ 1131](#)]

6.2.30.1.4.2 IAdsAnyAccess.ReadAnyStringAsync Method (UInt32, UInt32, Int32, Encoding, CancellationToken)

Reads a string from a specified address asynchronously.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultAnyValue> ReadAnyStringAsync(
    uint indexGroup,
    uint indexOffset,
```

```

    int len,
    Encoding encoding,
    CancellationToken cancel
)

```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
len	Type: System.Int32 The string length to be read.
encoding	Type: System.Text.Encoding The encoding.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAnyValue](#) [[▶ 1129](#)].

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [[▶ 1129](#)] parameter contains the read value ([Value](#) [[▶ 1185](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Reference

[IAdsAnyAccess Interface](#) [[▶ 804](#)]

[ReadAnyStringAsync Overload](#) [[▶ 826](#)]





[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

 [ResultAnyValue.Value Property](#) [[▶ 1131](#)]

6.2.30.1.5 IAdsAnyAccess.WriteAny Method

Overload List

	Name	Description
	WriteAny(UInt32, Object) [▶ 829]	Writes an object synchronously to an ADS device.
	WriteAny(UInt32, Object, .Int32.) [▶ 829]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAny(UInt32, UInt32, Object) [▶ 830]	Writes an object synchronously to an ADS device.
	WriteAny(UInt32, UInt32, Object, .Int32.) [▶ 830]	Writes an object synchronously to an ADS device.

Reference

[IAdsAnyAccess Interface](#) [[▶ 804](#)]

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.30.1.5.1 IAdsAnyAccess.WriteAny Method (UInt32, Object)

Writes an object synchronously to an ADS device.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void WriteAny(
    uint variableHandle,
    Object value
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
value	Type: System.Object Object to write to the ADS device.

Reference

[IAdsAnyAccess Interface \[▶ 804\]](#)

[WriteAny Overload \[▶ 828\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.30.1.5.2 IAdsAnyAccess.WriteAny Method (UInt32, Object, .Int32.)

Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void WriteAny(
    uint variableHandle,
    Object value,
    int[]? args
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
value	Type: System.Object Object to write to the ADS device.
args	Type: .System.Int32 . Additional arguments.

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference

[IAdsAnyAccess Interface](#) [[▶ 804](#)]

[WriteAny Overload](#) [[▶ 828](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.30.1.5.3 IAdsAnyAccess.WriteAny Method (UInt32, UInt32, Object)

Writes an object synchronously to an ADS device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
void WriteAny(
    uint indexGroup,
    uint indexOffset,
    Object value
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
value	Type: System.Object Object to write to the ADS device.

Reference

[IAdsAnyAccess Interface](#) [[▶ 804](#)]

[WriteAny Overload](#) [[▶ 828](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.30.1.5.4 IAdsAnyAccess.WriteAny Method (UInt32, UInt32, Object, .Int32.)

Writes an object synchronously to an ADS device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void WriteAny(
    uint indexGroup,
    uint indexOffset,
    Object value,
    int[]? args
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
value	Type: System.Object Object to write to the ADS device.
args	Type: .System.Int32 . Additional arguments.

Remarks

If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference


[IAdsAnyAccess Interface](#) [[▶ 804](#)]




[WriteAny Overload](#) [[▶ 828](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.30.1.6 IAdsAnyAccess.WriteAnyAsync Method

Overload List

	Name	Description
	WriteAnyAsync(UInt32, Object, CancellationToken) [▶ 832]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

	Name	Description
	WriteAnyAsync(UInt32, Object, .Int32., CancellationToken) [▶ 833]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAnyAsync(UInt32, UInt32, Object, CancellationToken) [▶ 834]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.
	WriteAnyAsync(UInt32, UInt32, Object, .Int32., CancellationToken) [▶ 834]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

Reference

[IAdsAnyAccess Interface](#) [▶ 804]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.30.1.6.1 IAdsAnyAccess.WriteAnyAsync Method (UInt32, Object, CancellationToken)

Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultWrite> WriteAnyAsync (
    uint variableHandle,
    Object value,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
value	Type: System.Object Object to write to the ADS device.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite](#) [▶ 1187].

A task that represents the asynchronous task operation. The result parameter [ResultWrite](#) [▶ 1187] of the write operation contains the [ErrorCode](#) [▶ 1120].

Reference

[IAdsAnyAccess Interface \[▶ 804\]](#)

[WriteAnyAsync Overload \[▶ 831\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.30.1.6.2 IAdsAnyAccess.WriteAnyAsync Method (UInt32, Object, .Int32., CancellationToken)

Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultWrite> WriteAnyAsync (
    uint variableHandle,
    Object value,
    int[]? args,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable.
value	Type: System.Object Object to write to the ADS device.
args	Type: .System.Int32. Additional arguments.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite \[▶ 1187\]](#).

A task that represents the asynchronous task operation. The result parameter [ResultWrite \[▶ 1187\]](#) of the write operation contains the [ErrorCode \[▶ 1120\]](#).

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927] .
string[]	
Array	

Reference

[IAdsAnyAccess Interface \[▶ 804\]](#)

[WriteAnyAsync Overload \[▸ 831\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.30.1.6.3 IAdsAnyAccess.WriteAnyAsync Method (UInt32, UInt32, Object, CancellationToken)

Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultWrite> WriteAnyAsync (
    uint indexGroup,
    uint indexOffset,
    Object value,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
value	Type: System.Object Object to write to the ADS device.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite \[▸ 1187\]](#).

A task that represents the asynchronous task operation. The result parameter [ResultWrite \[▸ 1187\]](#) of the write operation contains the [ErrorCode \[▸ 1120\]](#).

Reference

[IAdsAnyAccess Interface \[▸ 804\]](#)

[WriteAnyAsync Overload \[▸ 831\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.30.1.6.4 IAdsAnyAccess.WriteAnyAsync Method (UInt32, UInt32, Object, .Int32., CancellationToken)

Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultWrite> WriteAnyAsync (
    uint indexGroup,
    uint indexOffset,
    Object value,
    int[]? args,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
value	Type: System.Object Object to write to the ADS device.
args	Type: .System.Int32 . Additional arguments.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite](#) [[▶ 1187](#)].

A task that represents the asynchronous task operation. The result parameter [ResultWrite](#) [[▶ 1187](#)] of the write operation contains the [ErrorCode](#) [[▶ 1120](#)].

Remarks

As object types only primitive types are supported. If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args.

Type of value Parameter	Necessary Arguments (args)
string	args[0]: Number of characters in the string typed as FixedLengthZeroTerminated [▶ 2927].
string[]	
Array	

Reference

[IAdsAnyAccess Interface](#) [[▶ 804](#)]

[WriteAnyAsync Overload](#) [[▶ 831](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.31 IAdsConnectAddress Interface

Interface for method to connect the ADS client via AMS Address.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229











Syntax

C#




```
public interface IAdsConnectAddress : IAdsConnection,
    IConnection, IConnectionStateProvider, IAdsNotifications, IAdsSymbolicAccess, IAdsAnyAccess,
    IAdsHandle, IAdsReadWrite2, IAdsReadWrite, IAdsStateProvider, IAdsStateControl,
    IAdsSymbolChangedProvider, IAdsRpcInvoke
```








The IAdsConnectAddress type exposes the following members.





Properties




	Name	Description
	Address [▶ 895]	Gets the AmsAddress [▶ 752] of the ADS server. (Inherited from IAdsConnection [▶ 876].)
	ClientAddress [▶ 895]	Get the AmsAddress [▶ 752] of the ADS client. (Inherited from IAdsConnection [▶ 876].)
 	ConnectionState [▶ 90]	Gets the current Connection state of the IConnectionStateProvider [▶ 89] (Inherited from IConnectionStateProvider [▶ 89].)
	DefaultValueEncoding [▶ 81]	Gets the default value encoding. (Inherited from IConnection [▶ 79].)
	Id [▶ 81]	Gets the Connection Identifier . (Inherited from IConnection [▶ 79].)
	IsConnected [▶ 82]	Gets a value indicating whether the local ADS port was opened successfully. It does not indicate if the target port is available. Use the method ReadState to determine if the target port is available. (Inherited from IConnection [▶ 79].)
	IsLocal [▶ 896]	Gets a value indicating whether the ADS client is connected to a ADS Server on the local computer. (Inherited from IAdsConnection [▶ 876].)
	Session [▶ 82]	Gets the session that initiated this IConnection [▶ 79] (Inherited from IConnection [▶ 79].)
	Timeout [▶ 83]	Gets the timeout (in milliseconds) (Inherited from IConnection [▶ 79].)



Methods

















	Name	Description
	AddDeviceNotification(String, Int32, NotificationSettings, Object) [▶ 971]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object) [▶ 972]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationAsync(String, Int32, NotificationSettings,	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)











	Name	Description
	Object , CancellationToken) [▶ 974]	
	AddDeviceNotificati onAsync(UInt32, UInt32, Int32, NotificationSettings, Object, CancellationToken) [▶ 975]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificati onEx(String, NotificationSettings, Object, Type) [▶ 977]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificati onEx(String, NotificationSettings, Object, Type, .Int32.) [▶ 977]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificati onEx(UInt32, UInt32, NotificationSettings, Object, Type) [▶ 978]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificati onEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32.) [▶ 980]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive ('AnyType') types are allowed for the parameter type. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificati onExAsync(String, NotificationSettings, Object, Type, .Int32., CancellationToken) [▶ 981]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificati onExAsync(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., CancellationToken) [▶ 982]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)















	Name	Description
	CleanupSymbolTable [▶ 1070]	Clears the internal symbol / DataTypes cache. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	Close [▶ 83]	Closes this IConnection [▶ 79] (Inherited from IConnection [▶ 79].)
	Connect . [▶ 84]	(Re)Connects the IConnection [▶ 79] when disconnected. (Inherited from IConnection [▶ 79].)
	Connect(AmsAddress) [▶ 873]	Connects the target ADS Device.
	Connect(Int32) [▶ 873]	Connects to the local target ADS Device.
	Connect(AmsNetId, Int32) [▶ 874]	Connects to the target ADS Device.
	Connect(String, Int32) [▶ 874]	Connects to the target ADS Device.
	CreateVariableHandle [▶ 956]	Determines the Symbol handle by its instance path synchronously. (Inherited from IAdsHandle [▶ 954].)
	CreateVariableHandleAsync [▶ 956]	Determines the Symbol handle by its instance path asynchronously. (Inherited from IAdsHandle [▶ 954].)
	DeleteDeviceNotification [▶ 983]	Deletes a registered notification. (Inherited from IAdsNotifications [▶ 966].)
	DeleteDeviceNotificationAsync [▶ 984]	Deletes a registered notification asynchronously. (Inherited from IAdsNotifications [▶ 966].)
	DeleteVariableHandle [▶ 957]	Releases the specified symbol/variable handle synchronously. (Inherited from IAdsHandle [▶ 954].)
	DeleteVariableHandleAsync [▶ 958]	Releases the specified symbol/variable handle asynchronously. (Inherited from IAdsHandle [▶ 954].)
	Disconnect [▶ 84]	Disconnects this IConnection [▶ 79]. (Inherited from IConnection [▶ 79].)
 	InvokeRpcMethodAsync(String, String, Object, CancellationTokentoken) [▶ 1021]	Invokes the specified RPC Method asynchronously (Inherited from IAdsRpcInvoke [▶ 1012].)
 	InvokeRpcMethodAsync(IRpcCallableInstance, IRpcMethod, Object, AnyTypeSpecifier, AnyTypeSpecifier, CancellationTokentoken) [▶ 1024]	Invokes the specified RPC Method asynchronously (Inherited from IAdsRpcInvoke [▶ 1012].)















	Name	Description
 	InvokeRpcMethodAsync(String, String, Object, AnyTypeSpecifier, AnyTypeSpecifier, CancellationToken) [▶ 1022]	Invokes the specified RPC Method asynchronously (Inherited from IAdsRpcInvoke [▶ 1012].)
	Read(UInt32, Memory.Byte) [▶ 959]	Reads data synchronously from an ADS device and writes to the specified readBuffer. (Inherited from IAdsHandle [▶ 954].)
	Read(UInt32, UInt32, Memory.Byte) [▶ 1002]	Reads data synchronously from an ADS device and writes it to the given readBuffer (Inherited from IAdsReadWrite2 [▶ 1000].)
	ReadAny(UInt32, Type) [▶ 812]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny(UInt32, Type, Int32) [▶ 813]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny(UInt32, UInt32, Type) [▶ 815]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny(UInt32, UInt32, Type, Int32) [▶ 815]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32) [▶ 810]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32, Int32) [▶ 811]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32, UInt32) [▶ 812]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32, UInt32, Int32) [▶ 814]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt32, Type, CancellationToken) [▶ 819]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt32, Type, Int32, CancellationToken) [▶ 821]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt32, UInt32, Type, CancellationToken) [▶ 823]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)















	Name	Description
	ReadAnyAsync(UInt32, UInt32, Type, .Int32., CancellationToken) [▶ 823]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T.(UInt32, CancellationToken) [▶ 817]	Reads data asynchronously from an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T.(UInt32, .Int32., CancellationToken) [▶ 818]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T.(UInt32, UInt32, CancellationToken) [▶ 820]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T.(UInt32, UInt32, .Int32., CancellationToken) [▶ 822]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyString(UInt32, Int32, Encoding) [▶ 825]	Reads a string from the specified symbol/variable. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyString(UInt32, UInt32, Int32, Encoding) [▶ 825]	Reads as string from a specified address. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyStringAsync(UInt32, Int32, Encoding, CancellationToken) [▶ 827]	Reads a string asynchronously from the specified symbol/variable (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyStringAsync(UInt32, UInt32, Int32, Encoding, CancellationToken) [▶ 827]	Reads a string from a specified address asynchronously. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAsync(UInt32, Memory.Byte., CancellationToken) [▶ 959]	Reads the value data of the symbol asynchronously into the readBuffer. (Inherited from IAdsHandle [▶ 954].)








	Name	Description
	ReadAsync(UInt32, UInt32, Memory.Byte, CancellationToken) [▶ 996]	Reads the data asynchronously from specified IndexGroup/IndexOffset (Inherited from IAdsReadWrite [▶ 995].)
	ReadDataType [▶ 1070]	Call this method to obtain information about the specified data type. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadDataTypeAsync [▶ 1070]	Call this method to obtain information about the specified data type. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadDeviceInfo [▶ 914]	Reads the identification and version number of an ADS server. (Inherited from IAdsConnection [▶ 876].)
	ReadDeviceInfoAsync [▶ 914]	Reads the identification and version number of an ADS server. (Inherited from IAdsConnection [▶ 876].)
	ReadState [▶ 1058]	Reads the ADS status and the device status from an ADS server. (Inherited from IAdsStateProvider [▶ 1056].)
	ReadStateAsync [▶ 1058]	Reads the ADS status and the device status from an ADS server. (Inherited from IAdsStateProvider [▶ 1056].)
	ReadSymbol [▶ 1071]	Call this method to obtain information about the individual symbols (variables) in ADS devices. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadSymbolAsync [▶ 1072]	Call this method to obtain information about the individual symbols (variables) in ADS devices. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValue(ISymbol) [▶ 1073]	Reads the value of a symbol and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValue(String, Type) [▶ 1075]	Reads the value of a symbol and returns the value as object. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValue.T.(String) [▶ 1073]	Reads the value of a symbol and returns the value. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValueAsync(ISymbol, CancellationToken) [▶ 1077]	Reads the value of a symbol asynchronously and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValueAsync(String, Type, CancellationToken) [▶ 1079]	Reads the value of a symbol asynchronously. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValueAsync.T.(String, CancellationToken) [▶ 1076]	Reads the value of a symbol asynchronously. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadWrite(UInt32, Memory.Byte, ReadOnlyMemory.Byte) [▶ 960]	Writes data synchronously to an ADS device and then Reads data from that target. (Inherited from IAdsHandle [▶ 954].)





	Name	Description
	ReadWrite(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte.) [▶ 1002]	Writes data synchronously to an ADS device and then Reads data from this device into the readBuffer (Inherited from IAdsReadWrite2 [▶ 1000].)
	ReadWriteAsync(UInt32, Memory.Byte, ReadOnlyMemory.Byte, CancellationToken) [▶ 961]	ReadWrites value data asynchronously to/from the symbol represented by the variableHandle. (Inherited from IAdsHandle [▶ 954].)
	ReadWriteAsync(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte, CancellationToken) [▶ 997]	Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer (Inherited from IAdsReadWrite [▶ 995].)
	RegisterAdsStateChangedAsync [▶ 1059]	Registers for AdsStateChanged [▶ 1061] events as an asynchronous operation. (Inherited from IAdsStateProvider [▶ 1056].)
	RegisterSymbolVersionChanged [▶ 1064]	Registers the symbol version changed. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	RegisterSymbolVersionChangedAsync [▶ 1062]	Registers the symbol version changed asynchronously. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	TryAddDeviceNotification(String, Int32, NotificationSettings, Object, UInt32.) [▶ 985]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)
	TryAddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.) [▶ 986]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)
	TryAddDeviceNotificationEx(String, NotificationSettings, Object, Type, Int32, UInt32.) [▶ 988]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	TryAddDeviceNotificationEx(UInt32,	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)

	Name	Description
	UInt32 , NotificationSettings , Object , Type , .Int32 , UInt32 .) [▶ 989]	
	TryCreateVariableHandle [▶ 961]	Determines the Symbol handle by its instance path synchronously. (Inherited from IAdsHandle [▶ 954].)
	TryDeleteDeviceNotification [▶ 990]	Deletes a registered notification. (Inherited from IAdsNotifications [▶ 966].)
	TryDeleteVariableHandle [▶ 962]	Releases the specified symbol/variable handle synchronously. (Inherited from IAdsHandle [▶ 954].)
 	TryInvokeRpcMethod (String , String , .Object , Object .) [▶ 1026]	Invokes the specified RPC Method (Inherited from IAdsRpcInvoke [▶ 1012].)
 	TryInvokeRpcMethod (String , String , .Object , .Object , Object .) [▶ 1028]	Invokes the specified RPC Method (Inherited from IAdsRpcInvoke [▶ 1012].)
 	TryInvokeRpcMethod (IRpcCallableInstance , IRpcMethod , .Object , .AnyTypeSpecifier ,, AnyTypeSpecifier , .Object , Object .) [▶ 1031]	Invokes the rpc method. (Inherited from IAdsRpcInvoke [▶ 1012].)
 	TryInvokeRpcMethod (String , String , .Object , .AnyTypeSpecifier , AnyTypeSpecifier , .Object , Object .) [▶ 1030]	Invokes the rpc method. (Inherited from IAdsRpcInvoke [▶ 1012].)
	TryRead (UInt32 , Memory.Byte , Int32 .) [▶ 963]	Reads the value synchronously data of the symbol, that is represented by the variable handle into the readBuffer. (Inherited from IAdsHandle [▶ 954].)
	TryRead (UInt32 , UInt32 , Memory.Byte , Int32 .) [▶ 998]	Reads data synchronously from an ADS device and writes it to the given stream. (Inherited from IAdsReadWrite [▶ 995].)
	TryReadDataType [▶ 1079]	Call this method to obtain information about the specified data type. (Inherited from IAdsSymbolicAccess [▶ 1065].)












	Name	Description
	TryReadState [▶ 1059]	Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful. (Inherited from IAdsStateProvider [▶ 1056].)
	TryReadSymbol [▶ 1080]	Call this method to obtain information about the individual symbols (variables) in ADS devices. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadValue(ISymbol, Object.) [▶ 1082]	Reads the value of a symbol and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadValue(String, Type, Object.) [▶ 1083]	Reads the value of a symbol and returns the value as object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadValue.T. (ISymbol, T.) [▶ 1083]	Reads the value of a symbol and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadValue.T. (String, T.) [▶ 1081]	Reads the value of a symbol and returns the value as object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadWrite(UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.) [▶ 963]	ReadWrites value data synchronously to/from the symbol represented by the variableHandle. (Inherited from IAdsHandle [▶ 954].)
	TryReadWrite(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.) [▶ 998]	Writes data synchronously to an ADS device and reads data from that device. (Inherited from IAdsReadWrite [▶ 995].)
	TryWrite(UInt32, ReadOnlyMemory.Byte.) [▶ 964]	Writes the value data synchronously that is represented in the writeBuffer to the symbol with the specified variableHandle. (Inherited from IAdsHandle [▶ 954].)
	TryWrite(UInt32, UInt32, ReadOnlyMemory.Byte.) [▶ 999]	Writes data synchronously to an ADS device. (Inherited from IAdsReadWrite [▶ 995].)
	TryWriteControl(StationInfo) [▶ 1042]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	TryWriteControl(StationInfo, ReadOnlyMemory.Byte.) [▶ 1043]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	TryWriteValue(ISymbol, Object) [▶ 1086]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryWriteValue(String, Object) [▶ 1085]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)

	Name	Description
	TryWriteValue.T. (ISymbol , T) [▶ 1087]	Writes a value to the symbol. Strings and all primitive data types(UInt32 , Int32 , Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryWriteValue.T. (String , T) [▶ 1085]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	UnregisterAdsState ChangedAsync [▶ 1060]	Registers for AdsStateChanged [▶ 1061] events as an asynchronous operation. (Inherited from IAdsStateProvider [▶ 1056].)
	UnregisterSymbolV ersionChanged [▶ 1064]	Unregisters the symbol version changed. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	UnregisterSymbolV ersionChangedAsyn c [▶ 1063]	Unregisters the symbol version changed asynchronous. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	Write(UInt32, ReadOnlyMemory.B yte.) [▶ 964]	Writes data synchronously to an ADS device. (Inherited from IAdsHandle [▶ 954].)
	Write(UInt32, UInt32) [▶ 1003]	Triggers a 'Write' call to the ADS device at the specified address. (Inherited from IAdsReadWrite2 [▶ 1000].)
	Write(UInt32, UInt32, ReadOnlyMemory.B yte.) [▶ 1004]	Writes data synchronously to an ADS device. (Inherited from IAdsReadWrite2 [▶ 1000].)
	WriteAny(UInt32, Object) [▶ 829]	Writes an object synchronously to an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAny(UInt32, Object, .Int32.) [▶ 829]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAny(UInt32, UInt32, Object) [▶ 830]	Writes an object synchronously to an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAny(UInt32, UInt32, Object, .Int32.) [▶ 830]	Writes an object synchronously to an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt 32, Object, CancellationTok en) [▶ 832]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt 32, Object, .Int32, CancellationTok en) [▶ 833]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)




	Name	Description
	WriteAnyAsync(UInt32, UInt32, Object, CancellationToken) [▶ 834]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt32, UInt32, Object, .Int32., CancellationToken) [▶ 834]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAsync(UInt32, ReadOnlyMemory.Byte., CancellationToken) [▶ 965]	Writes the value data asynchronously that is represented by the writeBuffer to the symbol specified by the variableHandle. (Inherited from IAdsHandle [▶ 954].)
	WriteAsync(UInt32, UInt32, ReadOnlyMemory.Byte., CancellationToken) [▶ 1000]	Writes the data / Value asynchronously into the specified writeBuffer. (Inherited from IAdsReadWrite [▶ 995].)
	WriteControl(StateInfo) [▶ 1044]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	WriteControl(StateInfo, ReadOnlyMemory.Byte.) [▶ 1045]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	WriteControlAsync(AdsState, UInt16, CancellationToken) [▶ 1046]	Changes the ADS status and device status of the ADS server asynchronously. (Inherited from IAdsStateControl [▶ 1039].)
	WriteControlAsync(AdsState, UInt16, ReadOnlyMemory.Byte., CancellationToken) [▶ 1047]	Changes the ADS status and device status of the ADS server asynchronously. (Inherited from IAdsStateControl [▶ 1039].)
	WriteValue(ISymbol, Object) [▶ 1089]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValue(String, Object) [▶ 1088]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValue.T.(ISymbol, T) [▶ 1090]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)

	Name	Description
	WriteValue.T.(String, T) [▶ 1089]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValueAsync(ISymbol, Object, CancellationToken) [▶ 1092]	Writes a value to the symbol. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValueAsync.T.(ISymbol, T, CancellationToken) [▶ 1093]	Writes a value to the symbol. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValueAsync.T.(String, T, CancellationToken) [▶ 1091]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)

Events

	Name	Description
 	AdsNotification [▶ 991]	Occurs when the ADS device sends a notification to the client. (Inherited from IAdsNotifications [▶ 966].)
	AdsNotificationError [▶ 993]	Occurs when a exception has occurred during notification management. (Inherited from IAdsNotifications [▶ 966].)
 	AdsNotificationEx [▶ 993]	Occurs when the ADS devices sends a notification to the client. (Inherited from IAdsNotifications [▶ 966].)
	AdsStateChanged [▶ 1061]	Occurs when the AdsState of the target system has been changed. (Inherited from IAdsStateProvider [▶ 1056].)
 	AdsSumNotification [▶ 994]	Occurs when Notifications are send (bundled notifications) (Inherited from IAdsNotifications [▶ 966].)
	AdsSymbolVersionChanged [▶ 1065]	Occurs when the ADS Symbol Version changed. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
 	ConnectionStateChanged [▶ 91]	Occurs when connection status of the IConnectionStateProvider [▶ 89] has been changed. (Inherited from IConnectionStateProvider [▶ 89].)


Extension Methods

	Name	Description
	PollAdsState(IObservable.Unit.) [▶ 3294]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146] s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
 	PollAdsState(IObservable.Unit.) [▶ 1256]	Overloaded. Gets an observable sequence of AdsState [▶ 729] s via Polling. (Defined by AdsClientExtensions [▶ 1249].)

	Name	Description
 	PollAdsState(TimeSpan) [▸ 3295]	Overloaded. Gets an observable sequence of ResultReadAdsState [▸ 1146]s via Polling. (Defined by ConnectionStateExtension [▸ 3292].)
 	PollAdsState(TimeSpan) [▸ 1257]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
 	PollAdsState2(IObservable.Unit.) [▸ 1271]	Overloaded. Gets an observable sequence of ResultReadAdsState [▸ 1146]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
 	PollAdsState2(TimeSpan) [▸ 1272]	Overloaded. Gets an observable sequence of ResultReadAdsState [▸ 1146]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
 	PollAdsState2Async(IObservable.Unit., Cancellation.Token) [▸ 1274]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
 	PollAdsState2Async(TimeSpan, Cancellation.Token) [▸ 1275]	Overloaded. Gets an observable sequence of ResultReadAdsState [▸ 1146]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
 	PollAdsStateAsync(IObservable.Unit., Cancellation.Token) [▸ 3296]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by ConnectionStateExtension [▸ 3292].)
 	PollAdsStateAsync(IObservable.Unit., Cancellation.Token) [▸ 1259]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
 	PollAdsStateAsync(TimeSpan, Cancellation.Token) [▸ 3298]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by ConnectionStateExtension [▸ 3292].)
 	PollAdsStateAsync(TimeSpan, Cancellation.Token) [▸ 1260]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
 	PollDeviceState(IObservable.Unit.) [▸ 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▸ 1157]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
 	PollDeviceState(TimeSpan) [▸ 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▸ 1157]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)









	Name	Description
	PollDeviceStateAsync(IObservable.Unit, CancellationToken) [▶ 1278]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157] s via Polling. (Defined by AdsClientExtensions [▶ 1249] .)
	PollDeviceStateAsync(TimeSpan, CancellationToken) [▶ 1279]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157] s via Polling. (Defined by AdsClientExtensions [▶ 1249] .)
	PollSystemServiceState [▶ 3282]	Polls the state of the system service. (Defined by SystemServiceExtension [▶ 3281] .)
	PollSystemServiceStateAsync [▶ 3283]	Polls the system service state asynchronously (Defined by SystemServiceExtension [▶ 3281] .)
	PollValues(ISymbol, Type, IObservable.Unit) [▶ 1310]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(ISymbol, Type, TimeSpan) [▶ 1311]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(String, Type, IObservable.Unit) [▶ 1300]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(String, Type, TimeSpan) [▶ 1301]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(ISymbol, Type, .Int32, TimeSpan) [▶ 1316]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(ISymbol, Type, IObservable.Unit, Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(ISymbol, Type, TimeSpan, Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(String, Type, .Int32, TimeSpan) [▶ 1304]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(String, Type, IObservable.Unit, Func.Exception, Object.) [▶ 1305]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281] .)

	Name	Description
	PollValues(String, Type, TimeSpan, Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32., IObservable.Unit, Func.Exception, Object.) [▶ 1321]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32., TimeSpan, Func.Exception, Object.) [▶ 1322]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32., IObservable.Unit, Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32., TimeSpan, Func.Exception, Object.) [▶ 1307]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, IObservable.Unit.) [▶ 1308]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, TimeSpan) [▶ 1309]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, IObservable.Unit.) [▶ 1294]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, TimeSpan) [▶ 1295]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, IObservable.Unit., Func.Exception, T.) [▶ 1314]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, TimeSpan, Func.Exception, T.) [▶ 1315]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, .Int32., IObservable.Unit.) [▶ 1312]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	PollValues.T. (ISymbol , .Int32 , TimeSpan) [▶ 1313]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, IObservable.Unit, Func.Exception, T.) [▶ 1298]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, TimeSpan, Func.Exception, T.) [▶ 1299]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32 , IObservable.Unit .) [▶ 1296]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32 , TimeSpan) [▶ 1297]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol , .Int32 , IObservable.Unit , Func.Exception, T.) [▶ 1318]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol , .Int32 , TimeSpan , Func.Exception, T.) [▶ 1319]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32 , IObservable.Unit , Func.Exception, T.) [▶ 1302]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32 , TimeSpan , Func.Exception, T.) [▶ 1303]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues2.T. (ISymbol , .Int32 , IObservable.Unit .) [▶ 1330]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues2.T. (String , .Int32 , IObservable.Unit .) [▶ 1329]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	ReadSysServState [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (Defined by SystemServiceExtension [▶ 3281].)

	Name	Description
	ReadSysServStateAsync [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (asynchronous) (Defined by SystemServiceExtension [▶ 3281].)
	ReadWithFallback(UInt32, UInt32, Memory.Byte., UInt32, Boolean.) [▶ 1219]	Overloaded. Ads Read with fallback. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallback(UInt32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., Boolean.) [▶ 1220]	Overloaded. Ads Read with Fallback. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallbackAsync(UInt32, UInt32, UInt32, Memory.Byte., CancellationToken) [▶ 1221]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallbackAsync(UInt32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., CancellationToken) [▶ 1222]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedRead(UInt32, UInt32, Memory.Byte., Int32, TimeSpan) [▶ 1224]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedRead(UInt32, UInt32, Memory.Byte., Int32, TimeSpan, Func.ResultRead, Boolean.) [▶ 1225]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedReadAsync(UInt32, UInt32, Memory.Byte., Int32, TimeSpan, CancellationToken) [▶ 1226]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)


	Name	Description
	RepeatedReadAsync (UInt32, UInt32, Memory.Byte., Int32, TimeSpan, Func.ResultRead, Boolean., CancellationToken) [1227]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [1215].)
	RestartTwinCATAsync [3285]	Sends a SystemService Restart to the target system and waits until the system has finished the state changes (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [3281].)
	SetAdsState [3299]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [3292].)
	SetAdsStateAsync [3300]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [3292].)
	WaitUntilRestarted [3286]	Waits until the Restart is detected on the client (SystemService, Port 10000) (Defined by SystemServiceExtension [3281].)
	WaitUntilRestartedAsync [3287]	Waits until the SystemService Restart is detected on the client (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [3281].)
 	WhenAdsStateChanges [1261]	Gets an observable sequence of AdsState [729]s. (Defined by AdsClientExtensions [1249].)
	WhenNotification(ISymbol) [1263]	Overloaded. Gets an observable sequence of Notification [1100]s. (Defined by AdsClientExtensions [1249].)
 	WhenNotification(ISymbolCollection) [1264]	Overloaded. Gets an observable sequence of Notification [1100] objects. (Defined by AdsClientExtensions [1249].)
 	WhenNotification(IList.ISymbol, NotificationSettings) [1268]	Overloaded. Gets an observable sequence of Notification [1100] objects. (Defined by AdsClientExtensions [1249].)
	WhenNotification(ISymbol, NotificationSettings) [1265]	Overloaded. Gets an observable sequence of SymbolValueNotification [1331]s. (Defined by AdsClientExtensions [1249].)
 	WhenNotification(STRING, Type, NotificationSettings) [1324]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [1281].)
 	WhenNotification.T.(String, NotificationSettings) [1323]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [1281].)

	Name	Description
	WhenSymbolVersionChanges [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
	WhenSymbolVersionChanges(IScheduler) [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenValueChanged [▶ 1345]	Observable sequence of Values driven by ADS Notifications on the specified symbol. (Defined by ValueSymbolExtensions [▶ 1333].)
 	WriteValues.T.(String, IObservable.T.) [▶ 1326]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)
 	WriteValues.T.(String, IObservable.T., Action.Exception.) [▶ 1327]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)

Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]








Also see about this

-  [AdsClientExtensions.WhenNotification Method \(IAdsConnection, ISymbolCollection, NotificationSettings\)](#) [[▶ 1266](#)]

6.2.31.1 IAdsConnectAddress Properties

The [IAdsConnectAddress](#) [[▶ 835](#)] type exposes the following members.

Properties

	Name	Description
	Address [▶ 895]	Gets the AmsAddress [▶ 752] of the ADS server. (Inherited from IAdsConnection [▶ 876].)
	ClientAddress [▶ 895]	Get the AmsAddress [▶ 752] of the ADS client. (Inherited from IAdsConnection [▶ 876].)
 	ConnectionState [▶ 90]	Gets the current Connection state of the IConnectionStateProvider [▶ 89] (Inherited from IConnectionStateProvider [▶ 89].)
	DefaultValueEncoding [▶ 81]	Gets the default value encoding. (Inherited from IConnection [▶ 79].)
	Id [▶ 81]	Gets the Connection Identifier . (Inherited from IConnection [▶ 79].)
	IsConnected [▶ 82]	Gets a value indicating whether the local ADS port was opened successfully. It does not indicate if the target port is available. Use the method ReadState to determine if the target port is available. (Inherited from IConnection [▶ 79].)

	Name	Description
	IsLocal [▶ 896]	Gets a value indicating whether the ADS client is connected to a ADS Server on the local computer. (Inherited from IAdsConnection [▶ 876].)
	Session [▶ 82]	Gets the session that initiated this IConnection [▶ 79] (Inherited from IConnection [▶ 79].)
	Timeout [▶ 83]	Gets the timeout (in milliseconds) (Inherited from IConnection [▶ 79].)

Reference

[IAdsConnectAddress Interface](#) [[▶ 835](#)]














[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.31.2 IAdsConnectAddress Methods




The [IAdsConnectAddress](#) [[▶ 835](#)] type exposes the following members.
















Methods

	Name	Description
	AddDeviceNotification(String, Int32, NotificationSettings, Object) [▶ 971]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object) [▶ 972]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationAsync(String, Int32, NotificationSettings, Object, CancellationToken) [▶ 974]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationAsync(UInt32, UInt32, Int32, NotificationSettings, Object, CancellationToken) [▶ 975]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type) [▶ 977]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)















	Name	Description
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32.) [▶ 977]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type) [▶ 978]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32.) [▶ 980]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive ('AnyType') types are allowed for the parameter type. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationExAsync(String, NotificationSettings, Object, Type, .Int32., CancellationToken) [▶ 981]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationExAsync(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., CancellationToken) [▶ 982]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	CleanupSymbolTable [▶ 1070]	Clears the internal symbol / DataTypes cache. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	Close [▶ 83]	Closes this IConnection [▶ 79] (Inherited from IConnection [▶ 79].)
	Connect. [▶ 84]	(Re)Connects the IConnection [▶ 79] when disconnected. (Inherited from IConnection [▶ 79].)
	Connect(AmsAddress) [▶ 873]	Connects the target ADS Device.
	Connect(Int32) [▶ 873]	Connects to the local target ADS Device.
	Connect(AmsNetId, Int32) [▶ 874]	Connects to the target ADS Device.
	Connect(String, Int32) [▶ 874]	Connects to the target ADS Device.
	CreateVariableHandle [▶ 956]	Determines the Symbol handle by its instance path synchronously. (Inherited from IAdsHandle [▶ 954].)














	Name	Description
	CreateVariableHandleAsync [▶ 956]	Determines the Symbol handle by its instance path asynchronously. (Inherited from IAdsHandle [▶ 954].)
	DeleteDeviceNotification [▶ 983]	Deletes a registered notification. (Inherited from IAdsNotifications [▶ 966].)
	DeleteDeviceNotificationAsync [▶ 984]	Deletes a registered notification asynchronously. (Inherited from IAdsNotifications [▶ 966].)
	DeleteVariableHandle [▶ 957]	Releases the specified symbol/variable handle synchronously. (Inherited from IAdsHandle [▶ 954].)
	DeleteVariableHandleAsync [▶ 958]	Releases the specified symbol/variable handle asynchronously. (Inherited from IAdsHandle [▶ 954].)
	Disconnect [▶ 84]	Disconnects this IConnection [▶ 79]. (Inherited from IConnection [▶ 79].)
 	InvokeRpcMethodAsync(String, String, Object, CancellationToken) [▶ 1021]	Invokes the specified RPC Method asynchronously (Inherited from IAdsRpcInvoke [▶ 1012].)
 	InvokeRpcMethodAsync(IRpcCallableObject, IRpcMethod, Object, AnyTypeSpecifier, AnyTypeSpecifier, CancellationToken) [▶ 1024]	Invokes the specified RPC Method asynchronously (Inherited from IAdsRpcInvoke [▶ 1012].)
 	InvokeRpcMethodAsync(String, String, Object, AnyTypeSpecifier, AnyTypeSpecifier, CancellationToken) [▶ 1022]	Invokes the specified RPC Method asynchronously (Inherited from IAdsRpcInvoke [▶ 1012].)
	Read(UInt32, Memory.Byte) [▶ 959]	Reads data synchronously from an ADS device and writes to the specified readBuffer. (Inherited from IAdsHandle [▶ 954].)
	Read(UInt32, UInt32, Memory.Byte) [▶ 1002]	Reads data synchronously from an ADS device and writes it to the given readBuffer (Inherited from IAdsReadWrite2 [▶ 1000].)
	ReadAny(UInt32, Type) [▶ 812]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny(UInt32, Type, Int32) [▶ 813]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)














	Name	Description
	ReadAny(UInt32, UInt32, Type) [▶ 815]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny(UInt32, UInt32, Type, .Int32.) [▶ 815]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32) [▶ 810]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32, .Int32.) [▶ 811]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32, UInt32) [▶ 812]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32, UInt32, .Int32.) [▶ 814]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt32, Type, CancellationToken) [▶ 819]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt32, Type, .Int32., CancellationToken) [▶ 821]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt32, UInt32, Type, CancellationToken) [▶ 823]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt32, UInt32, Type, .Int32., CancellationToken) [▶ 823]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T.(UInt32, CancellationToken) [▶ 817]	Reads data asynchronously from an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T.(UInt32, .Int32., CancellationToken) [▶ 818]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T.(UInt32, UInt32, CancellationToken) [▶ 820]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)












	Name	Description
	ReadAnyAsync.T. (UInt32 , UInt32 , Int32 , CancellationToken) [▶ 822]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyString (UInt32 , Int32 , Encoding) [▶ 825]	Reads a string from the specified symbol/variable. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyString (UInt32 , UInt32 , Int32 , Encoding) [▶ 825]	Reads as string from a specified address. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyStringAsync (UInt32 , Int32 , Encoding , CancellationToken) [▶ 827]	Reads a string asynchronously from the specified symbol/variable (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyStringAsync (UInt32 , UInt32 , Int32 , Encoding , CancellationToken) [▶ 827]	Reads a string from a specified address asynchronously. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAsync (UInt32 , Memory.Byte , CancellationToken) [▶ 959]	Reads the value data of the symbol asynchronously into the readBuffer. (Inherited from IAdsHandle [▶ 954].)
	ReadAsync (UInt32 , UInt32 , Memory.Byte , CancellationToken) [▶ 996]	Reads the data asynchronously from specified IndexGroup/IndexOffset (Inherited from IAdsReadWrite [▶ 995].)
	ReadDataType [▶ 1070]	Call this method to obtain information about the specified data type. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadDataTypeAsync [▶ 1070]	Call this method to obtain information about the specified data type. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadDeviceInfo [▶ 914]	Reads the identification and version number of an ADS server. (Inherited from IAdsConnection [▶ 876].)
	ReadDeviceInfoAsync [▶ 914]	Reads the identification and version number of an ADS server. (Inherited from IAdsConnection [▶ 876].)
	ReadState [▶ 1058]	Reads the ADS status and the device status from an ADS server. (Inherited from IAdsStateProvider [▶ 1056].)
	ReadStateAsync [▶ 1058]	Reads the ADS status and the device status from an ADS server. (Inherited from IAdsStateProvider [▶ 1056].)
	ReadSymbol [▶ 1071]	Call this method to obtain information about the individual symbols (variables) in ADS devices. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadSymbolAsync [▶ 1072]	Call this method to obtain information about the individual symbols (variables) in ADS devices. (Inherited from IAdsSymbolicAccess [▶ 1065].)












	Name	Description
	ReadValue(ISymbol) [▶ 1073]	Reads the value of a symbol and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValue(String, Type) [▶ 1075]	Reads the value of a symbol and returns the value as object. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValue.T.(String) [▶ 1073]	Reads the value of a symbol and returns the value. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValueAsync(ISymbol, CancellationToken) [▶ 1077]	Reads the value of a symbol asynchronously and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValueAsync(String, Type, CancellationToken) [▶ 1079]	Reads the value of a symbol asynchronously. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValueAsync.T.(String, CancellationToken) [▶ 1076]	Reads the value of a symbol asynchronously. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadWrite(UInt32, Memory.Byte, ReadOnlyMemory.Byte) [▶ 960]	Writes data synchronously to an ADS device and then Reads data from that target. (Inherited from IAdsHandle [▶ 954].)
	ReadWrite(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte) [▶ 1002]	Writes data synchronously to an ADS device and then Reads data from this device into the readBuffer (Inherited from IAdsReadWrite2 [▶ 1000].)
	ReadWriteAsync(UInt32, Memory.Byte, ReadOnlyMemory.Byte, CancellationToken) [▶ 961]	ReadWrites value data asynchronously to/from the symbol represented by the variableHandle. (Inherited from IAdsHandle [▶ 954].)
	ReadWriteAsync(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte, CancellationToken) [▶ 997]	Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer (Inherited from IAdsReadWrite [▶ 995].)
	RegisterAdsStateChangedAsync [▶ 1059]	Registers for AdsStateChanged [▶ 1061] events as an asynchronous operation. (Inherited from IAdsStateProvider [▶ 1056].)
	RegisterSymbolVersionChanged [▶ 1064]	Registers the symbol version changed. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)

	Name	Description
	RegisterSymbolVersionChangedAsynchronous [1062]	Registers the symbol version changed asynchronously. (Inherited from IAdsSymbolChangedProvider [1061].)
	TryAddDeviceNotification (String, Int32, NotificationSettings, Object, UInt32.) [985]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [991] event. (Inherited from IAdsNotifications [966].)
	TryAddDeviceNotification (UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.) [986]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [991] event. (Inherited from IAdsNotifications [966].)
	TryAddDeviceNotificationEx (String, NotificationSettings, Object, Type, .Int32, UInt32.) [988]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [993] event. (Inherited from IAdsNotifications [966].)
	TryAddDeviceNotificationEx (UInt32, UInt32, NotificationSettings, Object, Type, .Int32, UInt32.) [989]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [993] event. (Inherited from IAdsNotifications [966].)
	TryCreateVariableHandle [961]	Determines the Symbol handle by its instance path synchronously. (Inherited from IAdsHandle [954].)
	TryDeleteDeviceNotification [990]	Deletes a registered notification. (Inherited from IAdsNotifications [966].)
	TryDeleteVariableHandle [962]	Releases the specified symbol/variable handle synchronously. (Inherited from IAdsHandle [954].)
 	TryInvokeRpcMethod (String, String, .Object, Object.) [1026]	Invokes the specified RPC Method (Inherited from IAdsRpcInvoke [1012].)
 	TryInvokeRpcMethod (String, String, .Object, .Object, Object.) [1028]	Invokes the specified RPC Method (Inherited from IAdsRpcInvoke [1012].)
 	TryInvokeRpcMethod (IRpcCallableInstance)	Invokes the rpc method. (Inherited from IAdsRpcInvoke [1012].)




















	Name	Description
	ce , IRpcMethod , .Object , .AnyTypeSpecifier “ AnyTypeSpecifier , .Object , Object .” [▶ 1031]	
 	TryInvokeRpcMethod(String, .Object, .AnyTypeSpecifier, .AnyTypeSpecifier, .Object, Object.) [▶ 1030]	Invokes the rpc method. (Inherited from IAdsRpcInvoke [▶ 1012] .)
	TryRead(UInt32, Memory.Byte, Int32.) [▶ 963]	Reads the value synchronously data of the symbol, that is represented by the variable handle into the readBuffer. (Inherited from IAdsHandle [▶ 954] .)
	TryRead(UInt32, UInt32, Memory.Byte, Int32.) [▶ 998]	Reads data synchronously from an ADS device and writes it to the given stream. (Inherited from IAdsReadWrite [▶ 995] .)
	TryReadDataType [▶ 1079]	Call this method to obtain information about the specified data type. (Inherited from IAdsSymbolicAccess [▶ 1065] .)
	TryReadState [▶ 1059]	Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful. (Inherited from IAdsStateProvider [▶ 1056] .)
	TryReadSymbol [▶ 1080]	Call this method to obtain information about the individual symbols (variables) in ADS devices. (Inherited from IAdsSymbolicAccess [▶ 1065] .)
	TryReadValue(ISymbol, Object.) [▶ 1082]	Reads the value of a symbol and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065] .)
	TryReadValue(String, Type, Object.) [▶ 1083]	Reads the value of a symbol and returns the value as object. (Inherited from IAdsSymbolicAccess [▶ 1065] .)
	TryReadValue.T.(ISymbol, T.) [▶ 1083]	Reads the value of a symbol and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065] .)
	TryReadValue.T.(String, T.) [▶ 1081]	Reads the value of a symbol and returns the value as object. (Inherited from IAdsSymbolicAccess [▶ 1065] .)
	TryReadWrite(UInt32, Memory.Byte, ReadOnlyMemory.Byte, Int32.) [▶ 963]	ReadWrites value data synchronously to/from the symbol represented by the variableHandle. (Inherited from IAdsHandle [▶ 954] .)
	TryReadWrite(UInt32, UInt32,	Writes data synchronously to an ADS device and reads data from that device. (Inherited from IAdsReadWrite [▶ 995] .)

	Name	Description
	Memory.Byte. , ReadOnlyMemory.B yte., Int32.) [▶ 998]	
	TryWrite(UInt32, ReadOnlyMemory.B yte.) [▶ 964]	Writes the value data synchronously that is represented in the writeBuffer to the symbol with the specified variableHandle. (Inherited from IAdsHandle [▶ 954].)
	TryWrite(UInt32, UInt32, ReadOnlyMemory.B yte.) [▶ 999]	Writes data synchronously to an ADS device. (Inherited from IAdsReadWrite [▶ 995].)
	TryWriteControl(Sta telInfo) [▶ 1042]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	TryWriteControl(Sta telInfo, ReadOnlyMemory.B yte.) [▶ 1043]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	TryWriteValue(ISym bol, Object) [▶ 1086]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryWriteValue(Strin g, Object) [▶ 1085]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryWriteValue.T. (ISymbol, T) [▶ 1087]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryWriteValue.T. (String, T) [▶ 1085]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	UnregisterAdsState ChangedAsync [▶ 1060]	Registers for AdsStateChanged [▶ 1061] events as an asynchronous operation. (Inherited from IAdsStateProvider [▶ 1056].)
	UnregisterSymbolV ersionChanged [▶ 1064]	Unregisters the symbol version changed. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	UnregisterSymbolV ersionChangedAsyn c [▶ 1063]	Unregisters the symbol version changed asynchronous. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	Write(UInt32, ReadOnlyMemory.B yte.) [▶ 964]	Writes data synchronously to an ADS device. (Inherited from IAdsHandle [▶ 954].)
	Write(UInt32, UInt32) [▶ 1003]	Triggers a 'Write' call to the ADS device at the specified address. (Inherited from IAdsReadWrite2 [▶ 1000].)













	Name	Description
	Write(UInt32, UInt32, ReadOnlyMemory.BYTE) [▶ 1004]	Writes data synchronously to an ADS device. (Inherited from IAdsReadWrite2 [▶ 1000].)
	WriteAny(UInt32, Object) [▶ 829]	Writes an object synchronously to an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAny(UInt32, Object, .Int32.) [▶ 829]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAny(UInt32, UInt32, Object) [▶ 830]	Writes an object synchronously to an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAny(UInt32, UInt32, Object, .Int32.) [▶ 830]	Writes an object synchronously to an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt32, Object, CancellationToken) [▶ 832]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt32, Object, .Int32., CancellationToken) [▶ 833]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt32, UInt32, Object, CancellationToken) [▶ 834]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt32, UInt32, Object, .Int32., CancellationToken) [▶ 834]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAsync(UInt32, ReadOnlyMemory.BYTE, CancellationToken) [▶ 965]	Writes the value data asynchronously that is represented by the writeBuffer to the symbol specified by the variableHandle. (Inherited from IAdsHandle [▶ 954].)
	WriteAsync(UInt32, UInt32, ReadOnlyMemory.BYTE, CancellationToken) [▶ 1000]	Writes the data / Value asynchronously into the specified writeBuffer. (Inherited from IAdsReadWrite [▶ 995].)















	Name	Description
	WriteControl(StateInfo) [► 1044]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [► 1039].)
	WriteControl(StateInfo, ReadOnlyMemory.Byte) [► 1045]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [► 1039].)
	WriteControlAsync(AdsState, UInt16, CancellationToken) [► 1046]	Changes the ADS status and device status of the ADS server asynchronously. (Inherited from IAdsStateControl [► 1039].)
	WriteControlAsync(AdsState, UInt16, ReadOnlyMemory.Byte, CancellationToken) [► 1047]	Changes the ADS status and device status of the ADS server asynchronously. (Inherited from IAdsStateControl [► 1039].)
	WriteValue(ISymbol, Object) [► 1089]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [► 1065].)
	WriteValue(String, Object) [► 1088]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [► 1065].)
	WriteValue.T.(ISymbol, T) [► 1090]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [► 1065].)
	WriteValue.T.(String, T) [► 1089]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [► 1065].)
	WriteValueAsync(ISymbol, Object, CancellationToken) [► 1092]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [► 1065].)
	WriteValueAsync.T.(ISymbol, T, CancellationToken) [► 1093]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [► 1065].)
	WriteValueAsync.T.(String, T, CancellationToken) [► 1091]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [► 1065].)

Extension Methods















	Name	Description
	PollAdsState(IObservable.Unit.) [▶ 3294]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
 	PollAdsState(IObservable.Unit.) [▶ 1256]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsState(TimeSpan) [▶ 3295]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
 	PollAdsState(TimeSpan) [▶ 1257]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState2(IObservable.Unit.) [▶ 1271]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsState2(TimeSpan) [▶ 1272]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState2Async(IObservable.Unit., Cancellation.Token) [▶ 1274]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsState2Async(TimeSpan, Cancellation.Token) [▶ 1275]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsStateAsync(IObservable.Unit., Cancellation.Token) [▶ 3296]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
 	PollAdsStateAsync(IObservable.Unit., Cancellation.Token) [▶ 1259]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsStateAsync(TimeSpan, Cancellation.Token) [▶ 3298]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)





	Name	Description
 	PollAdsStateAsync(TimeSpan, Cancellation-Token) [▶ 1260]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollDeviceState(IObservable.Unit.) [▶ 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollDeviceState(TimeSpan) [▶ 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollDeviceStateAsync(IObservable.Unit., Cancellation-Token) [▶ 1278]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollDeviceStateAsync(TimeSpan, Cancellation-Token) [▶ 1279]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollSystemServiceState [▶ 3282]	Polls the state of the system service. (Defined by SystemServiceExtension [▶ 3281].)
	PollSystemServiceStateAsync [▶ 3283]	Polls the system service state asynchronously (Defined by SystemServiceExtension [▶ 3281].)
	PollValues(ISymbol, Type, IObservable.Unit.) [▶ 1310]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, TimeSpan) [▶ 1311]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, IObservable.Unit.) [▶ 1300]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, TimeSpan) [▶ 1301]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32., TimeSpan) [▶ 1316]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, IObservable.Unit., Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	PollValues(ISymbol, Type, TimeSpan, Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32, TimeSpan) [▶ 1304]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, IObservable.Unit, Func.Exception, Object.) [▶ 1305]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, TimeSpan, Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32, IObservable.Unit, Func.Exception, Object.) [▶ 1321]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32, TimeSpan, Func.Exception, Object.) [▶ 1322]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32, IObservable.Unit, Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32, TimeSpan, Func.Exception, Object.) [▶ 1307]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, IObservable.Unit) [▶ 1308]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, TimeSpan) [▶ 1309]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, IObservable.Unit) [▶ 1294]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, TimeSpan) [▶ 1295]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	PollValues.T. (ISymbol , IObservable.Unit. , Func.Exception, T.) [▶ 1314]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol , TimeSpan , Func.Exception, T.) [▶ 1315]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol , .Int32. , IObservable.Unit.) [▶ 1312]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	PollValues.T. (ISymbol , .Int32. , TimeSpan) [▶ 1313]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String. IObservable.Unit. , Func.Exception, T.) [▶ 1298]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String. TimeSpan , Func.Exception, T.) [▶ 1299]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32. , IObservable.Unit.) [▶ 1296]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	PollValues.T. (String , .Int32. , TimeSpan) [▶ 1297]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol , .Int32. , IObservable.Unit. , Func.Exception, T.) [▶ 1318]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	PollValues.T. (ISymbol , .Int32. , TimeSpan , Func.Exception, T.) [▶ 1319]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32. , IObservable.Unit. , Func.Exception, T.) [▶ 1302]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	PollValues.T. (String , .Int32 , TimeSpan , Func.Exception , T .) [▶ 1303]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues2.T. (ISymbol , .Int32 , IObservable.Unit .) [▶ 1330]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues2.T. (String , .Int32 , IObservable.Unit .) [▶ 1329]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	ReadSysServState [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (Defined by SystemServiceExtension [▶ 3281].)
	ReadSysServStateAs ync [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (asynchronous) (Defined by SystemServiceExtension [▶ 3281].)
	ReadWithFallback(U Int32, UInt32, Memory.Byte, UInt32, Boolean.) [▶ 1219]	Overloaded. Ads Read with fallback. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallback(U Int32, UInt32, UInt32, Memory.Byte, Func.ResultRead, Boolean., Boolean.) [▶ 1220]	Overloaded. Ads Read with Fallback. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallbackAs ync(UInt32, UInt32, UInt32, Memory.Byte, CancellationToken) [▶ 1221]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallbackAs ync(UInt32, UInt32, UInt32, Memory.Byte, Func.ResultRead, Boolean., CancellationToken) [▶ 1222]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)

	Name	Description
	RepeatedRead(UInt32, UInt32, Memory.Byte, Int32, TimeSpan) [▸ 1224]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [▸ 1215] .)
	RepeatedRead(UInt32, UInt32, Memory.Byte, Int32, TimeSpan, Func.ResultRead, Boolean.) [▸ 1225]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [▸ 1215] .)
	RepeatedReadAsync(UInt32, UInt32, Memory.Byte, Int32, TimeSpan, CancellationToken) [▸ 1226]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [▸ 1215] .)
	RepeatedReadAsync(UInt32, UInt32, Memory.Byte, Int32, TimeSpan, Func.ResultRead, Boolean, CancellationToken) [▸ 1227]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [▸ 1215] .)
	RestartTwinCATAsync [▸ 3285]	Sends a SystemService Restart to the target system and waits until the system has finished the state changes (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [▸ 3281] .)
	SetAdsState [▸ 3299]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [▸ 3292] .)
	SetAdsStateAsync [▸ 3300]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [▸ 3292] .)
	WaitUntilRestarted [▸ 3286]	Waits until the Restart is detected on the client (SystemService, Port 10000) (Defined by SystemServiceExtension [▸ 3281] .)
	WaitUntilRestartedAsync [▸ 3287]	Waits until the SystemService Restart is detected on the client (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [▸ 3281] .)
 	WhenAdsStateChanges [▸ 1261]	Gets an observable sequence of AdsState [▸ 729] s. (Defined by AdsClientExtensions [▸ 1249] .)
	WhenNotification(ISymbol) [▸ 1263]	Overloaded. Gets an observable sequence of Notification [▸ 1100] s. (Defined by AdsClientExtensions [▸ 1249] .)
 	WhenNotification(ISymbolCollection) [▸ 1264]	Overloaded. Gets an observable sequence of Notification [▸ 1100] objects. (Defined by AdsClientExtensions [▸ 1249] .)


	Name	Description
 	WhenNotification(IList.ISymbol, NotificationSettings) [▶ 1268]	Overloaded. Gets an observable sequence of Notification [▶ 1100] objects. (Defined by AdsClientExtensions [▶ 1249].)
	WhenNotification(ISymbol, NotificationSettings) [▶ 1265]	Overloaded. Gets an observable sequence of SymbolValueNotification [▶ 1331]s. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenNotification(String, Type, NotificationSettings) [▶ 1324]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [▶ 1281].)
 	WhenNotification.T.(String, NotificationSettings) [▶ 1323]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [▶ 1281].)
	WhenSymbolVersionChanges. [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
	WhenSymbolVersionChanges(IScheduler) [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenValueChanged [▶ 1345]	Observable sequence of Values driven by ADS Notifications on the specified symbol. (Defined by ValueSymbolExtensions [▶ 1333].)
 	WriteValues.T.(String, IObservable.T.) [▶ 1326]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)
 	WriteValues.T.(String, IObservable.T., Action.Exception.) [▶ 1327]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)

Reference

[IAdsConnectAddress Interface](#) [▶ 835]






[TwinCAT.Ads Namespace](#) [▶ 179]

Also see about this

-  [AdsClientExtensions.WhenNotification Method \(IAdsConnection, ISymbolCollection, NotificationSettings\)](#) [▶ 1266]

6.2.31.2.1 IAdsConnectAddress.Connect Method

Overload List

	Name	Description
	Connect. [▶ 84]	(Re)Connects the IConnection [▶ 79] when disconnected. (Inherited from IConnection [▶ 79].)
	Connect(AmsAddress) [▶ 873]	Connects the target ADS Device.
	Connect(Int32) [▶ 873]	Connects to the local target ADS Device.
	Connect(AmsNetId, Int32) [▶ 874]	Connects to the target ADS Device.
	Connect(String, Int32) [▶ 874]	Connects to the target ADS Device.

Reference

[IAdsConnectAddress Interface](#) [[▶ 835](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.31.2.1.1 IAdsConnectAddress.Connect Method (Int32)

Connects to the local target ADS Device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void Connect (
    int port
)
```

Parameters

port	Type: System.Int32 The port number of the local ADS target device to connect to.
------	---

Reference

[IAdsConnectAddress Interface](#) [[▶ 835](#)]

[Connect Overload](#) [[▶ 873](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.31.2.1.2 IAdsConnectAddress.Connect Method (AmsAddress)

Connects the target ADS Device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void Connect (
    AmsAddress address
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [► 752] The address of the target device.
---------	---

Reference

[IAdsConnectAddress Interface](#) [► 835]

[Connect Overload](#) [► 873]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.31.2.1.3 IAdsConnectAddress.Connect Method (String, Int32)

Connects to the target ADS Device.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void Connect (
    string netId,
    int port
)
```

Parameters

netId	Type: System.String The AmsNetId [► 767] of the ADS target device specified as string.
port	Type: System.Int32 The port number of the ADS target device.

Reference

[IAdsConnectAddress Interface](#) [► 835]

[Connect Overload](#) [► 873]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.31.2.1.4 IAdsConnectAddress.Connect Method (AmsNetId, Int32)

Connects to the target ADS Device.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void Connect(
    AmsNetId netId,
    int port
)
```

Parameters

netId	Type: TwinCAT.Ads.AmsNetId [▶ 767] The AmsNetId of the target device.
port	Type: System.Int32 The Ams Port number on the target device to connect to.

Reference

[IAdsConnectAddress Interface](#) [▶ 835]












[Connect Overload](#) [▶ 873]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.31.3 IAdsConnectAddress Events

The [IAdsConnectAddress](#) [▶ 835] type exposes the following members.

Events

	Name	Description
 	AdsNotification [▶ 991]	Occurs when the ADS device sends a notification to the client. (Inherited from IAdsNotifications [▶ 966].)
	AdsNotificationError [▶ 993]	Occurs when a exception has occurred during notification management. (Inherited from IAdsNotifications [▶ 966].)
 	AdsNotificationEx [▶ 993]	Occurs when the ADS devices sends a notification to the client. (Inherited from IAdsNotifications [▶ 966].)
	AdsStateChanged [▶ 1061]	Occurs when the AdsState of the target system has been changed. (Inherited from IAdsStateProvider [▶ 1056].)
 	AdsSumNotification [▶ 994]	Occurs when Notifications are send (bundled notifications) (Inherited from IAdsNotifications [▶ 966].)
	AdsSymbolVersionC hanged [▶ 1065]	Occurs when the ADS Symbol Version changed. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
 	ConnectionStateCha nged [▶ 91]	Occurs when connection status of the IConnectionStateProvider [▶ 89] has been changed. (Inherited from IConnectionStateProvider [▶ 89].)

Reference

[IAdsConnectAddress Interface](#) [▶ 835]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.32 IAdsConnection Interface

ADS Connection interface

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229











Syntax

C#



```
public interface IAdsConnection : IConnection,
    IConnectionStateProvider, IAdsNotifications, IAdsSymbolicAccess, IAdsAnyAccess, IAdsHandle,
    IAdsReadWrite2, IAdsReadWrite, IAdsStateProvider, IAdsStateControl, IAdsSymbolChangedProvider,
    IAdsRpcInvoke
```








The IAdsConnection type exposes the following members.

Properties

	Name	Description
	Address [► 895]	Gets the AmsAddress [► 752] of the ADS server.
	ClientAddress [► 895]	Get the AmsAddress [► 752] of the ADS client.
 	ConnectionState [► 90]	Gets the current Connection state of the IConnectionStateProvider [► 89] (Inherited from IConnectionStateProvider [► 89].)
	DefaultValueEncoding [► 81]	Gets the default value encoding. (Inherited from IConnection [► 79].)
	Id [► 81]	Gets the Connection Identifier . (Inherited from IConnection [► 79].)
	IsConnected [► 82]	Gets a value indicating whether the local ADS port was opened successfully. It does not indicate if the target port is available. Use the method ReadState to determine if the target port is available. (Inherited from IConnection [► 79].)
	IsLocal [► 896]	Gets a value indicating whether the ADS client is connected to a ADS Server on the local computer.
	Session [► 82]	Gets the session that initiated this IConnection [► 79] (Inherited from IConnection [► 79].)
	Timeout [► 83]	Gets the timeout (in milliseconds) (Inherited from IConnection [► 79].)












Methods

















	Name	Description
	AddDeviceNotification(String, Int32, NotificationSettings, Object) [► 971]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [► 991] event. (Inherited from IAdsNotifications [► 966].)
	AddDeviceNotification(UInt32, UInt32,	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. (Inherited from IAdsNotifications [► 966].)










	Name	Description
	Int32, NotificationSettings, Object) [▶ 972]	
	AddDeviceNotificationAsync(String, Int32, NotificationSettings, Object, CancellationTok en) [▶ 974]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationAsync(UInt32, UInt32, Int32, NotificationSettings, Object, CancellationTok en) [▶ 975]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type) [▶ 977]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32.) [▶ 977]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type) [▶ 978]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32.) [▶ 980]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive ('AnyType') types are allowed for the parameter type. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationExAsync(String, NotificationSettings, Object, Type, .Int32., CancellationTok en) [▶ 981]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
















	Name	Description
	AddDeviceNotificationExAsync(UInt32, UInt32, NotificationSettings, Object, Type, .Int32, CancellationToken) [▶ 982]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	CleanupSymbolTable [▶ 1070]	Clears the internal symbol / DataTypes cache. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	Close [▶ 83]	Closes this IConnection [▶ 79] (Inherited from IConnection [▶ 79].)
	Connect [▶ 84]	(Re)Connects the IConnection [▶ 79] when disconnected. (Inherited from IConnection [▶ 79].)
	CreateVariableHandle [▶ 956]	Determines the Symbol handle by its instance path synchronously. (Inherited from IAdsHandle [▶ 954].)
	CreateVariableHandleAsync [▶ 956]	Determines the Symbol handle by its instance path asynchronously. (Inherited from IAdsHandle [▶ 954].)
	DeleteDeviceNotification [▶ 983]	Deletes a registered notification. (Inherited from IAdsNotifications [▶ 966].)
	DeleteDeviceNotificationAsync [▶ 984]	Deletes a registered notification asynchronously. (Inherited from IAdsNotifications [▶ 966].)
	DeleteVariableHandle [▶ 957]	Releases the specified symbol/variable handle synchronously. (Inherited from IAdsHandle [▶ 954].)
	DeleteVariableHandleAsync [▶ 958]	Releases the specified symbol/variable handle asynchronously. (Inherited from IAdsHandle [▶ 954].)
	Disconnect [▶ 84]	Disconnects this IConnection [▶ 79]. (Inherited from IConnection [▶ 79].)
 	InvokeRpcMethodAsync(String, String, .Object, CancellationToken) [▶ 1021]	Invokes the specified RPC Method asynchronously (Inherited from IAdsRpcInvoke [▶ 1012].)
 	InvokeRpcMethodAsync(IRpcCallableInstance, IRpcMethod, .Object, .AnyTypeSpecifier, .AnyTypeSpecifier, CancellationToken) [▶ 1024]	Invokes the specified RPC Method asynchronously (Inherited from IAdsRpcInvoke [▶ 1012].)
	InvokeRpcMethodAsync(String,	Invokes the specified RPC Method asynchronously (Inherited from IAdsRpcInvoke [▶ 1012].)















	Name	Description
	String, .Object, .Any TypeSpecifier, AnyTypeSpecifier, CancellationToken) [▶ 1022]	
	Read(UInt32, Memory.Byte.) [▶ 959]	Reads data synchronously from an ADS device and writes to the specified readBuffer. (Inherited from IAdsHandle [▶ 954].)
	Read(UInt32, UInt32, Memory.Byte.) [▶ 1002]	Reads data synchronously from an ADS device and writes it to the given readBuffer (Inherited from IAdsReadWrite2 [▶ 1000].)
	ReadAny(UInt32, Type) [▶ 812]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny(UInt32, Type, .Int32.) [▶ 813]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny(UInt32, UInt32, Type) [▶ 815]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny(UInt32, UInt32, Type, .Int32.) [▶ 815]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32) [▶ 810]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T. (UInt32, .Int32.) [▶ 811]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32, UInt32) [▶ 812]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32, UInt32, .Int32.) [▶ 814]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt 32, Type, CancellationToken) [▶ 819]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt 32, Type, .Int32., CancellationToken) [▶ 821]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt 32, UInt32, Type, CancellationToken) [▶ 823]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt 32, UInt32,	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)














	Name	Description
	Type, .Int32., CancellationToken) [▶ 823]	
	ReadAnyAsync.T. (UInt32, CancellationToken) [▶ 817]	Reads data asynchronously from an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T. (UInt32, .Int32., CancellationToken) [▶ 818]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T. (UInt32, UInt32, CancellationToken) [▶ 820]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T. (UInt32, UInt32, .Int32., CancellationToken) [▶ 822]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyString(UInt 32, Int32, Encoding) [▶ 825]	Reads a string from the specified symbol/variable. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyString(UInt 32, UInt32, Int32, Encoding) [▶ 825]	Reads as string from a specified address. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyStringAsyn c(UInt32, Int32, Encoding, CancellationToken) [▶ 827]	Reads a string asynchronously from the specified symbol/variable (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyStringAsyn c(UInt32, UInt32, Int32, Encoding, CancellationToken) [▶ 827]	Reads a string from a specified address asynchronously. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAsync(UInt32, Memory.Byte., CancellationToken) [▶ 959]	Reads the value data of the symbol asynchronously into the readBuffer. (Inherited from IAdsHandle [▶ 954].)
	ReadAsync(UInt32, UInt32, Memory.Byte., CancellationToken) [▶ 996]	Reads the data asynchronously from specified IndexGroup/IndexOffset (Inherited from IAdsReadWrite [▶ 995].)
	ReadDataType [▶ 1070]	Call this method to obtain information about the specified data type. (Inherited from IAdsSymbolicAccess [▶ 1065].)












	Name	Description
	ReadDataTypeAsync [▶ 1070]	Call this method to obtain information about the specified data type. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadDeviceInfo [▶ 914]	Reads the identification and version number of an ADS server.
	ReadDeviceInfoAsync [▶ 914]	Reads the identification and version number of an ADS server.
	ReadState [▶ 1058]	Reads the ADS status and the device status from an ADS server. (Inherited from IAdsStateProvider [▶ 1056].)
	ReadStateAsync [▶ 1058]	Reads the ADS status and the device status from an ADS server. (Inherited from IAdsStateProvider [▶ 1056].)
	ReadSymbol [▶ 1071]	Call this method to obtain information about the individual symbols (variables) in ADS devices. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadSymbolAsync [▶ 1072]	Call this method to obtain information about the individual symbols (variables) in ADS devices. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValue(ISymbol) [▶ 1073]	Reads the value of a symbol and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValue(String, Type) [▶ 1075]	Reads the value of a symbol and returns the value as object. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValue.T.(String) [▶ 1073]	Reads the value of a symbol and returns the value. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValueAsync(ISymbol, CancellationToken) [▶ 1077]	Reads the value of a symbol asynchronously and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValueAsync(String, Type, CancellationToken) [▶ 1079]	Reads the value of a symbol asynchronously. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValueAsync.T.(String, CancellationToken) [▶ 1076]	Reads the value of a symbol asynchronously. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadWrite(UInt32, Memory.Byte, ReadOnlyMemory.Byte) [▶ 960]	Writes data synchronously to an ADS device and then Reads data from that target. (Inherited from IAdsHandle [▶ 954].)
	ReadWrite(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte) [▶ 1002]	Writes data synchronously to an ADS device and then Reads data from this device into the readBuffer (Inherited from IAdsReadWrite2 [▶ 1000].)
	ReadWriteAsync(UInt32, Memory.Byte, ReadOnlyMemory.Byte)	ReadWrites value data asynchronously to/from the symbol represented by the variableHandle. (Inherited from IAdsHandle [▶ 954].)




	Name	Description
	yte, CancellationToken) [► 961]	
	ReadWriteAsync(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte, CancellationToken) [► 997]	Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer (Inherited from IAdsReadWrite [► 995].)
	RegisterAdsStateChangedAsync [► 1059]	Registers for AdsStateChanged [► 1061] events as an asynchronous operation. (Inherited from IAdsStateProvider [► 1056].)
	RegisterSymbolVersionChanged [► 1064]	Registers the symbol version changed. (Inherited from IAdsSymbolChangedProvider [► 1061].)
	RegisterSymbolVersionChangedAsync [► 1062]	Registers the symbol version changed asynchronously. (Inherited from IAdsSymbolChangedProvider [► 1061].)
	TryAddDeviceNotification(String, Int32, NotificationSettings, Object, UInt32.) [► 985]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [► 991] event. (Inherited from IAdsNotifications [► 966].)
	TryAddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.) [► 986]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [► 991] event. (Inherited from IAdsNotifications [► 966].)
	TryAddDeviceNotificationEx(String, NotificationSettings, Object, Type, Int32, UInt32.) [► 988]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [► 993] event. (Inherited from IAdsNotifications [► 966].)
	TryAddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, Int32, UInt32.) [► 989]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [► 993] event. (Inherited from IAdsNotifications [► 966].)
	TryCreateVariableHandle [► 961]	Determines the Symbol handle by its instance path synchronously. (Inherited from IAdsHandle [► 954].)

	Name	Description
	TryDeleteDeviceNotification [▶ 990]	Deletes a registered notification. (Inherited from IAdsNotifications [▶ 966].)
	TryDeleteVariableHandle [▶ 962]	Releases the specified symbol/variable handle synchronously. (Inherited from IAdsHandle [▶ 954].)
 	TryInvokeRpcMethod(String, String, Object, Object.) [▶ 1026]	Invokes the specified RPC Method (Inherited from IAdsRpcInvoke [▶ 1012].)
 	TryInvokeRpcMethod(String, String, Object, Object.) [▶ 1028]	Invokes the specified RPC Method (Inherited from IAdsRpcInvoke [▶ 1012].)
 	TryInvokeRpcMethod(IRpcCallableInstance, IRpcMethod, Object, AnyTypeSpecifier, AnyTypeSpecifier, Object, Object.) [▶ 1031]	Invokes the rpc method. (Inherited from IAdsRpcInvoke [▶ 1012].)
 	TryInvokeRpcMethod(String, String, Object, AnyTypeSpecifier, AnyTypeSpecifier, Object, Object.) [▶ 1030]	Invokes the rpc method. (Inherited from IAdsRpcInvoke [▶ 1012].)
	TryRead(UInt32, Memory.Byte, Int32.) [▶ 963]	Reads the value synchronously data of the symbol, that is represented by the variable handle into the readBuffer. (Inherited from IAdsHandle [▶ 954].)
	TryRead(UInt32, UInt32, Memory.Byte, Int32.) [▶ 998]	Reads data synchronously from an ADS device and writes it to the given stream. (Inherited from IAdsReadWrite [▶ 995].)
	TryReadDataType [▶ 1079]	Call this method to obtain information about the specified data type. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadState [▶ 1059]	Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful. (Inherited from IAdsStateProvider [▶ 1056].)
	TryReadSymbol [▶ 1080]	Call this method to obtain information about the individual symbols (variables) in ADS devices. (Inherited from IAdsSymbolicAccess [▶ 1065].)












	Name	Description
	TryReadValue(ISymbol, Object.) [▶ 1082]	Reads the value of a symbol and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadValue(String, Type, Object.) [▶ 1083]	Reads the value of a symbol and returns the value as object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadValue.T.(ISymbol, T.) [▶ 1083]	Reads the value of a symbol and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadValue.T.(String, T.) [▶ 1081]	Reads the value of a symbol and returns the value as object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadWrite(UInt32, Memory.Byte, ReadOnlyMemory.Byte, Int32.) [▶ 963]	ReadWrites value data synchronously to/from the symbol represented by the variableHandle. (Inherited from IAdsHandle [▶ 954].)
	TryReadWrite(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte, Int32.) [▶ 998]	Writes data synchronously to an ADS device and reads data from that device. (Inherited from IAdsReadWrite [▶ 995].)
	TryWrite(UInt32, ReadOnlyMemory.Byte.) [▶ 964]	Writes the value data synchronously that is represented in the writeBuffer to the symbol with the specified variableHandle. (Inherited from IAdsHandle [▶ 954].)
	TryWrite(UInt32, UInt32, ReadOnlyMemory.Byte.) [▶ 999]	Writes data synchronously to an ADS device. (Inherited from IAdsReadWrite [▶ 995].)
	TryWriteControl(StateInfo) [▶ 1042]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	TryWriteControl(StateInfo, ReadOnlyMemory.Byte.) [▶ 1043]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	TryWriteValue(ISymbol, Object) [▶ 1086]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryWriteValue(String, Object) [▶ 1085]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryWriteValue.T.(ISymbol, T) [▶ 1087]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryWriteValue.T.(String, T) [▶ 1085]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)

	Name	Description
	UnregisterAdsState ChangedAsync [▶ 1060]	Registers for AdsStateChanged [▶ 1061] events as an asynchronous operation. (Inherited from IAdsStateProvider [▶ 1056].)
	UnregisterSymbolV ersionChanged [▶ 1064]	Unregisters the symbol version changed. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	UnregisterSymbolV ersionChangedAsyn c [▶ 1063]	Unregisters the symbol version changed asynchronously. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	Write(UInt32, ReadOnlyMemory.B yte.) [▶ 964]	Writes data synchronously to an ADS device. (Inherited from IAdsHandle [▶ 954].)
	Write(UInt32, UInt32) [▶ 1003]	Triggers a 'Write' call to the ADS device at the specified address. (Inherited from IAdsReadWrite2 [▶ 1000].)
	Write(UInt32, UInt32, ReadOnlyMemory.B yte.) [▶ 1004]	Writes data synchronously to an ADS device. (Inherited from IAdsReadWrite2 [▶ 1000].)
	WriteAny(UInt32, Object) [▶ 829]	Writes an object synchronously to an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAny(UInt32, Object, .Int32.) [▶ 829]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAny(UInt32, UInt32, Object) [▶ 830]	Writes an object synchronously to an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAny(UInt32, UInt32, Object, .Int32.) [▶ 830]	Writes an object synchronously to an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt 32, Object, CancellationTok en) [▶ 832]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt 32, Object, .Int32, CancellationTok en) [▶ 833]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt 32, UInt32, Object, CancellationTok en) [▶ 834]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)







	Name	Description
	WriteAnyAsync(UInt32, UInt32, Object, Int32, CancellationToken) [▶ 834]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAsync(UInt32, ReadOnlyMemory.Byte, CancellationToken) [▶ 965]	Writes the value data asynchronously that is represented by the writeBuffer to the symbol specified by the variableHandle. (Inherited from IAdsHandle [▶ 954].)
	WriteAsync(UInt32, UInt32, ReadOnlyMemory.Byte, CancellationToken) [▶ 1000]	Writes the data / Value asynchronously into the specified writeBuffer. (Inherited from IAdsReadWrite [▶ 995].)
	WriteControl(StateInfo) [▶ 1044]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	WriteControl(StateInfo, ReadOnlyMemory.Byte) [▶ 1045]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	WriteControlAsync(AdsState, UInt16, CancellationToken) [▶ 1046]	Changes the ADS status and device status of the ADS server asynchronously. (Inherited from IAdsStateControl [▶ 1039].)
	WriteControlAsync(AdsState, UInt16, ReadOnlyMemory.Byte, CancellationToken) [▶ 1047]	Changes the ADS status and device status of the ADS server asynchronously. (Inherited from IAdsStateControl [▶ 1039].)
	WriteValue(ISymbol, Object) [▶ 1089]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValue(String, Object) [▶ 1088]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValue.T.(ISymbol, T) [▶ 1090]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValue.T.(String, T) [▶ 1089]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)

	Name	Description
	WriteValueAsync(ISymbol, Object, CancellationToken) [▶ 1092]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValueAsync.T.(ISymbol, T, CancellationToken) [▶ 1093]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValueAsync.T.(String, T, CancellationToken) [▶ 1091]	Writes the passed object value to the specified ADS symbol.The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)

Events

	Name	Description
 	AdsNotification [▶ 991]	Occurs when the ADS device sends a notification to the client. (Inherited from IAdsNotifications [▶ 966].)
	AdsNotificationError [▶ 993]	Occurs when a exception has occurred during notification management. (Inherited from IAdsNotifications [▶ 966].)
 	AdsNotificationEx [▶ 993]	Occurs when the ADS devices sends a notification to the client. (Inherited from IAdsNotifications [▶ 966].)
	AdsStateChanged [▶ 1061]	Occurs when the AdsState of the target system has been changed. (Inherited from IAdsStateProvider [▶ 1056].)
 	AdsSumNotification [▶ 994]	Occurs when Notifications are send (bundled notifications) (Inherited from IAdsNotifications [▶ 966].)
	AdsSymbolVersionChanged [▶ 1065]	Occurs when the ADS Symbol Version changed. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
 	ConnectionStateChanged [▶ 91]	Occurs when connection status of the IConnectionStateProvider [▶ 89] has been changed. (Inherited from IConnectionStateProvider [▶ 89].)


Extension Methods

	Name	Description
 	PollAdsState(IObservable.Unit.) [▶ 3294]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
 	PollAdsState(IObservable.Unit.) [▶ 1256]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsState(TimeSpan) [▶ 3295]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)

	Name	Description
 	PollAdsState(TimeSpan) [▶ 1257]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState2(IObservable.Unit.) [▶ 1271]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsState2(TimeSpan) [▶ 1272]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState2Async(IObservable.Unit., Cancellation.Token) [▶ 1274]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState2Async(TimeSpan, Cancellation.Token) [▶ 1275]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsStateAsync(IObservable.Unit., Cancellation.Token) [▶ 3296]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
 	PollAdsStateAsync(IObservable.Unit., Cancellation.Token) [▶ 1259]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsStateAsync(TimeSpan, Cancellation.Token) [▶ 3298]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
 	PollAdsStateAsync(TimeSpan, Cancellation.Token) [▶ 1260]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollDeviceState(IObservable.Unit.) [▶ 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollDeviceState(TimeSpan) [▶ 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)









	Name	Description
	PollDeviceStateAsync(IObservable.Unit., Cancellation.Token) [▶ 1278]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157] s via Polling. (Defined by AdsClientExtensions [▶ 1249] .)
	PollDeviceStateAsync(TimeSpan, Cancellation.Token) [▶ 1279]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157] s via Polling. (Defined by AdsClientExtensions [▶ 1249] .)
	PollSystemServiceState [▶ 3282]	Polls the state of the system service. (Defined by SystemServiceExtension [▶ 3281] .)
	PollSystemServiceStateAsync [▶ 3283]	Polls the system service state asynchronously (Defined by SystemServiceExtension [▶ 3281] .)
	PollValues(ISymbol, Type, IObservable.Unit.) [▶ 1310]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(ISymbol, Type, TimeSpan) [▶ 1311]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(String, Type, IObservable.Unit.) [▶ 1300]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(String, Type, TimeSpan) [▶ 1301]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(ISymbol, Type, .Int32., TimeSpan) [▶ 1316]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(ISymbol, Type, IObservable.Unit., Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(ISymbol, Type, TimeSpan, Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(String, Type, .Int32., TimeSpan) [▶ 1304]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(String, Type, IObservable.Unit., Func.Exception, Object.) [▶ 1305]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281] .)

	Name	Description
	PollValues(String, Type, TimeSpan, Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32., IObservable.Unit, Func.Exception, Object.) [▶ 1321]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32., TimeSpan, Func.Exception, Object.) [▶ 1322]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32., IObservable.Unit, Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32., TimeSpan, Func.Exception, Object.) [▶ 1307]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol, IObservable.Unit.) [▶ 1308]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol, TimeSpan) [▶ 1309]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, IObservable.Unit.) [▶ 1294]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, TimeSpan) [▶ 1295]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol, IObservable.Unit., Func.Exception, T.) [▶ 1314]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol, TimeSpan, Func.Exception, T.) [▶ 1315]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol, .Int32., IObservable.Unit.) [▶ 1312]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	PollValues.T. (ISymbol , .Int32 , TimeSpan) [▶ 1313]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, IObservable.Unit, Func.Exception, T.) [▶ 1298]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, TimeSpan, Func.Exception, T.) [▶ 1299]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32 , IObservable.Unit .) [▶ 1296]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32 , TimeSpan) [▶ 1297]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol , .Int32 , IObservable.Unit , Func.Exception, T.) [▶ 1318]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol , .Int32 , TimeSpan , Func.Exception, T.) [▶ 1319]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32 , IObservable.Unit , Func.Exception, T.) [▶ 1302]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32 , TimeSpan , Func.Exception, T.) [▶ 1303]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues2.T. (ISymbol , .Int32 , IObservable.Unit .) [▶ 1330]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues2.T. (String , .Int32 , IObservable.Unit .) [▶ 1329]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	ReadSysServState [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (Defined by SystemServiceExtension [▶ 3281].)

	Name	Description
	ReadSysServStateAsync [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (asynchronous) (Defined by SystemServiceExtension [▶ 3281].)
	ReadWithFallback(UInt32, UInt32, Memory.Byte., UInt32, Boolean.) [▶ 1219]	Overloaded. Ads Read with fallback. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallback(UInt32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., Boolean.) [▶ 1220]	Overloaded. Ads Read with Fallback. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallbackAsync(UInt32, UInt32, UInt32, Memory.Byte., CancellationToken) [▶ 1221]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallbackAsync(UInt32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., CancellationToken) [▶ 1222]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedRead(UInt32, UInt32, Memory.Byte., Int32, TimeSpan) [▶ 1224]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedRead(UInt32, UInt32, Memory.Byte., Int32, TimeSpan, Func.ResultRead, Boolean.) [▶ 1225]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedReadAsync(UInt32, UInt32, Memory.Byte., Int32, TimeSpan, CancellationToken) [▶ 1226]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)

	Name	Description
	RepeatedReadAsync (UInt32, UInt32, Memory.Byte., Int32, TimeSpan, Func.ResultRead, Boolean., CancellationToken) [1227]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [1215].)
	RestartTwinCATAsync [3285]	Sends a SystemService Restart to the target system and waits until the system has finished the state changes (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [3281].)
	SetAdsState [3299]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [3292].)
	SetAdsStateAsync [3300]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [3292].)
	WaitUntilRestarted [3286]	Waits until the Restart is detected on the client (SystemService, Port 10000) (Defined by SystemServiceExtension [3281].)
	WaitUntilRestartedAsync [3287]	Waits until the SystemService Restart is detected on the client (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [3281].)
 	WhenAdsStateChanges [1261]	Gets an observable sequence of AdsState [729]s. (Defined by AdsClientExtensions [1249].)
	WhenNotification(ISymbol) [1263]	Overloaded. Gets an observable sequence of Notification [1100]s. (Defined by AdsClientExtensions [1249].)
 	WhenNotification(ISymbolCollection) [1264]	Overloaded. Gets an observable sequence of Notification [1100] objects. (Defined by AdsClientExtensions [1249].)
 	WhenNotification(IList.ISymbol, NotificationSettings) [1268]	Overloaded. Gets an observable sequence of Notification [1100] objects. (Defined by AdsClientExtensions [1249].)
	WhenNotification(ISymbol, NotificationSettings) [1265]	Overloaded. Gets an observable sequence of SymbolValueNotification [1331]s. (Defined by AdsClientExtensions [1249].)
 	WhenNotification(STRING, Type, NotificationSettings) [1324]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [1281].)
 	WhenNotification.T.(String, NotificationSettings) [1323]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [1281].)

	Name	Description
	WhenSymbolVersionChanges [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
	WhenSymbolVersionChanges(IScheduler) [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenValueChanged [▶ 1345]	Observable sequence of Values driven by ADS Notifications on the specified symbol. (Defined by ValueSymbolExtensions [▶ 1333].)
 	WriteValues.T.(String, IObservable.T.) [▶ 1326]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)
 	WriteValues.T.(String, IObservable.T., Action.Exception.) [▶ 1327]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)

Remarks

The ADS connection interface represents the the logical point-to-point exception between Client and Server within an ADS Session.

Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[TwinCAT.IConnection](#) [[▶ 79](#)]

[TwinCAT.Ads.IAdsNotifications](#) [[▶ 966](#)]


[TwinCAT.Ads.IAdsAnyAccess](#) [[▶ 804](#)]

[TwinCAT.Ads.IAdsHandle](#) [[▶ 954](#)]

[TwinCAT.Ads.IAdsReadWrite2](#) [[▶ 1000](#)]

[TwinCAT.Ads.IAdsStateControl](#) [[▶ 1039](#)]


Also see about this










-  [AdsClientExtensions.WhenNotification Method \(IAdsConnection, ISymbolCollection, NotificationSettings\)](#) [[▶ 1266](#)]

6.2.32.1 IAdsConnection Properties

The [IAdsConnection](#) [[▶ 876](#)] type exposes the following members.

Properties

	Name	Description
	Address [▶ 895]	Gets the AmsAddress [▶ 752] of the ADS server.

	Name	Description
	ClientAddress [▶ 895]	Get the AmsAddress [▶ 752] of the ADS client.
 	ConnectionState [▶ 90]	Gets the current Connection state of the IConnectionStateProvider [▶ 89] (Inherited from IConnectionStateProvider [▶ 89] .)
	DefaultValueEncoding [▶ 81]	Gets the default value encoding. (Inherited from IConnection [▶ 79] .)
	Id [▶ 81]	Gets the Connection Identifier . (Inherited from IConnection [▶ 79] .)
	IsConnected [▶ 82]	Gets a value indicating whether the local ADS port was opened successfully. It does not indicate if the target port is available. Use the method ReadState to determine if the target port is available. (Inherited from IConnection [▶ 79] .)
	IsLocal [▶ 896]	Gets a value indicating whether the ADS client is connected to a ADS Server on the local computer.
	Session [▶ 82]	Gets the session that initiated this IConnection [▶ 79] (Inherited from IConnection [▶ 79] .)
	Timeout [▶ 83]	Gets the timeout (in milliseconds) (Inherited from IConnection [▶ 79] .)

Reference

[IAdsConnection Interface](#) [\[▶ 876\]](#)

[TwinCAT.Ads Namespace](#) [\[▶ 179\]](#)

6.2.32.1.1 IAdsConnection.Address Property

Gets the [AmsAddress](#) [\[▶ 752\]](#) of the ADS server.

Namespace: [TwinCAT.Ads](#) [\[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AmsAddress Address { get; }
```

Property Value

Type: [AmsAddress](#) [\[▶ 752\]](#)

Reference

[IAdsConnection Interface](#) [\[▶ 876\]](#)

[TwinCAT.Ads Namespace](#) [\[▶ 179\]](#)

6.2.32.1.2 IAdsConnection.ClientAddress Property

Get the [AmsAddress](#) [\[▶ 752\]](#) of the ADS client.

Namespace: [TwinCAT.Ads](#) [\[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AmsAddress ClientAddress { get; }
```

Property Value

Type: [AmsAddress](#) [► 752]

Reference

[IAdsConnection Interface](#) [► 876]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.32.1.3 IAdsConnection.IsLocal Property

Gets a value indicating whether the ADS client is connected to a ADS Server on the local computer.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsLocal { get; }
```

Property Value

Type: [Boolean](#)

Reference




[IAdsConnection Interface](#) [► 876]








[TwinCAT.Ads Namespace](#) [► 179]

6.2.32.2 IAdsConnection Methods











The [IAdsConnection](#) [► 876] type exposes the following members.










Methods















	Name	Description
	AddDeviceNotification(String, Int32, NotificationSettings, Object) [► 971]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [► 991] event. (Inherited from IAdsNotifications [► 966].)
	AddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object) [► 972]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. (Inherited from IAdsNotifications [► 966].)
	AddDeviceNotificationAsync(String,	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [► 991] event. (Inherited from IAdsNotifications [► 966].)











	Name	Description
	Int32 , NotificationSettings , Object , CancellationToken) [▶ 974]	
	AddDeviceNotificationAsync (UInt32 , UInt32 , Int32 , NotificationSettings , Object , CancellationToken) [▶ 975]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationEx (String , NotificationSettings , Object , Type) [▶ 977]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationEx (String , NotificationSettings , Object , Type , .Int32 .) [▶ 977]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationEx (UInt32 , UInt32 , NotificationSettings , Object , Type) [▶ 978]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationEx (UInt32 , UInt32 , NotificationSettings , Object , Type , .Int32 .) [▶ 980]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive ('AnyType') types are allowed for the parameter type. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationExAsync (String , NotificationSettings , Object , Type , .Int32 , CancellationToken) [▶ 981]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationExAsync (UInt32 , UInt32 , NotificationSettings , Object , Type , .Int32 , CancellationToken) [▶ 982]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
















	Name	Description
	CleanupSymbolTable [▶ 1070]	Clears the internal symbol / DataTypes cache. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	Close [▶ 83]	Closes this IConnection [▶ 79] (Inherited from IConnection [▶ 79].)
	Connect [▶ 84]	(Re)Connects the IConnection [▶ 79] when disconnected. (Inherited from IConnection [▶ 79].)
	CreateVariableHandle [▶ 956]	Determines the Symbol handle by its instance path synchronously. (Inherited from IAdsHandle [▶ 954].)
	CreateVariableHandleAsync [▶ 956]	Determines the Symbol handle by its instance path asynchronously. (Inherited from IAdsHandle [▶ 954].)
	DeleteDeviceNotification [▶ 983]	Deletes a registered notification. (Inherited from IAdsNotifications [▶ 966].)
	DeleteDeviceNotificationAsync [▶ 984]	Deletes a registered notification asynchronously. (Inherited from IAdsNotifications [▶ 966].)
	DeleteVariableHandle [▶ 957]	Releases the specified symbol/variable handle synchronously. (Inherited from IAdsHandle [▶ 954].)
	DeleteVariableHandleAsync [▶ 958]	Releases the specified symbol/variable handle asynchronously. (Inherited from IAdsHandle [▶ 954].)
	Disconnect [▶ 84]	Disconnects this IConnection [▶ 79]. (Inherited from IConnection [▶ 79].)
 	InvokeRpcMethodAsync(String, String, Object, CancellationToken) [▶ 1021]	Invokes the specified RPC Method asynchronously (Inherited from IAdsRpcInvoke [▶ 1012].)
 	InvokeRpcMethodAsync(IRpcCallableObject, IRpcMethod, Object, AnyTypeSpecifier, AnyTypeSpecifier, CancellationToken) [▶ 1024]	Invokes the specified RPC Method asynchronously (Inherited from IAdsRpcInvoke [▶ 1012].)
 	InvokeRpcMethodAsync(String, String, Object, AnyTypeSpecifier, AnyTypeSpecifier, CancellationToken) [▶ 1022]	Invokes the specified RPC Method asynchronously (Inherited from IAdsRpcInvoke [▶ 1012].)
	Read(UInt32, Memory.Byte) [▶ 959]	Reads data synchronously from an ADS device and writes to the specified readBuffer. (Inherited from IAdsHandle [▶ 954].)















	Name	Description
	Read(UInt32, UInt32, Memory.Byte.) [▶ 1002]	Reads data synchronously from an ADS device and writes it to the given readBuffer (Inherited from IAdsReadWrite2 [▶ 1000].)
	ReadAny(UInt32, Type) [▶ 812]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny(UInt32, Type, .Int32.) [▶ 813]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny(UInt32, UInt32, Type) [▶ 815]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny(UInt32, UInt32, Type, .Int32.) [▶ 815]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32) [▶ 810]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32, .Int32.) [▶ 811]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32, UInt32) [▶ 812]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32, UInt32, .Int32.) [▶ 814]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt32, Type, CancellationToken) [▶ 819]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt32, Type, .Int32., CancellationToken) [▶ 821]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt32, UInt32, Type, CancellationToken) [▶ 823]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt32, UInt32, Type, .Int32., CancellationToken) [▶ 823]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T.(UInt32, CancellationToken) [▶ 817]	Reads data asynchronously from an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)













	Name	Description
	ReadAnyAsync.T. (UInt32, Int32, CancellationToken) [▶ 818]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T. (UInt32, UInt32, CancellationToken) [▶ 820]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T. (UInt32, UInt32, Int32, CancellationToken) [▶ 822]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyString (UInt32, Int32, Encoding) [▶ 825]	Reads a string from the specified symbol/variable. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyString (UInt32, UInt32, Int32, Encoding) [▶ 825]	Reads as string from a specified address. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyStringAsync (UInt32, Int32, Encoding, CancellationToken) [▶ 827]	Reads a string asynchronously from the specified symbol/variable (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyStringAsync (UInt32, UInt32, Int32, Encoding, CancellationToken) [▶ 827]	Reads a string from a specified address asynchronously. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAsync (UInt32, Memory.Byte, CancellationToken) [▶ 959]	Reads the value data of the symbol asynchronously into the readBuffer. (Inherited from IAdsHandle [▶ 954].)
	ReadAsync (UInt32, UInt32, Memory.Byte, CancellationToken) [▶ 996]	Reads the data asynchronously from specified IndexGroup/IndexOffset (Inherited from IAdsReadWrite [▶ 995].)
	ReadDataType [▶ 1070]	Call this method to obtain information about the specified data type. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadDataTypeAsync [▶ 1070]	Call this method to obtain information about the specified data type. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadDeviceInfo [▶ 914]	Reads the identification and version number of an ADS server.
	ReadDeviceInfoAsync [▶ 914]	Reads the identification and version number of an ADS server.












	Name	Description
	ReadState [▶ 1058]	Reads the ADS status and the device status from an ADS server. (Inherited from IAdsStateProvider [▶ 1056].)
	ReadStateAsync [▶ 1058]	Reads the ADS status and the device status from an ADS server. (Inherited from IAdsStateProvider [▶ 1056].)
	ReadSymbol [▶ 1071]	Call this method to obtain information about the individual symbols (variables) in ADS devices. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadSymbolAsync [▶ 1072]	Call this method to obtain information about the individual symbols (variables) in ADS devices. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValue(ISymbol) [▶ 1073]	Reads the value of a symbol and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValue(String, Type) [▶ 1075]	Reads the value of a symbol and returns the value as object. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValue.T.(String) [▶ 1073]	Reads the value of a symbol and returns the value. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValueAsync(ISymbol, CancellationToken) [▶ 1077]	Reads the value of a symbol asynchronously and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValueAsync(String, Type, CancellationToken) [▶ 1079]	Reads the value of a symbol asynchronously. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValueAsync.T.(String, CancellationToken) [▶ 1076]	Reads the value of a symbol asynchronously. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadWrite(UInt32, Memory.Byte, ReadOnlyMemory.Byte.) [▶ 960]	Writes data synchronously to an ADS device and then Reads data from that target. (Inherited from IAdsHandle [▶ 954].)
	ReadWrite(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte.) [▶ 1002]	Writes data synchronously to an ADS device and then Reads data from this device into the readBuffer (Inherited from IAdsReadWrite2 [▶ 1000].)
	ReadWriteAsync(UInt32, Memory.Byte, ReadOnlyMemory.Byte, CancellationToken) [▶ 961]	ReadWrites value data asynchronously to/from the symbol represented by the variableHandle. (Inherited from IAdsHandle [▶ 954].)
	ReadWriteAsync(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.B	Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer (Inherited from IAdsReadWrite [▶ 995].)



	Name	Description
	yte, CancellationToken [▶ 997]	
	RegisterAdsStateCh angedAsync [▶ 1059]	Registers for AdsStateChanged [▶ 1061] events as an asynchronous operation. (Inherited from IAdsStateProvider [▶ 1056].)
	RegisterSymbolVers ionChanged [▶ 1064]	Registers the symbol version changed. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	RegisterSymbolVers ionChangedAsync [▶ 1062]	Registers the symbol version changed asynchronously. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	TryAddDeviceNotifi cation(String, Int32, NotificationSettings, Object, UInt32.) [▶ 985]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)
	TryAddDeviceNotifi cation(UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.) [▶ 986]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)
	TryAddDeviceNotifi cationEx(String, NotificationSettings, Object, Type, .Int32., UInt32.) [▶ 988]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	TryAddDeviceNotifi cationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., UInt32.) [▶ 989]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	TryCreateVariableH andle [▶ 961]	Determines the Symbol handle by its instance path synchronously. (Inherited from IAdsHandle [▶ 954].)
	TryDeleteDeviceNot ification [▶ 990]	Deletes a registered notification. (Inherited from IAdsNotifications [▶ 966].)
	TryDeleteVariableHa ndle [▶ 962]	Releases the specified symbol/variable handle synchronously. (Inherited from IAdsHandle [▶ 954].)

	Name	Description
 	TryInvokeRpcMethod(String, String, Object, Object.) [▶ 1026]	Invokes the specified RPC Method (Inherited from IAdsRpcInvoke [▶ 1012].)
 	TryInvokeRpcMethod(String, String, Object, Object, Object, Object.) [▶ 1028]	Invokes the specified RPC Method (Inherited from IAdsRpcInvoke [▶ 1012].)
 	TryInvokeRpcMethod(IRpcCallableInstance, IRpcMethod, Object, AnyTypeSpecifier, AnyTypeSpecifier, Object, Object.) [▶ 1031]	Invokes the rpc method. (Inherited from IAdsRpcInvoke [▶ 1012].)
 	TryInvokeRpcMethod(String, String, Object, AnyTypeSpecifier, AnyTypeSpecifier, Object, Object.) [▶ 1030]	Invokes the rpc method. (Inherited from IAdsRpcInvoke [▶ 1012].)
	TryRead(UInt32, Memory.Byte, Int32.) [▶ 963]	Reads the value synchronously data of the symbol, that is represented by the variable handle into the readBuffer. (Inherited from IAdsHandle [▶ 954].)
	TryRead(UInt32, UInt32, Memory.Byte, Int32.) [▶ 998]	Reads data synchronously from an ADS device and writes it to the given stream. (Inherited from IAdsReadWrite [▶ 995].)
	TryReadDataType [▶ 1079]	Call this method to obtain information about the specified data type. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadState [▶ 1059]	Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful. (Inherited from IAdsStateProvider [▶ 1056].)
	TryReadSymbol [▶ 1080]	Call this method to obtain information about the individual symbols (variables) in ADS devices. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadValue(ISymbol, Object.) [▶ 1082]	Reads the value of a symbol and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadValue(String, Type, Object.) [▶ 1083]	Reads the value of a symbol and returns the value as object. (Inherited from IAdsSymbolicAccess [▶ 1065].)

	Name	Description
	TryReadValue.T. (ISymbol, T.) [▶ 1083]	Reads the value of a symbol and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadValue.T. (String, T.) [▶ 1081]	Reads the value of a symbol and returns the value as object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadWrite(UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.) [▶ 963]	ReadWrites value data synchronously to/from the symbol represented by the variableHandle. (Inherited from IAdsHandle [▶ 954].)
	TryReadWrite(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.) [▶ 998]	Writes data synchronously to an ADS device and reads data from that device. (Inherited from IAdsReadWrite [▶ 995].)
	TryWrite(UInt32, ReadOnlyMemory.Byte.) [▶ 964]	Writes the value data synchronously that is represented in the writeBuffer to the symbol with the specified variableHandle. (Inherited from IAdsHandle [▶ 954].)
	TryWrite(UInt32, UInt32, ReadOnlyMemory.Byte.) [▶ 999]	Writes data synchronously to an ADS device. (Inherited from IAdsReadWrite [▶ 995].)
	TryWriteControl(StateInfo) [▶ 1042]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	TryWriteControl(StateInfo, ReadOnlyMemory.Byte.) [▶ 1043]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	TryWriteValue(ISymbol, Object) [▶ 1086]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryWriteValue(String, Object) [▶ 1085]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryWriteValue.T. (ISymbol, T.) [▶ 1087]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryWriteValue.T. (String, T.) [▶ 1085]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	UnregisterAdsStateChangedAsync [▶ 1060]	Registers for AdsStateChanged [▶ 1061] events as an asynchronous operation. (Inherited from IAdsStateProvider [▶ 1056].)
	UnregisterSymbolVersionChanged [▶ 1064]	Unregisters the symbol version changed. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)

	Name	Description
	UnregisterSymbolVersionChangedAsynchronous [▶ 1063]	Unregisters the symbol version changed asynchronous. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	Write(UInt32, ReadOnlyMemory<Byte>) [▶ 964]	Writes data synchronously to an ADS device. (Inherited from IAdsHandle [▶ 954].)
	Write(UInt32, UInt32) [▶ 1003]	Triggers a 'Write' call to the ADS device at the specified address. (Inherited from IAdsReadWrite2 [▶ 1000].)
	Write(UInt32, UInt32, ReadOnlyMemory<Byte>) [▶ 1004]	Writes data synchronously to an ADS device. (Inherited from IAdsReadWrite2 [▶ 1000].)
	WriteAny(UInt32, Object) [▶ 829]	Writes an object synchronously to an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAny(UInt32, Object, Int32) [▶ 829]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAny(UInt32, UInt32, Object) [▶ 830]	Writes an object synchronously to an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAny(UInt32, UInt32, Object, Int32) [▶ 830]	Writes an object synchronously to an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt32, Object, CancellationToken) [▶ 832]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt32, Object, Int32, CancellationToken) [▶ 833]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt32, UInt32, Object, CancellationToken) [▶ 834]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt32, UInt32, Object, Int32, CancellationToken) [▶ 834]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)

	Name	Description
	WriteAsync(UInt32, ReadOnlyMemory<Byte>, CancellationToken) [▶ 965]	Writes the value data asynchronously that is represented by the writeBuffer to the symbol specified by the variableHandle. (Inherited from IAdsHandle [▶ 954].)
	WriteAsync(UInt32, UInt32, ReadOnlyMemory<Byte>, CancellationToken) [▶ 1000]	Writes the data / Value asynchronously into the specified writeBuffer. (Inherited from IAdsReadWrite [▶ 995].)
	WriteControl(StateInfo) [▶ 1044]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	WriteControl(StateInfo, ReadOnlyMemory<Byte>) [▶ 1045]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	WriteControlAsync(AdsState, UInt16, CancellationToken) [▶ 1046]	Changes the ADS status and device status of the ADS server asynchronously. (Inherited from IAdsStateControl [▶ 1039].)
	WriteControlAsync(AdsState, UInt16, ReadOnlyMemory<Byte>, CancellationToken) [▶ 1047]	Changes the ADS status and device status of the ADS server asynchronously. (Inherited from IAdsStateControl [▶ 1039].)
	WriteValue(ISymbol, Object) [▶ 1089]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValue(String, Object) [▶ 1088]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValue.T.(ISymbol, T) [▶ 1090]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValue.T.(String, T) [▶ 1089]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValueAsync(ISymbol, Object, CancellationToken) [▶ 1092]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)

	Name	Description
	WriteValueAsync.T. (ISymbol, T, CancellationToken) [▶ 1093]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValueAsync.T. (String, T, CancellationToken) [▶ 1091]	Writes the passed object value to the specified ADS symbol.The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)


Extension Methods












	Name	Description
	PollAdsState(IObservable.Unit.) [▶ 3294]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
	PollAdsState(IObservable.Unit.) [▶ 1256]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState(TimeSpan) [▶ 3295]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
	PollAdsState(TimeSpan) [▶ 1257]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState2(IObservable.Unit.) [▶ 1271]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState2(TimeSpan) [▶ 1272]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState2Async(IObservable.Unit., CancellationToken) [▶ 1274]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState2Async(TimeSpan, CancellationToken) [▶ 1275]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsStateAsync(IObservable.Unit., CancellationToken) [▶ 3296]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)

	Name	Description
	PollAdsStateAsync(IObservable.Unit., CancellationToken) [▶ 1259]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsStateAsync(TimeSpan, CancellationToken) [▶ 3298]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
	PollAdsStateAsync(TimeSpan, CancellationToken) [▶ 1260]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollDeviceState(IObservable.Unit.) [▶ 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollDeviceState(TimeSpan) [▶ 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollDeviceStateAsync(IObservable.Unit., CancellationToken) [▶ 1278]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollDeviceStateAsync(TimeSpan, CancellationToken) [▶ 1279]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollSystemServiceState [▶ 3282]	Polls the state of the system service. (Defined by SystemServiceExtension [▶ 3281].)
	PollSystemServiceStateAsync [▶ 3283]	Polls the system service state asynchronously (Defined by SystemServiceExtension [▶ 3281].)
	PollValues(ISymbol, Type, IObservable.Unit.) [▶ 1310]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, TimeSpan) [▶ 1311]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, IObservable.Unit.) [▶ 1300]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	PollValues(String, Type, TimeSpan) [▶ 1301]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32, TimeSpan) [▶ 1316]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, IObservable.Unit, Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, TimeSpan, Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32, TimeSpan) [▶ 1304]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, IObservable.Unit, Func.Exception, Object.) [▶ 1305]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, TimeSpan, Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32, IObservable.Unit, Func.Exception, Object.) [▶ 1321]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32, TimeSpan, Func.Exception, Object.) [▶ 1322]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32, IObservable.Unit, Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32, TimeSpan, Func.Exception, Object.) [▶ 1307]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	PollValues.T. (ISymbol , IObservable.Unit.) [▶ 1308]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol , TimeSpan) [▶ 1309]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String. IObservable.Unit.) [▶ 1294]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String. TimeSpan) [▶ 1295]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol , IObservable.Unit. , Func.Exception , T.) [▶ 1314]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol , TimeSpan , Func.Exception , T.) [▶ 1315]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol , .Int32. , IObservable.Unit.) [▶ 1312]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	PollValues.T. (ISymbol , .Int32. , TimeSpan) [▶ 1313]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String. IObservable.Unit. , Func.Exception , T.) [▶ 1298]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String. TimeSpan , Func.Exception , T.) [▶ 1299]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32. , IObservable.Unit.) [▶ 1296]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	PollValues.T. (String , .Int32. , TimeSpan) [▶ 1297]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol , .Int32. , IObservable.Unit. , Func.Exception , T.) [▶ 1318]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	PollValues.T. (ISymbol , .Int32 , TimeSpan , Func.Exception , T.) [▶ 1319]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32 , IObservable.Unit , Func.Exception , T.) [▶ 1302]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32 , TimeSpan , Func.Exception , T.) [▶ 1303]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues2.T. (ISymbol , .Int32 , IObservable.Unit .) [▶ 1330]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues2.T. (String , .Int32 , IObservable.Unit .) [▶ 1329]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	ReadSysServState [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (Defined by SystemServiceExtension [▶ 3281].)
	ReadSysServStateAs ync [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (asynchronous) (Defined by SystemServiceExtension [▶ 3281].)
	ReadWithFallback(U Int32, UInt32, Memory.Byte., UInt32, Boolean.) [▶ 1219]	Overloaded. Ads Read with fallback. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallback(U Int32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., Boolean.) [▶ 1220]	Overloaded. Ads Read with Fallback. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallbackAs ync(UInt32, UInt32, UInt32, Memory.Byte., CancellationTok en) [▶ 1221]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)

	Name	Description
	ReadWithFallbackAsync(UInt32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., CancellationToken) [▶ 1222]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedRead(UInt32, UInt32, Memory.Byte., Int32, TimeSpan) [▶ 1224]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedRead(UInt32, UInt32, Memory.Byte., Int32, TimeSpan, Func.ResultRead, Boolean.) [▶ 1225]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedReadAsync(UInt32, UInt32, Memory.Byte., Int32, TimeSpan, CancellationToken) [▶ 1226]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedReadAsync(UInt32, UInt32, Memory.Byte., Int32, TimeSpan, Func.ResultRead, Boolean., CancellationToken) [▶ 1227]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	RestartTwinCATAsync [▶ 3285]	Sends a SystemService Restart to the target system and waits until the system has finished the state changes (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [▶ 3281].)
	SetAdsState [▶ 3299]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [▶ 3292].)
	SetAdsStateAsync [▶ 3300]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [▶ 3292].)
	WaitUntilRestarted [▶ 3286]	Waits until the Restart is detected on the client (SystemService, Port 10000) (Defined by SystemServiceExtension [▶ 3281].)
	WaitUntilRestartedAsync [▶ 3287]	Waits until the SystemService Restart is detected on the client (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [▶ 3281].)
	WhenAdsStateChanges [▶ 1261]	Gets an observable sequence of AdsState [▶ 729]s. (Defined by AdsClientExtensions [▶ 1249].)


	Name	Description
	WhenNotification(ISymbol) [▶ 1263]	Overloaded. Gets an observable sequence of Notification [▶ 1100]s. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenNotification(ISymbolCollection) [▶ 1264]	Overloaded. Gets an observable sequence of Notification [▶ 1100] objects. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenNotification(IList.ISymbol, NotificationSettings) [▶ 1268]	Overloaded. Gets an observable sequence of Notification [▶ 1100] objects. (Defined by AdsClientExtensions [▶ 1249].)
	WhenNotification(ISymbol, NotificationSettings) [▶ 1265]	Overloaded. Gets an observable sequence of SymbolValueNotification [▶ 1331]s. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenNotification(String, Type, NotificationSettings) [▶ 1324]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [▶ 1281].)
 	WhenNotification.T.(String, NotificationSettings) [▶ 1323]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [▶ 1281].)
	WhenSymbolVersionChanges . [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
	WhenSymbolVersionChanges(IScheduler) [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenValueChanged [▶ 1345]	Observable sequence of Values driven by ADS Notifications on the specified symbol. (Defined by ValueSymbolExtensions [▶ 1333].)
 	WriteValues.T.(String, IObservable.T.) [▶ 1326]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)
 	WriteValues.T.(String, IObservable.T., Action.Exception.) [▶ 1327]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)

Reference

[IAdsConnection Interface](#) [[▶ 876](#)]

[TwinCAT.Ads Namespace](#) [▶ 179]

Also see about this

-  [AdsClientExtensions.WhenNotification Method \(IAdsConnection, ISymbolCollection, NotificationSettings\)](#) [▶ 1266]

6.2.32.2.1 IAdsConnection.ReadDeviceInfo Method

Reads the identification and version number of an ADS server.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
DeviceInfo ReadDeviceInfo ()
```

Return Value

Type: [DeviceInfo](#) [▶ 801]

DeviceInfo struct containing the name of the device and the version information.

Reference

[IAdsConnection Interface](#) [▶ 876]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.32.2.2 IAdsConnection.ReadDeviceInfoAsync Method

Reads the identification and version number of an ADS server.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultDeviceInfo> ReadDeviceInfoAsync (
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultDeviceInfo](#) [▶ 1132].

A task that represents the asynchronous 'ReadDeviceState' operation. The [ResultDeviceInfo](#) [▶ 1132] parameter contains the value [DeviceInfo](#) [▶ 1134] and the [ErrorCode](#) [▶ 1120] of the ADS communication after execution.

Reference












[IAdsConnection Interface](#) [▶ 876]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.32.3 IAdsConnection Events

The [IAdsConnection](#) [▶ 876] type exposes the following members.

Events

	Name	Description
 	AdsNotification [▶ 991]	Occurs when the ADS device sends a notification to the client. (Inherited from IAdsNotifications [▶ 966].)
	AdsNotificationError [▶ 993]	Occurs when a exception has occurred during notification management. (Inherited from IAdsNotifications [▶ 966].)
 	AdsNotificationEx [▶ 993]	Occurs when the ADS devices sends a notification to the client. (Inherited from IAdsNotifications [▶ 966].)
	AdsStateChanged [▶ 1061]	Occurs when the AdsState of the target system has been changed. (Inherited from IAdsStateProvider [▶ 1056].)
 	AdsSumNotification [▶ 994]	Occurs when Notifications are send (bundled notifications) (Inherited from IAdsNotifications [▶ 966].)
	AdsSymbolVersionC hanged [▶ 1065]	Occurs when the ADS Symbol Version changed. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
 	ConnectionStateCha nged [▶ 91]	Occurs when connection status of the IConnectionStateProvider [▶ 89] has been changed. (Inherited from IConnectionStateProvider [▶ 89].)

Reference

[IAdsConnection Interface](#) [▶ 876]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.33 IAdsDisposableConnection Interface

Interface [IAdsDisposableConnection](#) Implements the [IAdsConnectAddress](#) [▶ 835] Implements the [IRouterNotificationProvider](#) [▶ 1099] Implements the [IAdsSymbolChangedProvider](#) [▶ 1061] Implements the [IDisposable](#)

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229












Syntax

C#





```
public interface IAdsDisposableConnection : IAdsConnectAddress,
    IAdsConnection, IConnection, IConnectionStateProvider, IAdsNotifications, IAdsSymbolicAccess,
    IAdsAnyAccess, IAdsHandle, IAdsReadWrite2, IAdsReadWrite, IAdsStateProvider,
    IAdsStateControl, IAdsSymbolChangedProvider, IAdsRpcInvoke, IRouterNotificationProvider, IDispos
able
```










The [IAdsDisposableConnection](#) type exposes the following members.




Properties















	Name	Description
	Address [▶ 895]	Gets the AmsAddress [▶ 752] of the ADS server. (Inherited from IAdsConnection [▶ 876].)
	ClientAddress [▶ 895]	Get the AmsAddress [▶ 752] of the ADS client. (Inherited from IAdsConnection [▶ 876].)
 	ConnectionState [▶ 90]	Gets the current Connection state of the IConnectionStateProvider [▶ 89] (Inherited from IConnectionStateProvider [▶ 89].)
	DefaultValueEncoding [▶ 81]	Gets the default value encoding. (Inherited from IConnection [▶ 79].)
	Id [▶ 81]	Gets the Connection Identifier . (Inherited from IConnection [▶ 79].)
	IsConnected [▶ 82]	Gets a value indicating whether the local ADS port was opened successfully. It does not indicate if the target port is available. Use the method ReadState to determine if the target port is available. (Inherited from IConnection [▶ 79].)
	IsDisposed [▶ 935]	Gets a value indicating whether this instance is disposed.
	IsLocal [▶ 896]	Gets a value indicating whether the ADS client is connected to a ADS Server on the local computer. (Inherited from IAdsConnection [▶ 876].)
	Session [▶ 82]	Gets the session that initiated this IConnection [▶ 79] (Inherited from IConnection [▶ 79].)
	Timeout [▶ 83]	Gets the timeout (in milliseconds) (Inherited from IConnection [▶ 79].)











Methods

















	Name	Description
	AddDeviceNotification(String, Int32, NotificationSettings, Object) [▶ 971]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object) [▶ 972]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationAsync(String, Int32, NotificationSettings, Object, CancellationTokens) [▶ 974]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationAsync(UInt32, UInt32, Int32,	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)










	Name	Description
	NotificationSettings, Object, CancellationToken [▶ 975]	
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type) [▶ 977]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32.) [▶ 977]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type) [▶ 978]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32.) [▶ 980]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive ('AnyType') types are allowed for the parameter type. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationExAsync(String, NotificationSettings, Object, Type, .Int32., CancellationToken) [▶ 981]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	AddDeviceNotificationExAsync(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., CancellationToken) [▶ 982]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	CleanupSymbolTable [▶ 1070]	Clears the internal symbol / DataTypes cache. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	Close [▶ 83]	Closes this IConnection [▶ 79] (Inherited from IConnection [▶ 79].)
	Connect . [▶ 84]	(Re)Connects the IConnection [▶ 79] when disconnected. (Inherited from IConnection [▶ 79].)
















	Name	Description
	Connect(AmsAddress) [▶ 873]	Connects the target ADS Device. (Inherited from IAdsConnectAddress [▶ 835].)
	Connect(Int32) [▶ 873]	Connects to the local target ADS Device. (Inherited from IAdsConnectAddress [▶ 835].)
	Connect(AmsNetId, Int32) [▶ 874]	Connects to the target ADS Device. (Inherited from IAdsConnectAddress [▶ 835].)
	Connect(String, Int32) [▶ 874]	Connects to the target ADS Device. (Inherited from IAdsConnectAddress [▶ 835].)
	CreateVariableHandle [▶ 956]	Determines the Symbol handle by its instance path synchronously. (Inherited from IAdsHandle [▶ 954].)
	CreateVariableHandleAsync [▶ 956]	Determines the Symbol handle by its instance path asynchronously. (Inherited from IAdsHandle [▶ 954].)
	DeleteDeviceNotification [▶ 983]	Deletes a registered notification. (Inherited from IAdsNotifications [▶ 966].)
	DeleteDeviceNotificationAsync [▶ 984]	Deletes a registered notification asynchronously. (Inherited from IAdsNotifications [▶ 966].)
	DeleteVariableHandle [▶ 957]	Releases the specified symbol/variable handle synchronously. (Inherited from IAdsHandle [▶ 954].)
	DeleteVariableHandleAsync [▶ 958]	Releases the specified symbol/variable handle asynchronously. (Inherited from IAdsHandle [▶ 954].)
	Disconnect [▶ 84]	Disconnects this IConnection [▶ 79]. (Inherited from IConnection [▶ 79].)
	Dispose	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from IDisposable.)
 	InvokeRpcMethodAsync(String, String, Object, CancellationToken) [▶ 1021]	Invokes the specified RPC Method asynchronously (Inherited from IAdsRpcInvoke [▶ 1012].)
 	InvokeRpcMethodAsync(IRpcCallableInstance, IRpcMethod, Object, AnyTypeSpecifier, AnyTypeSpecifier, CancellationToken) [▶ 1024]	Invokes the specified RPC Method asynchronously (Inherited from IAdsRpcInvoke [▶ 1012].)
 	InvokeRpcMethodAsync(String, String, Object, AnyTypeSpecifier, AnyTypeSpecifier) [▶ 1021]	Invokes the specified RPC Method asynchronously (Inherited from IAdsRpcInvoke [▶ 1012].)















	Name	Description
	AnyTypeSpecifier, CancellationToken [▶ 1022]	
	Read(UInt32, Memory.Byte.) [▶ 959]	Reads data synchronously from an ADS device and writes to the specified readBuffer. (Inherited from IAdsHandle [▶ 954].)
	Read(UInt32, UInt32, Memory.Byte.) [▶ 1002]	Reads data synchronously from an ADS device and writes it to the given readBuffer (Inherited from IAdsReadWrite2 [▶ 1000].)
	ReadAny(UInt32, Type) [▶ 812]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny(UInt32, Type, .Int32.) [▶ 813]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny(UInt32, UInt32, Type) [▶ 815]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny(UInt32, UInt32, Type, .Int32.) [▶ 815]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32) [▶ 810]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32, .Int32.) [▶ 811]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32, UInt32) [▶ 812]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32, UInt32, .Int32.) [▶ 814]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt32, Type, CancellationToken) [▶ 819]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt32, Type, .Int32., CancellationToken) [▶ 821]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt32, UInt32, Type, CancellationToken) [▶ 823]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt32, UInt32,	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)














	Name	Description
	Type, .Int32., CancellationToken) [▶ 823]	
	ReadAnyAsync.T. (UInt32, CancellationToken) [▶ 817]	Reads data asynchronously from an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T. (UInt32, .Int32., CancellationToken) [▶ 818]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T. (UInt32, UInt32, CancellationToken) [▶ 820]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T. (UInt32, UInt32, .Int32., CancellationToken) [▶ 822]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyString(UInt 32, Int32, Encoding) [▶ 825]	Reads a string from the specified symbol/variable. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyString(UInt 32, UInt32, Int32, Encoding) [▶ 825]	Reads as string from a specified address. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyStringAsyn c(UInt32, Int32, Encoding, CancellationToken) [▶ 827]	Reads a string asynchronously from the specified symbol/variable (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyStringAsyn c(UInt32, UInt32, Int32, Encoding, CancellationToken) [▶ 827]	Reads a string from a specified address asynchronously. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAsync(UInt32, Memory.Byte., CancellationToken) [▶ 959]	Reads the value data of the symbol asynchronously into the readBuffer. (Inherited from IAdsHandle [▶ 954].)
	ReadAsync(UInt32, UInt32, Memory.Byte., CancellationToken) [▶ 996]	Reads the data asynchronously from specified IndexGroup/IndexOffset (Inherited from IAdsReadWrite [▶ 995].)
	ReadDataType [▶ 1070]	Call this method to obtain information about the specified data type. (Inherited from IAdsSymbolicAccess [▶ 1065].)












	Name	Description
	ReadDataTypeAsync [▶ 1070]	Call this method to obtain information about the specified data type. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadDeviceInfo [▶ 914]	Reads the identification and version number of an ADS server. (Inherited from IAdsConnection [▶ 876].)
	ReadDeviceInfoAsync [▶ 914]	Reads the identification and version number of an ADS server. (Inherited from IAdsConnection [▶ 876].)
	ReadState [▶ 1058]	Reads the ADS status and the device status from an ADS server. (Inherited from IAdsStateProvider [▶ 1056].)
	ReadStateAsync [▶ 1058]	Reads the ADS status and the device status from an ADS server. (Inherited from IAdsStateProvider [▶ 1056].)
	ReadSymbol [▶ 1071]	Call this method to obtain information about the individual symbols (variables) in ADS devices. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadSymbolAsync [▶ 1072]	Call this method to obtain information about the individual symbols (variables) in ADS devices. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValue(ISymbol) [▶ 1073]	Reads the value of a symbol and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValue(String, Type) [▶ 1075]	Reads the value of a symbol and returns the value as object. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValue.T.(String) [▶ 1073]	Reads the value of a symbol and returns the value. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValueAsync(ISymbol, CancellationToken) [▶ 1077]	Reads the value of a symbol asynchronously and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValueAsync(String, Type, CancellationToken) [▶ 1079]	Reads the value of a symbol asynchronously. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValueAsync.T.(String, CancellationToken) [▶ 1076]	Reads the value of a symbol asynchronously. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadWrite(UInt32, Memory.Byte, ReadOnlyMemory.Byte.) [▶ 960]	Writes data synchronously to an ADS device and then Reads data from that target. (Inherited from IAdsHandle [▶ 954].)
	ReadWrite(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte.) [▶ 1002]	Writes data synchronously to an ADS device and then Reads data from this device into the readBuffer (Inherited from IAdsReadWrite2 [▶ 1000].)
	ReadWriteAsync(UInt32, Memory.Byte, ReadOnlyMemory.Byte)	ReadWrites value data asynchronously to/from the symbol represented by the variableHandle. (Inherited from IAdsHandle [▶ 954].)




	Name	Description
	yte, CancellationToken) [▶ 961]	
	ReadWriteAsync(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte, CancellationToken) [▶ 997]	Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer (Inherited from IAdsReadWrite [▶ 995].)
	RegisterAdsStateChangedAsync [▶ 1059]	Registers for AdsStateChanged [▶ 1061] events as an asynchronous operation. (Inherited from IAdsStateProvider [▶ 1056].)
	RegisterSymbolVersionChanged [▶ 1064]	Registers the symbol version changed. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	RegisterSymbolVersionChangedAsync [▶ 1062]	Registers the symbol version changed asynchronously. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	TryAddDeviceNotification(String, Int32, NotificationSettings, Object, UInt32.) [▶ 985]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)
	TryAddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.) [▶ 986]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)
	TryAddDeviceNotificationEx(String, NotificationSettings, Object, Type, Int32, UInt32.) [▶ 988]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	TryAddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, Int32, UInt32.) [▶ 989]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	TryCreateVariableHandle [▶ 961]	Determines the Symbol handle by its instance path synchronously. (Inherited from IAdsHandle [▶ 954].)

	Name	Description
	TryDeleteDeviceNotification [▶ 990]	Deletes a registered notification. (Inherited from IAdsNotifications [▶ 966].)
	TryDeleteVariableHandle [▶ 962]	Releases the specified symbol/variable handle synchronously. (Inherited from IAdsHandle [▶ 954].)
 	TryInvokeRpcMethod(String, String, Object, Object.) [▶ 1026]	Invokes the specified RPC Method (Inherited from IAdsRpcInvoke [▶ 1012].)
 	TryInvokeRpcMethod(String, String, Object, Object.) [▶ 1028]	Invokes the specified RPC Method (Inherited from IAdsRpcInvoke [▶ 1012].)
 	TryInvokeRpcMethod(IRpcCallableInstance, IRpcMethod, Object, AnyTypeSpecifier, AnyTypeSpecifier, Object, Object.) [▶ 1031]	Invokes the rpc method. (Inherited from IAdsRpcInvoke [▶ 1012].)
 	TryInvokeRpcMethod(String, String, Object, AnyTypeSpecifier, AnyTypeSpecifier, Object, Object.) [▶ 1030]	Invokes the rpc method. (Inherited from IAdsRpcInvoke [▶ 1012].)
	TryRead(UInt32, Memory.Byte, Int32.) [▶ 963]	Reads the value synchronously data of the symbol, that is represented by the variable handle into the readBuffer. (Inherited from IAdsHandle [▶ 954].)
	TryRead(UInt32, UInt32, Memory.Byte, Int32.) [▶ 998]	Reads data synchronously from an ADS device and writes it to the given stream. (Inherited from IAdsReadWrite [▶ 995].)
	TryReadDataType [▶ 1079]	Call this method to obtain information about the specified data type. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadState [▶ 1059]	Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful. (Inherited from IAdsStateProvider [▶ 1056].)
	TryReadSymbol [▶ 1080]	Call this method to obtain information about the individual symbols (variables) in ADS devices. (Inherited from IAdsSymbolicAccess [▶ 1065].)









	Name	Description
	TryReadValue(ISymbol, Object.) [▶ 1082]	Reads the value of a symbol and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadValue(String, Type, Object.) [▶ 1083]	Reads the value of a symbol and returns the value as object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadValue.T.(ISymbol, T.) [▶ 1083]	Reads the value of a symbol and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadValue.T.(String, T.) [▶ 1081]	Reads the value of a symbol and returns the value as object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadWrite(UInt32, Memory.Byte, ReadOnlyMemory.Byte, Int32.) [▶ 963]	ReadWrites value data synchronously to/from the symbol represented by the variableHandle. (Inherited from IAdsHandle [▶ 954].)
	TryReadWrite(UInt32, UInt32, Memory.Byte, ReadOnlyMemory.Byte, Int32.) [▶ 998]	Writes data synchronously to an ADS device and reads data from that device. (Inherited from IAdsReadWrite [▶ 995].)
	TryWrite(UInt32, ReadOnlyMemory.Byte.) [▶ 964]	Writes the value data synchronously that is represented in the writeBuffer to the symbol with the specified variableHandle. (Inherited from IAdsHandle [▶ 954].)
	TryWrite(UInt32, UInt32, ReadOnlyMemory.Byte.) [▶ 999]	Writes data synchronously to an ADS device. (Inherited from IAdsReadWrite [▶ 995].)
	TryWriteControl(StateInfo) [▶ 1042]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	TryWriteControl(StateInfo, ReadOnlyMemory.Byte.) [▶ 1043]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	TryWriteValue(ISymbol, Object) [▶ 1086]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryWriteValue(String, Object) [▶ 1085]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryWriteValue.T.(ISymbol, T) [▶ 1087]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryWriteValue.T.(String, T) [▶ 1085]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)

	Name	Description
	UnregisterAdsState ChangedAsync [▶ 1060]	Registers for AdsStateChanged [▶ 1061] events as an asynchronous operation. (Inherited from IAdsStateProvider [▶ 1056].)
	UnregisterSymbolV ersionChanged [▶ 1064]	Unregisters the symbol version changed. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	UnregisterSymbolV ersionChangedAsyn c [▶ 1063]	Unregisters the symbol version changed asynchronously. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	Write(UInt32, ReadOnlyMemory.B yte.) [▶ 964]	Writes data synchronously to an ADS device. (Inherited from IAdsHandle [▶ 954].)
	Write(UInt32, UInt32) [▶ 1003]	Triggers a 'Write' call to the ADS device at the specified address. (Inherited from IAdsReadWrite2 [▶ 1000].)
	Write(UInt32, UInt32, ReadOnlyMemory.B yte.) [▶ 1004]	Writes data synchronously to an ADS device. (Inherited from IAdsReadWrite2 [▶ 1000].)
	WriteAny(UInt32, Object) [▶ 829]	Writes an object synchronously to an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAny(UInt32, Object, .Int32.) [▶ 829]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAny(UInt32, UInt32, Object) [▶ 830]	Writes an object synchronously to an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAny(UInt32, UInt32, Object, .Int32.) [▶ 830]	Writes an object synchronously to an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt 32, Object, CancellationTok en) [▶ 832]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt 32, Object, .Int32, CancellationTok en) [▶ 833]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt 32, UInt32, Object, CancellationTok en) [▶ 834]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)



	Name	Description
	WriteAnyAsync(UInt32, UInt32, Object, Int32, CancellationToken) [▶ 834]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAsync(UInt32, ReadOnlyMemory.BYTE, CancellationToken) [▶ 965]	Writes the value data asynchronously that is represented by the writeBuffer to the symbol specified by the variableHandle. (Inherited from IAdsHandle [▶ 954].)
	WriteAsync(UInt32, UInt32, ReadOnlyMemory.BYTE, CancellationToken) [▶ 1000]	Writes the data / Value asynchronously into the specified writeBuffer. (Inherited from IAdsReadWrite [▶ 995].)
	WriteControl(StateInfo) [▶ 1044]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	WriteControl(StateInfo, ReadOnlyMemory.BYTE) [▶ 1045]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	WriteControlAsync(AdsState, UInt16, CancellationToken) [▶ 1046]	Changes the ADS status and device status of the ADS server asynchronously. (Inherited from IAdsStateControl [▶ 1039].)
	WriteControlAsync(AdsState, UInt16, ReadOnlyMemory.BYTE, CancellationToken) [▶ 1047]	Changes the ADS status and device status of the ADS server asynchronously. (Inherited from IAdsStateControl [▶ 1039].)
	WriteValue(ISymbol, Object) [▶ 1089]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValue(String, Object) [▶ 1088]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValue.T.(ISymbol, T) [▶ 1090]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValue.T.(String, T) [▶ 1089]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)

	Name	Description
	WriteValueAsync(ISymbol, Object, CancellationToken) [▶ 1092]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValueAsync.T.(ISymbol, T, CancellationToken) [▶ 1093]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValueAsync.T.(String, T, CancellationToken) [▶ 1091]	Writes the passed object value to the specified ADS symbol.The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)

Events

	Name	Description
	AdsNotification [▶ 991]	Occurs when the ADS device sends a notification to the client. (Inherited from IAdsNotifications [▶ 966].)
	AdsNotificationError [▶ 993]	Occurs when a exception has occurred during notification management. (Inherited from IAdsNotifications [▶ 966].)
	AdsNotificationEx [▶ 993]	Occurs when the ADS devices sends a notification to the client. (Inherited from IAdsNotifications [▶ 966].)
	AdsStateChanged [▶ 1061]	Occurs when the AdsState of the target system has been changed. (Inherited from IAdsStateProvider [▶ 1056].)
	AdsSumNotification [▶ 994]	Occurs when Notifications are send (bundled notifications) (Inherited from IAdsNotifications [▶ 966].)
	AdsSymbolVersionChanged [▶ 1065]	Occurs when the ADS Symbol Version changed. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	ConnectionStateChanged [▶ 91]	Occurs when connection status of the IConnectionStateProvider [▶ 89] has been changed. (Inherited from IConnectionStateProvider [▶ 89].)
	RouterStateChange [▶ 1100]	(Local) Router state changed event. (Inherited from IRouterNotificationProvider [▶ 1099].)


Extension Methods

	Name	Description
	PollAdsState(IObservable.Unit.) [▶ 3294]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
	PollAdsState(IObservable.Unit.) [▶ 1256]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)

	Name	Description
 	PollAdsState(TimeSpan) [▸ 3295]	Overloaded. Gets an observable sequence of ResultReadAdsState [▸ 1146]s via Polling. (Defined by ConnectionStateExtension [▸ 3292].)
 	PollAdsState(TimeSpan) [▸ 1257]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
 	PollAdsState2(IObservable.Unit.) [▸ 1271]	Overloaded. Gets an observable sequence of ResultReadAdsState [▸ 1146]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
 	PollAdsState2(TimeSpan) [▸ 1272]	Overloaded. Gets an observable sequence of ResultReadAdsState [▸ 1146]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
 	PollAdsState2Async(IObservable.Unit., Cancellation.Token) [▸ 1274]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
 	PollAdsState2Async(TimeSpan, Cancellation.Token) [▸ 1275]	Overloaded. Gets an observable sequence of ResultReadAdsState [▸ 1146]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
 	PollAdsStateAsync(IObservable.Unit., Cancellation.Token) [▸ 3296]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by ConnectionStateExtension [▸ 3292].)
 	PollAdsStateAsync(IObservable.Unit., Cancellation.Token) [▸ 1259]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
 	PollAdsStateAsync(TimeSpan, Cancellation.Token) [▸ 3298]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by ConnectionStateExtension [▸ 3292].)
 	PollAdsStateAsync(TimeSpan, Cancellation.Token) [▸ 1260]	Overloaded. Gets an observable sequence of AdsState [▸ 729]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
 	PollDeviceState(IObservable.Unit.) [▸ 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▸ 1157]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)
 	PollDeviceState(TimeSpan) [▸ 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▸ 1157]s via Polling. (Defined by AdsClientExtensions [▸ 1249].)









	Name	Description
	PollDeviceStateAsync(IObservable.Unit., Cancellation.Token) [▶ 1278]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157] s via Polling. (Defined by AdsClientExtensions [▶ 1249] .)
	PollDeviceStateAsync(TimeSpan, Cancellation.Token) [▶ 1279]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157] s via Polling. (Defined by AdsClientExtensions [▶ 1249] .)
	PollSystemServiceState [▶ 3282]	Polls the state of the system service. (Defined by SystemServiceExtension [▶ 3281] .)
	PollSystemServiceStateAsync [▶ 3283]	Polls the system service state asynchronously (Defined by SystemServiceExtension [▶ 3281] .)
	PollValues(ISymbol, Type, IObservable.Unit.) [▶ 1310]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(ISymbol, Type, TimeSpan) [▶ 1311]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(String, Type, IObservable.Unit.) [▶ 1300]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(String, Type, TimeSpan) [▶ 1301]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(ISymbol, Type, .Int32., TimeSpan) [▶ 1316]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(ISymbol, Type, IObservable.Unit., Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(ISymbol, Type, TimeSpan, Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(String, Type, .Int32., TimeSpan) [▶ 1304]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281] .)
	PollValues(String, Type, IObservable.Unit., Func.Exception, Object.) [▶ 1305]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281] .)

	Name	Description
	PollValues(String, Type, TimeSpan, Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32., IObservable.Unit, Func.Exception, Object.) [▶ 1321]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32., TimeSpan, Func.Exception, Object.) [▶ 1322]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32., IObservable.Unit, Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32., TimeSpan, Func.Exception, Object.) [▶ 1307]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol, IObservable.Unit.) [▶ 1308]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol, TimeSpan) [▶ 1309]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, IObservable.Unit.) [▶ 1294]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, TimeSpan) [▶ 1295]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol, IObservable.Unit., Func.Exception, T.) [▶ 1314]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol, TimeSpan, Func.Exception, T.) [▶ 1315]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol, .Int32., IObservable.Unit.) [▶ 1312]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	PollValues.T. (ISymbol , .Int32 , TimeSpan) [▶ 1313]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, IObservable.Unit, Func.Exception, T.) [▶ 1298]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, TimeSpan, Func.Exception, T.) [▶ 1299]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32 , IObservable.Unit .) [▶ 1296]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32 , TimeSpan) [▶ 1297]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol , .Int32 , IObservable.Unit , Func.Exception, T.) [▶ 1318]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (ISymbol , .Int32 , TimeSpan , Func.Exception, T.) [▶ 1319]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32 , IObservable.Unit , Func.Exception, T.) [▶ 1302]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T. (String , .Int32 , TimeSpan , Func.Exception, T.) [▶ 1303]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues2.T. (ISymbol , .Int32 , IObservable.Unit .) [▶ 1330]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues2.T. (String , .Int32 , IObservable.Unit .) [▶ 1329]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	ReadSysServState [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (Defined by SystemServiceExtension [▶ 3281].)

	Name	Description
	ReadSysServStateAsync [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (asynchronous) (Defined by SystemServiceExtension [▶ 3281].)
	ReadWithFallback(UInt32, UInt32, Memory.Byte., UInt32, Boolean.) [▶ 1219]	Overloaded. Ads Read with fallback. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallback(UInt32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., Boolean.) [▶ 1220]	Overloaded. Ads Read with Fallback. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallbackAsync(UInt32, UInt32, UInt32, Memory.Byte., CancellationToken) [▶ 1221]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallbackAsync(UInt32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., CancellationToken) [▶ 1222]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedRead(UInt32, UInt32, Memory.Byte., Int32, TimeSpan) [▶ 1224]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedRead(UInt32, UInt32, Memory.Byte., Int32, TimeSpan, Func.ResultRead, Boolean.) [▶ 1225]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedReadAsync(UInt32, UInt32, Memory.Byte., Int32, TimeSpan, CancellationToken) [▶ 1226]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)

	Name	Description
	RepeatedReadAsync (UInt32, UInt32, Memory.Byte., Int32, TimeSpan, Func.ResultRead, Boolean., CancellationToken) [1227]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [1215].)
	RestartTwinCATAsync [3285]	Sends a SystemService Restart to the target system and waits until the system has finished the state changes (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [3281].)
	SetAdsState [3299]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [3292].)
	SetAdsStateAsync [3300]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [3292].)
	WaitUntilRestarted [3286]	Waits until the Restart is detected on the client (SystemService, Port 10000) (Defined by SystemServiceExtension [3281].)
	WaitUntilRestartedAsync [3287]	Waits until the SystemService Restart is detected on the client (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [3281].)
 	WhenAdsStateChanges [1261]	Gets an observable sequence of AdsState [729]s. (Defined by AdsClientExtensions [1249].)
	WhenNotification(ISymbol) [1263]	Overloaded. Gets an observable sequence of Notification [1100]s. (Defined by AdsClientExtensions [1249].)
 	WhenNotification(ISymbolCollection) [1264]	Overloaded. Gets an observable sequence of Notification [1100] objects. (Defined by AdsClientExtensions [1249].)
 	WhenNotification(IList.ISymbol, NotificationSettings) [1268]	Overloaded. Gets an observable sequence of Notification [1100] objects. (Defined by AdsClientExtensions [1249].)
	WhenNotification(ISymbol, NotificationSettings) [1265]	Overloaded. Gets an observable sequence of SymbolValueNotification [1331]s. (Defined by AdsClientExtensions [1249].)
 	WhenNotification(String, Type, NotificationSettings) [1324]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [1281].)
 	WhenNotification.T.(String, NotificationSettings) [1323]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [1281].)

	Name	Description
	WhenSymbolVersionChanges [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
	WhenSymbolVersionChanges(IScheduler) [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenValueChanged [▶ 1345]	Observable sequence of Values driven by ADS Notifications on the specified symbol. (Defined by ValueSymbolExtensions [▶ 1333].)
 	WriteValues.T.(String, IObservable.T.) [▶ 1326]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)
 	WriteValues.T.(String, IObservable.T., Action.Exception.) [▶ 1327]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)


Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[TwinCAT.Ads.IAdsConnectAddress](#) [[▶ 835](#)]

[System.IDisposable](#)







Also see about this






-  [AdsClientExtensions.WhenNotification Method \(IAdsConnection, ISymbolCollection, NotificationSettings\)](#) [[▶ 1266](#)]

6.2.33.1 IAdsDisposableConnection Properties

The [IAdsDisposableConnection](#) [[▶ 915](#)] type exposes the following members.

Properties

	Name	Description
	Address [▶ 895]	Gets the AmsAddress [▶ 752] of the ADS server. (Inherited from IAdsConnection [▶ 876].)
	ClientAddress [▶ 895]	Get the AmsAddress [▶ 752] of the ADS client. (Inherited from IAdsConnection [▶ 876].)
 	ConnectionState [▶ 90]	Gets the current Connection state of the IConnectionStateProvider [▶ 89] (Inherited from IConnectionStateProvider [▶ 89].)
	DefaultValueEncoding [▶ 81]	Gets the default value encoding. (Inherited from IConnection [▶ 79].)
	Id [▶ 81]	Gets the Connection Identifier . (Inherited from IConnection [▶ 79].)

	Name	Description
	IsConnected [▶ 82]	Gets a value indicating whether the local ADS port was opened successfully. It does not indicate if the target port is available. Use the method <code>ReadState</code> to determine if the target port is available. (Inherited from IConnection [▶ 79].)
	IsDisposed [▶ 935]	Gets a value indicating whether this instance is disposed.
	IsLocal [▶ 896]	Gets a value indicating whether the ADS client is connected to a ADS Server on the local computer. (Inherited from IAdsConnection [▶ 876].)
	Session [▶ 82]	Gets the session that initiated this IConnection [▶ 79] (Inherited from IConnection [▶ 79].)
	Timeout [▶ 83]	Gets the timeout (in milliseconds) (Inherited from IConnection [▶ 79].)

Reference

[IAdsDisposableConnection Interface](#) [[▶ 915](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.33.1 IAdsDisposableConnection.IsDisposed Property

Gets a value indicating whether this instance is disposed.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsDisposed { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is disposed; otherwise, false.

Reference


[IAdsDisposableConnection Interface](#) [[▶ 915](#)]









[TwinCAT.Ads Namespace](#) [[▶ 179](#)]


















6.2.33.2 IAdsDisposableConnection Methods
















The [IAdsDisposableConnection](#) [[▶ 915](#)] type exposes the following members.












Methods

















	Name	Description
	AddDeviceNotification(String, Int32, NotificationSettings, Object) [▶ 971]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)









	Name	Description
	AddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object) [► 972]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. (Inherited from IAdsNotifications [► 966].)
	AddDeviceNotificationAsync(String, Int32, NotificationSettings, Object, CancellationToken) [► 974]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [► 991] event. (Inherited from IAdsNotifications [► 966].)
	AddDeviceNotificationAsync(UInt32, UInt32, Int32, NotificationSettings, Object, CancellationToken) [► 975]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [► 991] event. (Inherited from IAdsNotifications [► 966].)
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type) [► 977]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [► 993] event. (Inherited from IAdsNotifications [► 966].)
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32.) [► 977]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [► 993] event. (Inherited from IAdsNotifications [► 966].)
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type) [► 978]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [► 993] event. (Inherited from IAdsNotifications [► 966].)
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32.) [► 980]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive ('AnyType') types are allowed for the parameter type. (Inherited from IAdsNotifications [► 966].)
	AddDeviceNotificationExAsync(String, NotificationSettings,	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [► 993] event. (Inherited from IAdsNotifications [► 966].)














	Name	Description
	Object, Type, .Int32., CancellationToken [▶ 981]	
	AddDeviceNotificationExAsync(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., CancellationToken) [▶ 982]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	CleanupSymbolTable [▶ 1070]	Clears the internal symbol / DataTypes cache. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	Close [▶ 83]	Closes this IConnection [▶ 79] (Inherited from IConnection [▶ 79].)
	Connect. [▶ 84]	(Re)Connects the IConnection [▶ 79] when disconnected. (Inherited from IConnection [▶ 79].)
	Connect(AmsAddress) [▶ 873]	Connects the target ADS Device. (Inherited from IAdsConnectAddress [▶ 835].)
	Connect(Int32) [▶ 873]	Connects to the local target ADS Device. (Inherited from IAdsConnectAddress [▶ 835].)
	Connect(AmsNetId, Int32) [▶ 874]	Connects to the target ADS Device. (Inherited from IAdsConnectAddress [▶ 835].)
	Connect(String, Int32) [▶ 874]	Connects to the target ADS Device. (Inherited from IAdsConnectAddress [▶ 835].)
	CreateVariableHandle [▶ 956]	Determines the Symbol handle by its instance path synchronously. (Inherited from IAdsHandle [▶ 954].)
	CreateVariableHandleAsync [▶ 956]	Determines the Symbol handle by its instance path asynchronously. (Inherited from IAdsHandle [▶ 954].)
	DeleteDeviceNotification [▶ 983]	Deletes a registered notification. (Inherited from IAdsNotifications [▶ 966].)
	DeleteDeviceNotificationAsync [▶ 984]	Deletes a registered notification asynchronously. (Inherited from IAdsNotifications [▶ 966].)
	DeleteVariableHandle [▶ 957]	Releases the specified symbol/variable handle synchronously. (Inherited from IAdsHandle [▶ 954].)
	DeleteVariableHandleAsync [▶ 958]	Releases the specified symbol/variable handle asynchronously. (Inherited from IAdsHandle [▶ 954].)
	Disconnect [▶ 84]	Disconnects this IConnection [▶ 79]. (Inherited from IConnection [▶ 79].)
	Dispose	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from IDisposable .)
	InvokeRpcMethodAsync(String,	Invokes the specified RPC Method asynchronously (Inherited from IAdsRpcInvoke [▶ 1012].)















	Name	Description
	String, .Object, CancellationTok en) [▶ 1021]	
 	InvokeRpcMethodA sync(IRpcCallablIns tance, IRpcMethod, .Objec t, .AnyTypeSpecifier , .AnyTypeSpecifier, CancellationToken) [▶ 1024]	Invokes the specified RPC Method asynchronously (Inherited from IAdsRpcInvoke [▶ 1012].)
 	InvokeRpcMethodA sync(String, String, .Object, .Any TypeSpecifier, AnyTypeSpecifier, CancellationToken) [▶ 1022]	Invokes the specified RPC Method asynchronously (Inherited from IAdsRpcInvoke [▶ 1012].)
	Read(UInt32, Memory.Byte.) [▶ 959]	Reads data synchronously from an ADS device and writes to the specified readBuffer. (Inherited from IAdsHandle [▶ 954].)
	Read(UInt32, UInt32, Memory.Byte.) [▶ 1002]	Reads data synchronously from an ADS device and writes it to the given readBuffer (Inherited from IAdsReadWrite2 [▶ 1000].)
	ReadAny(UInt32, Type) [▶ 812]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny(UInt32, Type, .Int32.) [▶ 813]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny(UInt32, UInt32, Type) [▶ 815]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny(UInt32, UInt32, Type, .Int32.) [▶ 815]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32) [▶ 810]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T. (UInt32, .Int32.) [▶ 811]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32, UInt32) [▶ 812]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAny.T.(UInt32, UInt32, .Int32.) [▶ 814]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)















	Name	Description
	ReadAnyAsync(UInt32, Type, CancellationToken) [▶ 819]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt32, Type, .Int32., CancellationToken) [▶ 821]	Reads data synchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt32, UInt32, Type, CancellationToken) [▶ 823]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync(UInt32, UInt32, Type, .Int32., CancellationToken) [▶ 823]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T.(UInt32, CancellationToken) [▶ 817]	Reads data asynchronously from an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T.(UInt32, .Int32., CancellationToken) [▶ 818]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T.(UInt32, UInt32, CancellationToken) [▶ 820]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyAsync.T.(UInt32, UInt32, .Int32., CancellationToken) [▶ 822]	Reads data asynchronously from an ADS device and writes it to an object. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyString(UInt32, Int32, Encoding) [▶ 825]	Reads a string from the specified symbol/variable. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyString(UInt32, UInt32, Int32, Encoding) [▶ 825]	Reads as string from a specified address. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAnyStringAsync(UInt32, Int32, Encoding, CancellationToken) [▶ 827]	Reads a string asynchronously from the specified symbol/variable (Inherited from IAdsAnyAccess [▶ 804].)












	Name	Description
	ReadAnyStringAsync(UInt32, UInt32, Int32, Encoding, CancellationToken) [▶ 827]	Reads a string from a specified address asynchronously. (Inherited from IAdsAnyAccess [▶ 804].)
	ReadAsync(UInt32, Memory.Byte., CancellationToken) [▶ 959]	Reads the value data of the symbol asynchronously into the readBuffer. (Inherited from IAdsHandle [▶ 954].)
	ReadAsync(UInt32, UInt32, Memory.Byte., CancellationToken) [▶ 996]	Reads the data asynchronously from specified IndexGroup/IndexOffset (Inherited from IAdsReadWrite [▶ 995].)
	ReadDataType [▶ 1070]	Call this method to obtain information about the specified data type. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadDataTypeAsync [▶ 1070]	Call this method to obtain information about the specified data type. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadDeviceInfo [▶ 914]	Reads the identification and version number of an ADS server. (Inherited from IAdsConnection [▶ 876].)
	ReadDeviceInfoAsync [▶ 914]	Reads the identification and version number of an ADS server. (Inherited from IAdsConnection [▶ 876].)
	ReadState [▶ 1058]	Reads the ADS status and the device status from an ADS server. (Inherited from IAdsStateProvider [▶ 1056].)
	ReadStateAsync [▶ 1058]	Reads the ADS status and the device status from an ADS server. (Inherited from IAdsStateProvider [▶ 1056].)
	ReadSymbol [▶ 1071]	Call this method to obtain information about the individual symbols (variables) in ADS devices. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadSymbolAsync [▶ 1072]	Call this method to obtain information about the individual symbols (variables) in ADS devices. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValue(ISymbol) [▶ 1073]	Reads the value of a symbol and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValue(String, Type) [▶ 1075]	Reads the value of a symbol and returns the value as object. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValue.T.(String) [▶ 1073]	Reads the value of a symbol and returns the value. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValueAsync(ISymbol, CancellationToken) [▶ 1077]	Reads the value of a symbol asynchronously and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadValueAsync(String, Type, CancellationToken) [▶ 1079]	Reads the value of a symbol asynchronously. (Inherited from IAdsSymbolicAccess [▶ 1065].)








	Name	Description
	ReadValueAsync.T. (String, CancellationToken) [▶ 1076]	Reads the value of a symbol asynchronously. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	ReadWrite(UInt32, Memory.Byte., ReadOnlyMemory.B yte.) [▶ 960]	Writes data synchronously to an ADS device and then Reads data from that target. (Inherited from IAdsHandle [▶ 954].)
	ReadWrite(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.B yte.) [▶ 1002]	Writes data synchronously to an ADS device and then Reads data from this device into the readBuffer (Inherited from IAdsReadWrite2 [▶ 1000].)
	ReadWriteAsync(UIn t32, Memory.Byte., ReadOnlyMemory.B yte, CancellationToken) [▶ 961]	ReadWrites value data asynchronously to/from the symbol represented by the variableHandle. (Inherited from IAdsHandle [▶ 954].)
	ReadWriteAsync(UIn t32, UInt32, Memory.Byte., ReadOnlyMemory.B yte, CancellationToken) [▶ 997]	Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer (Inherited from IAdsReadWrite [▶ 995].)
	RegisterAdsStateCh angedAsync [▶ 1059]	Registers for AdsStateChanged [▶ 1061] events as an asynchronous operation. (Inherited from IAdsStateProvider [▶ 1056].)
	RegisterSymbolVers ionChanged [▶ 1064]	Registers the symbol version changed. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	RegisterSymbolVers ionChangedAsync [▶ 1062]	Registers the symbol version changed asynchronously. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	TryAddDeviceNotifi cation(String, Int32, NotificationSettings, Object, UInt32.) [▶ 985]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)
	TryAddDeviceNotifi cation(UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.) [▶ 986]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event. (Inherited from IAdsNotifications [▶ 966].)

	Name	Description
	TryAddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32., UInt32.) [▶ 988]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	TryAddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., UInt32.) [▶ 989]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event. (Inherited from IAdsNotifications [▶ 966].)
	TryCreateVariableHandle [▶ 961]	Determines the Symbol handle by its instance path synchronously. (Inherited from IAdsHandle [▶ 954].)
	TryDeleteDeviceNotification [▶ 990]	Deletes a registered notification. (Inherited from IAdsNotifications [▶ 966].)
	TryDeleteVariableHandle [▶ 962]	Releases the specified symbol/variable handle synchronously. (Inherited from IAdsHandle [▶ 954].)
 	TryInvokeRpcMethod(String, String, .Object, Object.) [▶ 1026]	Invokes the specified RPC Method (Inherited from IAdsRpcInvoke [▶ 1012].)
 	TryInvokeRpcMethod(String, String, .Object, .Object, Object.) [▶ 1028]	Invokes the specified RPC Method (Inherited from IAdsRpcInvoke [▶ 1012].)
 	TryInvokeRpcMethod(IRpcCallableInstance, IRpcMethod, .Object, .AnyTypeSpecifier, .AnyTypeSpecifier, .Object, Object.) [▶ 1031]	Invokes the rpc method. (Inherited from IAdsRpcInvoke [▶ 1012].)
 	TryInvokeRpcMethod(String, String, .Object, .AnyTypeSpecifier, .AnyTypeSpecifier, .Object, Object.) [▶ 1030]	Invokes the rpc method. (Inherited from IAdsRpcInvoke [▶ 1012].)






	Name	Description
	TryRead(UInt32, Memory.Byte., Int32.) [▶ 963]	Reads the value synchronously data of the symbol, that is represented by the variable handle into the readBuffer. (Inherited from IAdsHandle [▶ 954].)
	TryRead(UInt32, UInt32, Memory.Byte., Int32.) [▶ 998]	Reads data synchronously from an ADS device and writes it to the given stream. (Inherited from IAdsReadWrite [▶ 995].)
	TryReadDataType [▶ 1079]	Call this method to obtain information about the specified data type. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadState [▶ 1059]	Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful. (Inherited from IAdsStateProvider [▶ 1056].)
	TryReadSymbol [▶ 1080]	Call this method to obtain information about the individual symbols (variables) in ADS devices. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadValue(ISymbol, Object.) [▶ 1082]	Reads the value of a symbol and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadValue(String, Type, Object.) [▶ 1083]	Reads the value of a symbol and returns the value as object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadValue.T.(ISymbol, T.) [▶ 1083]	Reads the value of a symbol and returns it as an object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadValue.T.(String, T.) [▶ 1081]	Reads the value of a symbol and returns the value as object. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryReadWrite(UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.) [▶ 963]	ReadWrites value data synchronously to/from the symbol represented by the variableHandle. (Inherited from IAdsHandle [▶ 954].)
	TryReadWrite(UInt32, UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.) [▶ 998]	Writes data synchronously to an ADS device and reads data from that device. (Inherited from IAdsReadWrite [▶ 995].)
	TryWrite(UInt32, ReadOnlyMemory.Byte.) [▶ 964]	Writes the value data synchronously that is represented in the writeBuffer to the symbol with the specified variableHandle. (Inherited from IAdsHandle [▶ 954].)
	TryWrite(UInt32, UInt32, ReadOnlyMemory.Byte.) [▶ 999]	Writes data synchronously to an ADS device. (Inherited from IAdsReadWrite [▶ 995].)
	TryWriteControl(StateInfo) [▶ 1042]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)

	Name	Description
	TryWriteControl(Sta telInfo, ReadOnlyMemory.B yte.) [▶ 1043]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	TryWriteValue(ISym bol, Object) [▶ 1086]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryWriteValue(Strin g, Object) [▶ 1085]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryWriteValue.T. (ISymbol, T) [▶ 1087]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	TryWriteValue.T. (String, T) [▶ 1085]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	UnregisterAdsState ChangedAsync [▶ 1060]	Registers for AdsStateChanged [▶ 1061] events as an asynchronous operation. (Inherited from IAdsStateProvider [▶ 1056].)
	UnregisterSymbolV ersionChanged [▶ 1064]	Unregisters the symbol version changed. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	UnregisterSymbolV ersionChangedAsyn c [▶ 1063]	Unregisters the symbol version changed asynchronous. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
	Write(UInt32, ReadOnlyMemory.B yte.) [▶ 964]	Writes data synchronously to an ADS device. (Inherited from IAdsHandle [▶ 954].)
	Write(UInt32, UInt32) [▶ 1003]	Triggers a 'Write' call to the ADS device at the specified address. (Inherited from IAdsReadWrite2 [▶ 1000].)
	Write(UInt32, UInt32, ReadOnlyMemory.B yte.) [▶ 1004]	Writes data synchronously to an ADS device. (Inherited from IAdsReadWrite2 [▶ 1000].)
	WriteAny(UInt32, Object) [▶ 829]	Writes an object synchronously to an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAny(UInt32, Object, .Int32.) [▶ 829]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAny(UInt32, UInt32, Object) [▶ 830]	Writes an object synchronously to an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)














	Name	Description
	WriteAny(UInt32, UInt32, Object, .Int32.) [▶ 830]	Writes an object synchronously to an ADS device. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt32, Object, CancellationToken) [▶ 832]	Writes an object synchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt32, Object, .Int32, CancellationToken) [▶ 833]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt32, UInt32, Object, CancellationToken) [▶ 834]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAnyAsync(UInt32, UInt32, Object, .Int32, CancellationToken) [▶ 834]	Writes an object asynchronously to an ADS device. If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. (Inherited from IAdsAnyAccess [▶ 804].)
	WriteAsync(UInt32, ReadOnlyMemory.Byte, CancellationToken) [▶ 965]	Writes the value data asynchronously that is represented by the writeBuffer to the symbol specified by the variableHandle. (Inherited from IAdsHandle [▶ 954].)
	WriteAsync(UInt32, UInt32, ReadOnlyMemory.Byte, CancellationToken) [▶ 1000]	Writes the data / Value asynchronously into the specified writeBuffer. (Inherited from IAdsReadWrite [▶ 995].)
	WriteControl(StateInfo) [▶ 1044]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	WriteControl(StateInfo, ReadOnlyMemory.Byte.) [▶ 1045]	Changes the ADS status and the device status of an ADS server. (Inherited from IAdsStateControl [▶ 1039].)
	WriteControlAsync(AdsState, UInt16, CancellationToken) [▶ 1046]	Changes the ADS status and device status of the ADS server asynchronously. (Inherited from IAdsStateControl [▶ 1039].)
	WriteControlAsync(AdsState, UInt16,	Changes the ADS status and device status of the ADS server asynchronously. (Inherited from IAdsStateControl [▶ 1039].)









	Name	Description
	ReadOnlyMemory.Byte, CancellationToken [▶ 1047]	
	WriteValue(ISymbol, Object) [▶ 1089]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValue(String, Object) [▶ 1088]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValue.T.(ISymbol, T) [▶ 1090]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValue.T.(String, T) [▶ 1089]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValueAsync(ISymbol, Object, CancellationToken) [▶ 1092]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValueAsync.T.(ISymbol, T, CancellationToken) [▶ 1093]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)
	WriteValueAsync.T.(String, T, CancellationToken) [▶ 1091]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol. (Inherited from IAdsSymbolicAccess [▶ 1065].)








Extension Methods








	Name	Description
	PollAdsState(IObservable.Unit.) [▶ 3294]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
	PollAdsState(IObservable.Unit.) [▶ 1256]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState(TimeSpan) [▶ 3295]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
	PollAdsState(TimeSpan) [▶ 1257]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
	PollAdsState2(IObservable.Unit.) [▶ 1271]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)




	Name	Description
 	PollAdsState2(TimeSpan) [▶ 1272]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsState2Async(IObservable.Unit., CancellationTokens) [▶ 1274]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsState2Async(TimeSpan, CancellationTokens) [▶ 1275]	Overloaded. Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsStateAsync(IObservable.Unit., CancellationTokens) [▶ 3296]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
 	PollAdsStateAsync(IObservable.Unit., CancellationTokens) [▶ 1259]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollAdsStateAsync(TimeSpan, CancellationTokens) [▶ 3298]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by ConnectionStateExtension [▶ 3292].)
 	PollAdsStateAsync(TimeSpan, CancellationTokens) [▶ 1260]	Overloaded. Gets an observable sequence of AdsState [▶ 729]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollDeviceState(IObservable.Unit.) [▶ 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollDeviceState(TimeSpan) [▶ 1276]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollDeviceStateAsync(IObservable.Unit., CancellationTokens) [▶ 1278]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)
 	PollDeviceStateAsync(TimeSpan, CancellationTokens) [▶ 1279]	Overloaded. Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling. (Defined by AdsClientExtensions [▶ 1249].)






	Name	Description
	PollSystemServiceState [▶ 3282]	Polls the state of the system service. (Defined by SystemServiceExtension [▶ 3281].)
	PollSystemServiceStateAsync [▶ 3283]	Polls the system service state asynchronously (Defined by SystemServiceExtension [▶ 3281].)
	PollValues(ISymbol, Type, IObservable.Unit.) [▶ 1310]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, TimeSpan) [▶ 1311]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, IObservable.Unit.) [▶ 1300]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, TimeSpan) [▶ 1301]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32., TimeSpan) [▶ 1316]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, IObservable.Unit, Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, TimeSpan, Func.Exception, Object.) [▶ 1317]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32., TimeSpan) [▶ 1304]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, IObservable.Unit, Func.Exception, Object.) [▶ 1305]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, TimeSpan, Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(ISymbol, Type, .Int32., IObservable.Unit, Func.Exception, Object.) [▶ 1321]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	PollValues(ISymbol, Type, .Int32., TimeSpan, Func.Exception, Object.) [▶ 1322]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32., IObservable.Unit., Func.Exception, Object.) [▶ 1306]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues(String, Type, .Int32., TimeSpan, Func.Exception, Object.) [▶ 1307]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, IObservable.Unit.) [▶ 1308]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, TimeSpan) [▶ 1309]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, IObservable.Unit.) [▶ 1294]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, TimeSpan) [▶ 1295]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, IObservable.Unit., Func.Exception, T.) [▶ 1314]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, TimeSpan, Func.Exception, T.) [▶ 1315]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, .Int32., IObservable.Unit.) [▶ 1312]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	PollValues.T.(ISymbol, .Int32., TimeSpan) [▶ 1313]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, IObservable.Unit., Func.Exception, T.) [▶ 1298]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)

	Name	Description
	PollValues.T.(String, TimeSpan, Func.Exception, T.) [▶ 1299]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, .Int32., IObservable.Unit.) [▶ 1296]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	PollValues.T.(String, .Int32., TimeSpan) [▶ 1297]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(ISymbol, .Int32., IObservable.Unit., Func.Exception, T.) [▶ 1318]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	PollValues.T.(ISymbol, .Int32., TimeSpan, Func.Exception, T.) [▶ 1319]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues.T.(String, .Int32., IObservable.Unit., Func.Exception, T.) [▶ 1302]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
 	PollValues.T.(String, .Int32., TimeSpan, Func.Exception, T.) [▶ 1303]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues2.T.(ISymbol, .Int32., IObservable.Unit.) [▶ 1330]	Overloaded. Polls the symbol values on timepoints where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	PollValues2.T.(String, .Int32., IObservable.Unit.) [▶ 1329]	Overloaded. Polls the symbol values on time points where the polling observable streams data / triggers (Defined by AnyTypeExtensions [▶ 1281].)
	ReadSysServState [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (Defined by SystemServiceExtension [▶ 3281].)
	ReadSysServStateAsync [▶ 3284]	Reads the System Service state (AdsSysServState [▶ 3275]) (asynchronous) (Defined by SystemServiceExtension [▶ 3281].)
	ReadWithFallback(UInt32, UInt32, Memory.Byte., UInt32, Boolean.) [▶ 1219]	Overloaded. Ads Read with fallback. (Defined by AdsClientExtensions [▶ 1215].)

	Name	Description
	ReadWithFallback(UInt32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., Boolean.) [▶ 1220]	Overloaded. Ads Read with Fallback. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallbackAsync(UInt32, UInt32, UInt32, Memory.Byte., CancellationToken) [▶ 1221]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	ReadWithFallbackAsync(UInt32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., CancellationToken) [▶ 1222]	Overloaded. Ads Read with fallback as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedRead(UInt32, UInt32, Memory.Byte., Int32, TimeSpan) [▶ 1224]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedRead(UInt32, UInt32, Memory.Byte., Int32, TimeSpan, Func.ResultRead, Boolean.) [▶ 1225]	Overloaded. Repeated AdsRead. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedReadAsync(UInt32, UInt32, Memory.Byte., Int32, TimeSpan, CancellationToken) [▶ 1226]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)
	RepeatedReadAsync(UInt32, UInt32, Memory.Byte., Int32, TimeSpan, Func.ResultRead, Boolean., CancellationToken) [▶ 1227]	Overloaded. Repeated AdsRead as an asynchronous operation. (Defined by AdsClientExtensions [▶ 1215].)

	Name	Description
	RestartTwinCATAsync [▶ 3285]	Sends a SystemService Restart to the target system and waits until the system has finished the state changes (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [▶ 3281].)
	SetAdsState [▶ 3299]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [▶ 3292].)
	SetAdsStateAsync [▶ 3300]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes. (Defined by ConnectionStateExtension [▶ 3292].)
	WaitUntilRestarted [▶ 3286]	Waits until the Restart is detected on the client (SystemService, Port 10000) (Defined by SystemServiceExtension [▶ 3281].)
	WaitUntilRestartedAsync [▶ 3287]	Waits until the SystemService Restart is detected on the client (SystemService, Port 10000) (asynchronously) (Defined by SystemServiceExtension [▶ 3281].)
 	WhenAdsStateChanges [▶ 1261]	Gets an observable sequence of AdsState [▶ 729]s. (Defined by AdsClientExtensions [▶ 1249].)
	WhenNotification(ISymbol) [▶ 1263]	Overloaded. Gets an observable sequence of Notification [▶ 1100]s. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenNotification(ISymbolCollection) [▶ 1264]	Overloaded. Gets an observable sequence of Notification [▶ 1100] objects. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenNotification(IList.ISymbol, NotificationSettings) [▶ 1268]	Overloaded. Gets an observable sequence of Notification [▶ 1100] objects. (Defined by AdsClientExtensions [▶ 1249].)
	WhenNotification(ISymbol, NotificationSettings) [▶ 1265]	Overloaded. Gets an observable sequence of SymbolValueNotification [▶ 1331]s. (Defined by AdsClientExtensions [▶ 1249].)
 	WhenNotification(String, Type, NotificationSettings) [▶ 1324]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [▶ 1281].)
 	WhenNotification(String, NotificationSettings) [▶ 1323]	Overloaded. Creates an observable sequence of values that are created by ADS Notifications. (Defined by AnyTypeExtensions [▶ 1281].)
	WhenSymbolVersionChanges [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
	WhenSymbolVersionChanges(IScheduler) [▶ 1270]	Overloaded. Gets an observable sequence of SymbolVersion changed counts. (Defined by AdsClientExtensions [▶ 1249].)
	WhenValueChanged [▶ 1345]	Observable sequence of Values driven by ADS Notifications on the specified symbol. (Defined by ValueSymbolExtensions [▶ 1333].)


	Name	Description
		
 	WriteValues.T. (String, IObservable.T.) [▶ 1326]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)
 	WriteValues.T. (String, IObservable.T., Action.Exception.) [▶ 1327]	Overloaded. Writes the sequence of values to the symbol specified by the instance path. (Defined by AnyTypeExtensions [▶ 1281].)

Reference

[IAdsDisposableConnection Interface](#) [▶ 915]

[TwinCAT.Ads Namespace](#) [▶ 179]













Also see about this

-  [AdsClientExtensions.WhenNotification Method \(IAdsConnection, ISymbolCollection, NotificationSettings\)](#) [▶ 1266]

6.2.33.3 IAdsDisposableConnection Events

The [IAdsDisposableConnection](#) [▶ 915] type exposes the following members.

Events

	Name	Description
 	AdsNotification [▶ 991]	Occurs when the ADS device sends a notification to the client. (Inherited from IAdsNotifications [▶ 966].)
	AdsNotificationError [▶ 993]	Occurs when a exception has occurred during notification management. (Inherited from IAdsNotifications [▶ 966].)
 	AdsNotificationEx [▶ 993]	Occurs when the ADS devices sends a notification to the client. (Inherited from IAdsNotifications [▶ 966].)
	AdsStateChanged [▶ 1061]	Occurs when the AdsState of the target system has been changed. (Inherited from IAdsStateProvider [▶ 1056].)
 	AdsSumNotification [▶ 994]	Occurs when Notifications are send (bundled notifications) (Inherited from IAdsNotifications [▶ 966].)
	AdsSymbolVersionC hanged [▶ 1065]	Occurs when the ADS Symbol Version changed. (Inherited from IAdsSymbolChangedProvider [▶ 1061].)
 	ConnectionStateCha nged [▶ 91]	Occurs when connection status of the IConnectionStateProvider [▶ 89] has been changed. (Inherited from IConnectionStateProvider [▶ 89].)
	RouterStateChange d [▶ 1100]	(Local) Router state changed event. (Inherited from IRouterNotificationProvider [▶ 1099].)

Reference

[IAdsDisposableConnection Interface \[► 915\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.34 IAdsHandle Interface

Interface for ADS access via variable handle

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229














Syntax



C#

```
public interface IAdsHandle
```

The IAdsHandle type exposes the following members.

Methods

	Name	Description
	CreateVariableHandle [► 956]	Determines the Symbol handle by its instance path synchronously.
	CreateVariableHandleAsync [► 956]	Determines the Symbol handle by its instance path asynchronously.
	DeleteVariableHandle [► 957]	Releases the specified symbol/variable handle synchronously.
	DeleteVariableHandleAsync [► 958]	Releases the specified symbol/variable handle asynchronously.
	Read [► 959]	Reads data synchronously from an ADS device and writes to the specified readBuffer.
	ReadAsync [► 959]	Reads the value data of the symbol asynchronously into the readBuffer.
	ReadWrite [► 960]	Writes data synchronously to an ADS device and then Reads data from that target.
	ReadWriteAsync [► 961]	ReadWrites value data asynchronously to/from the symbol represented by the variableHandle.
	TryCreateVariableHandle [► 961]	Determines the Symbol handle by its instance path synchronously.
	TryDeleteVariableHandle [► 962]	Releases the specified symbol/variable handle synchronously.
	TryRead [► 963]	Reads the value synchronously data of the symbol, that is represented by the variable handle into the readBuffer.
	TryReadWrite [► 963]	ReadWrites value data synchronously to/from the symbol represented by the variableHandle.
	TryWrite [► 964]	Writes the value data synchronously that is represented in the writeBuffer to the symbol with the specified variableHandle.

	Name	Description
	Write [▶ 964]	Writes data synchronously to an ADS device.
	WriteAsync [▶ 965]	Writes the value data asynchronously that is represented by the writeBuffer to the symbol specified by the variableHandle.
















Reference

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.34.1 IAdsHandle Methods

The [IAdsHandle \[▶ 954\]](#) type exposes the following members.

Methods

	Name	Description
	CreateVariableHandle [▶ 956]	Determines the Symbol handle by its instance path synchronously.
	CreateVariableHandleAsync [▶ 956]	Determines the Symbol handle by its instance path asynchronously.
	DeleteVariableHandle [▶ 957]	Releases the specified symbol/variable handle synchronously.
	DeleteVariableHandleAsync [▶ 958]	Releases the specified symbol/variable handle asynchronously.
	Read [▶ 959]	Reads data synchronously from an ADS device and writes to the specified readBuffer.
	ReadAsync [▶ 959]	Reads the value data of the symbol asynchronously into the readBuffer.
	ReadWrite [▶ 960]	Writes data synchronously to an ADS device and then Reads data from that target.
	ReadWriteAsync [▶ 961]	ReadWrites value data asynchronously to/from the symbol represented by the variableHandle.
	TryCreateVariableHandle [▶ 961]	Determines the Symbol handle by its instance path synchronously.
	TryDeleteVariableHandle [▶ 962]	Releases the specified symbol/variable handle synchronously.
	TryRead [▶ 963]	Reads the value synchronously data of the symbol, that is represented by the variable handle into the readBuffer.
	TryReadWrite [▶ 963]	ReadWrites value data synchronously to/from the symbol represented by the variableHandle.
	TryWrite [▶ 964]	Writes the value data synchronously that is represented in the writeBuffer to the symbol with the specified variableHandle.
	Write [▶ 964]	Writes data synchronously to an ADS device.
	WriteAsync [▶ 965]	Writes the value data asynchronously that is represented by the writeBuffer to the symbol specified by the variableHandle.

Reference

[IAdsHandle Interface](#) [[▶ 954](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.34.1.1 IAdsHandle.CreateVariableHandle Method

Determines the Symbol handle by its instance path synchronously.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
uint CreateVariableHandle(  
    string symbolPath  
)
```

Parameters

symbolPath	Type: System.String SymbolName / InstancePath.
------------	---

Return Value

Type: [UInt32](#)

The symbols/variable handle

Remarks

It is a good practice to release all variable handles after use to regain internal resources in the TwinCAT subsystem. The composite method to this [CreateVariableHandle\(String\)](#) is the [DeleteVariableHandle\(UInt32\)](#) [[▶ 957](#)]

Reference

[IAdsHandle Interface](#) [[▶ 954](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsHandle.DeleteVariableHandle\(UInt32\)](#) [[▶ 957](#)]

[IAdsHandle.CreateVariableHandleAsync\(String, CancellationToken\)](#) [[▶ 956](#)]

[IAdsHandle.TryCreateVariableHandle\(String, UInt32.\)](#) [[▶ 961](#)]

6.2.34.1.2 IAdsHandle.CreateVariableHandleAsync Method

Determines the Symbol handle by its instance path asynchronously.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultHandle> CreateVariableHandleAsync(
    string symbolPath,
    CancellationToken cancel
)
```

Parameters

symbolPath	Type: System.String SymbolName / InstancePath.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultHandle](#) [[▶ 1136](#)].

A task that represents the asynchronous 'CreateVariableHandle' operation. The [ResultHandle](#) [[▶ 1136](#)] parameter contains the variable handle ([Handle](#) [[▶ 1138](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Remarks

It is a good practice to release all variable handles after use to regain internal resources in the TwinCAT subsystem. The composite method to this [CreateVariableHandleAsync\(String, CancellationToken\)](#) is the [DeleteVariableHandleAsync\(UInt32, CancellationToken\)](#) [[▶ 958](#)]

Reference

[IAdsHandle Interface](#) [[▶ 954](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsHandle.DeleteVariableHandleAsync\(UInt32, CancellationToken\)](#) [[▶ 958](#)]

[IAdsHandle.TryCreateVariableHandle\(String, UInt32.\)](#) [[▶ 961](#)]

[IAdsHandle.CreateVariableHandle\(String\)](#) [[▶ 956](#)]

6.2.34.1.3 IAdsHandle.DeleteVariableHandle Method

Releases the specified symbol/variable handle synchronously.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
void DeleteVariableHandle(
    uint variableHandle
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable
----------------	---

Return Value

Type:
The ADS error code.

Remarks

It is a good practice to release all variable handles after use to regain internal resources in the TwinCAT subsystem. The composite method to this [TryDeleteVariableHandle\(UInt32\)](#) [► 962] is the [TryCreateVariableHandle\(String, UInt32.\)](#) [► 961]

Reference

[IAdsHandle Interface](#) [► 954]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsHandle.CreateVariableHandle\(String\)](#) [► 956]

[IAdsHandle.TryDeleteVariableHandle\(UInt32\)](#) [► 962]

[IAdsHandle.DeleteVariableHandleAsync\(UInt32, CancellationToken\)](#) [► 958]

6.2.34.1.4 IAdsHandle.DeleteVariableHandleAsync Method

Releases the specified symbol/variable handle asynchronously.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
Task<ResultAds> DeleteVariableHandleAsync (
    uint variableHandle,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds](#) [► 1116].

A task that represents the asynchronous 'DeleteVariableHandle' operation. The [ResultAds](#) [► 1116] parameter contains the [ErrorCode](#) [► 1120] after execution.

Remarks

It is a good practice to release all variable handles after use to regain internal resources in the TwinCAT subsystem. The composite method to this [DeleteVariableHandleAsync\(UInt32, CancellationToken\)](#) is the [CreateVariableHandleAsync\(String, CancellationToken\)](#) [► 956]

Reference

[IAdsHandle Interface](#) [► 954]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsHandle.CreateVariableHandleAsync\(String, CancellationToken\)](#) [► 956]

[IAdsHandle.TryDeleteVariableHandle\(UInt32\)](#) [► 962]

[IAdsHandle.DeleteVariableHandle\(UInt32\)](#) [► 957]

6.2.34.1.5 IAdsHandle.Read Method

Reads data synchronously from an ADS device and writes to the specified readBuffer.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Read(
    uint variableHandle,
    Memory<byte> readBuffer
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable
readBuffer	Type: System.Memory.Byte. The read buffer / data

Return Value

Type: [Int32](#)

Number of successfully returned data bytes.

Reference

[IAdsHandle Interface](#) [► 954]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsHandle.TryRead\(UInt32, Memory.Byte., Int32.\)](#) [► 963]

[IAdsHandle.ReadAsync\(UInt32, Memory.Byte., CancellationToken\)](#) [► 959]

6.2.34.1.6 IAdsHandle.ReadAsync Method

Reads the value data of the symbol asynchronously into the readBuffer.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultRead> ReadAsync(
    uint variableHandle,
    Memory<byte> readBuffer,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
readBuffer	Type: System.Memory.Byte . The read buffer/data.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultRead](#) [[▶ 1143](#)].

A task that represents the asynchronous read operation. The [ResultRead](#) [[▶ 1143](#)] parameter contains the total number of bytes read into the buffer ([ReadBytes](#) [[▶ 1145](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution..

Reference

[IAdsHandle Interface](#) [[▶ 954](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.34.1.7 IAdsHandle.ReadWrite Method

Writes data synchronously to an ADS device and then Reads data from that target.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
int ReadWrite(
    uint variableHandle,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

variableHandle	Type: System.UInt32 Variable handle.
readBuffer	Type: System.Memory.Byte . The read buffer.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.

Return Value

Type: [Int32](#)

Number of successfully returned data bytes.

Reference

[IAdsHandle Interface](#) [[▶ 954](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsHandle.TryReadWrite\(UInt32, Memory.Byte., ReadOnlyMemory.Byte., Int32.\)](#) [[▶ 963](#)]

[IAdsHandle.ReadWriteAsync\(UInt32, Memory.Byte., ReadOnlyMemory.Byte., CancellationToken\)](#) [[▶ 961](#)]

6.2.34.1.8 IAdsHandle.ReadWriteAsync Method

ReadWrites value data asynchronously to/from the symbol represented by the variableHandle.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultReadWrite> ReadWriteAsync (
    uint variableHandle,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 Variable handle.
readBuffer	Type: System.Memory.Byte . The read data / value
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write data / value.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultReadWrite](#) [[▶ 1163](#)].

A task that represents the asynchronous 'ReadWrite' operation. The [ResultReadWrite](#) [[▶ 1163](#)] parameter contains the total number of bytes read into the buffer ([ReadBytes](#) [[▶ 1145](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Reference

[IAdsHandle Interface](#) [[▶ 954](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.34.1.9 IAdsHandle.TryCreateVariableHandle Method

Determines the Symbol handle by its instance path synchronously.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryCreateVariableHandle (
    string symbolPath,
    out uint variableHandle
)
```

Parameters

symbolPath	Type: System.String SymbolName / InstancePath.
------------	---

variableHandle	Type: System.UInt32 . The symbols handle.
----------------	--

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS error code.

Remarks

It is a good practice to release all variable handles after use to regain internal resources in the TwinCAT subsystem. The composite method to this `TryCreateVariableHandle(String, UInt32.)` is the `TryDeleteVariableHandle(UInt32)` [[▶ 962](#)]

Reference

[IAdsHandle Interface](#) [[▶ 954](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsHandle.TryDeleteVariableHandle\(UInt32\)](#) [[▶ 962](#)]

[IAdsHandle.CreateVariableHandleAsync\(String, CancellationToken\)](#) [[▶ 956](#)]

[IAdsHandle.CreateVariableHandle\(String\)](#) [[▶ 956](#)]

6.2.34.1.10 IAdsHandle.TryDeleteVariableHandle Method

Releases the specified symbol/variable handle synchronously.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax**C#**

```
AdsErrorCode TryDeleteVariableHandle(
    uint variableHandle
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable
----------------	---

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS error code.

Remarks

It is a good practice to release all variable handles after use to regain internal resources in the TwinCAT subsystem. The composite method to this `TryDeleteVariableHandle(UInt32)` is the `TryCreateVariableHandle(String, UInt32.)` [[▶ 961](#)]

Reference

[IAdsHandle Interface](#) [[▶ 954](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsHandle.TryCreateVariableHandle\(String, UInt32.\)](#) [[▶ 961](#)]

[IAdsHandle.DeleteVariableHandleAsync\(UInt32, CancellationToken\)](#) [[▶ 958](#)]

[IAdsHandle.DeleteVariableHandle\(UInt32\)](#) [[▶ 957](#)]

6.2.34.1.11 IAdsHandle.TryRead Method

Reads the value synchronously data of the symbol, that is represented by the variable handle into the readBuffer.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryRead(
    uint variableHandle,
    Memory<byte> readBuffer,
    out int readBytes
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
readBuffer	Type: System.Memory.Byte . The read buffer/data
readBytes	Type: System.Int32 . Number of read bytes.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The ADS error code.

Reference

[IAdsHandle Interface](#) [[▶ 954](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.34.1.12 IAdsHandle.TryReadWrite Method

ReadWrites value data synchronously to/from the symbol represented by the variableHandle.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryReadWrite(
    uint variableHandle,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer,
    out int readBytes
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
readBuffer	Type: System.Memory.Byte . The read buffer / read data.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer / write data.
readBytes	Type: System.Int32 . Number of read bytes.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS error code.

Reference

[IAdsHandle Interface](#) [[▶ 954](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.34.1.13 IAdsHandle.TryWrite Method

Writes the value data synchronously that is represented in the writeBuffer to the symbol with the specified variableHandle.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
AdsErrorCode TryWrite(
    uint variableHandle,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer / value.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS error code.

Reference

[IAdsHandle Interface](#) [[▶ 954](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.34.1.14 IAdsHandle.Write Method

Writes data synchronously to an ADS device.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void Write(
    uint variableHandle,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

variableHandle	Type: System.UInt32 Handle of the ADS variable
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer / value to be written

Reference

[IAdsHandle Interface](#) [► 954]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsHandle.TryWrite\(UInt32, ReadOnlyMemory.Byte.\)](#) [► 964]

[IAdsHandle.WriteAsync\(UInt32, ReadOnlyMemory.Byte., CancellationToken\)](#) [► 965]

6.2.34.1.15 IAdsHandle.WriteAsync Method

Writes the value data asynchronously that is represented by the writeBuffer to the symbol specified by the variableHandle.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultWrite> WriteAsync(
    uint variableHandle,
    ReadOnlyMemory<byte> writeBuffer,
    CancellationToken cancel
)
```

Parameters

variableHandle	Type: System.UInt32 The variable handle.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer/value.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite](#) [► 1187].

A task that represents the asynchronous write operation. The [ResultWrite](#) [► 1187] parameter contains the [ErrorCode](#) [► 1120] after execution.

Reference

[IAdsHandle Interface](#) [► 954]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.35 IAdsNotifications Interface

Interface for Notification management.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229






Syntax










C#




```
public interface IAdsNotifications
```

The IAdsNotifications type exposes the following members.








Methods

	Name	Description
	AddDeviceNotification(String, Int32, NotificationSettings, Object) [► 971]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [► 991] event.
	AddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object) [► 972]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event.
	AddDeviceNotificationAsync(String, Int32, NotificationSettings, Object, CancellationToken) [► 974]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [► 991] event.
	AddDeviceNotificationAsync(UInt32, UInt32, Int32, NotificationSettings, Object, CancellationToken) [► 975]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [► 991] event.
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type) [► 977]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [► 993] event.

	Name	Description
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32.) [▶ 977]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event.
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type) [▶ 978]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event.
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32.) [▶ 980]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive ('AnyType') types are allowed for the parameter type.
	AddDeviceNotificationExAsync(String, NotificationSettings, Object, Type, .Int32., CancellationTokens) [▶ 981]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 993] event.
	AddDeviceNotificationExAsync(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., CancellationTokens) [▶ 982]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 993] event.
	DeleteDeviceNotification [▶ 983]	Deletes a registered notification.
	DeleteDeviceNotificationAsync [▶ 984]	Deletes a registered notification asynchronously.
	TryAddDeviceNotification(String, Int32, NotificationSettings, Object, UInt32.) [▶ 985]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event.
	TryAddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.) [▶ 986]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event.

	Name	Description
	TryAddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32., UInt32.) [▶ 988]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event.
	TryAddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., UInt32.) [▶ 989]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event.
	TryDeleteDeviceNotification [▶ 990]	Deletes a registered notification.

Events

	Name	Description
 	AdsNotification [▶ 991]	Occurs when the ADS device sends a notification to the client.
	AdsNotificationError [▶ 993]	Occurs when a exception has occurred during notification management.
 	AdsNotificationEx [▶ 993]	Occurs when the ADS devices sends a notification to the client.
 	AdsSumNotification [▶ 994]	Occurs when Notifications are send (bundled notifications)



Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.35.1 IAdsNotifications Methods

The [IAdsNotifications](#) [[▶ 966](#)] type exposes the following members.

Methods

	Name	Description
	AddDeviceNotification(String, Int32, NotificationSettings, Object) [▶ 971]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event.
	AddDeviceNotification(UInt32, UInt32,	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event.

	Name	Description
	Int32, NotificationSettings, Object) [▶ 972]	
	AddDeviceNotificationAsync(String, Int32, NotificationSettings, Object, CancellationToken) [▶ 974]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 991] event.
	AddDeviceNotificationAsync(UInt32, UInt32, Int32, NotificationSettings, Object, CancellationToken) [▶ 975]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 991] event.
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type) [▶ 977]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event.
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32.) [▶ 977]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event.
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type) [▶ 978]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event.
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32.) [▶ 980]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive ('AnyType') types are allowed for the parameter type.
	AddDeviceNotificationExAsync(String, NotificationSettings, Object, Type, .Int32., CancellationToken) [▶ 981]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▶ 993] event.

	Name	Description
	AddDeviceNotificationExAsync(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., CancellationToken) [▸ 982]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▸ 993] event.
	DeleteDeviceNotification [▸ 983]	Deletes a registered notification.
	DeleteDeviceNotificationAsync [▸ 984]	Deletes a registered notification asynchronously.
	TryAddDeviceNotification(String, Int32, NotificationSettings, Object, UInt32.) [▸ 985]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▸ 991] event.
	TryAddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.) [▸ 986]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▸ 991] event.
	TryAddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32., UInt32.) [▸ 988]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▸ 993] event.
	TryAddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., UInt32.) [▸ 989]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▸ 993] event.
	TryDeleteDeviceNotification [▸ 990]	Deletes a registered notification.



Reference

[IAdsNotifications Interface](#) [[▸ 966](#)]

[TwinCAT.Ads Namespace](#) [[▸ 179](#)]

6.2.35.1.1 IAdsNotifications.AddDeviceNotification Method

Overload List

	Name	Description
	AddDeviceNotification(String, Int32, NotificationSettings, Object) [► 971]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [► 991] event.
	AddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object) [► 972]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event.

Reference

[IAdsNotifications Interface \[► 966\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.35.1.1.1 IAdsNotifications.AddDeviceNotification Method (String, Int32, NotificationSettings, Object)

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotification \[► 991\]](#) event.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
uint AddDeviceNotification(
    string symbolPath,
    int dataSize,
    NotificationSettings settings,
    Object? userData
)
```

Parameters

symbolPath	Type: System.String Symbol / Instance path of the ADS variable.
dataSize	Type: System.Int32 Maximum amount of data in bytes to receive with this ADS Notification
settings	Type: TwinCAT.Ads.NotificationSettings [► 1106] The settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)

Return Value

Type: [UInt32](#)
The notification handle.

Remarks

The
dataSize

Parameter defines the amount of bytes, that will be attached to the [AdsNotification](#) [[▶ 991](#)] as value. Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\)](#) [[▶ 983](#)] should always called when the notification is not used anymore.

Reference

[IAdsNotifications Interface](#) [[▶ 966](#)]

[AddDeviceNotification Overload](#) [[▶ 971](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsNotifications.AdsNotification](#) [[▶ 991](#)]

[IAdsNotifications.DeleteDeviceNotification\(UInt32\)](#) [[▶ 983](#)]

[AddDeviceNotification Overload](#) [[▶ 971](#)]

[AddDeviceNotificationAsync Overload](#) [[▶ 973](#)]

[TryAddDeviceNotification Overload](#) [[▶ 985](#)]

6.2.35.1.1.2 IAdsNotifications.AddDeviceNotification Method (UInt32, UInt32, Int32, NotificationSettings, Object)

Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
uint AddDeviceNotification(
    uint indexGroup,
    uint indexOffset,
    int dataSize,
    NotificationSettings settings,
    Object? userData
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
dataSize	Type: System.Int32 Maximum amount of data in bytes to receive with this ADS Notification
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)

Return Value

Type: [UInt32](#)
 The notification handle.

Remarks

The
 dataSize



Parameter defines the amount of bytes, that will be attached to the [AdsNotification](#) [[▶ 991](#)] as value. Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\)](#) [[▶ 983](#)] should always called when the notification is not used anymore.

Reference

- [IAdsNotifications Interface](#) [[▶ 966](#)]
- [AddDeviceNotification Overload](#) [[▶ 971](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]
- [IAdsNotifications.DeleteDeviceNotification\(UInt32\)](#) [[▶ 983](#)]
- [IAdsNotifications.AdsNotification](#) [[▶ 991](#)]
- [IAdsNotifications.AdsNotificationError](#) [[▶ 993](#)]
- [AddDeviceNotification Overload](#) [[▶ 971](#)]
- [TryAddDeviceNotification Overload](#) [[▶ 985](#)]
- [AddDeviceNotificationAsync Overload](#) [[▶ 973](#)]

6.2.35.1.2 IAdsNotifications.AddDeviceNotificationAsync Method

Overload List

	Name	Description
	AddDeviceNotificationAsync(String, Int32, NotificationSettings, Object, CancellationToken) [▶ 974]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 991] event.
	AddDeviceNotificationAsync(UInt32, UInt32, Int32, NotificationSettings, Object, CancellationToken) [▶ 975]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotification [▶ 991] event.

Reference

- [IAdsNotifications Interface](#) [[▶ 966](#)]

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.35.1.2.1 IAdsNotifications.AddDeviceNotificationAsync Method (String, Int32, NotificationSettings, Object, CancellationToken)

Connects a variable to the ADS client asynchronously. The ADS client will be notified by the [AdsNotification \[► 991\]](#) event.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultHandle> AddDeviceNotificationAsync(
    string symbolPath,
    int dataSize,
    NotificationSettings settings,
    Object? userData,
    CancellationToken cancel
)
```

Parameters

symbolPath	Type: System.String The symbol/instance path of the ADS variable.
dataSize	Type: System.Int32 Maximum amount of data in bytes to receive with this ADS Notification
settings	Type: TwinCAT.Ads.NotificationSettings [► 1106] The notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
cancel	Type: System.Threading.CancellationToken The Cancellation token.

Return Value

Type: [Task.ResultHandle \[► 1136\]](#).

A task that represents the asynchronous 'AddDeviceNotification' operation. The [ResultHandle \[► 1136\]](#) type parameter contains the created handle ([Handle \[► 1138\]](#)) and the [ErrorCode \[► 1120\]](#) after execution.

Remarks

The

dataSize

Parameter defines the amount of bytes, that will be attached to the [AdsNotification \[► 991\]](#) as value. Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotificationAsync\(UInt32, CancellationToken\) \[► 984\]](#) should always be called when the notification is not used anymore.

Reference

[IAdsNotifications Interface \[► 966\]](#)

[AddDeviceNotificationAsync Overload \[► 973\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

[IAdsNotifications.AdsNotification](#) [[▶ 991](#)]

[IAdsNotifications.DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [[▶ 984](#)]

[AddDeviceNotification Overload](#) [[▶ 971](#)]

[AddDeviceNotificationAsync Overload](#) [[▶ 973](#)]

[TryAddDeviceNotification Overload](#) [[▶ 985](#)]

6.2.35.1.2.2 IAdsNotifications.AddDeviceNotificationAsync Method (UInt32, UInt32, Int32, NotificationSettings, Object, CancellationToken)

Connects a variable to the ADS client asynchronously. The ADS client will be notified by the [AdsNotification](#) [[▶ 991](#)] event.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultHandle> AddDeviceNotificationAsync (
    uint indexGroup,
    uint indexOffset,
    int dataSize,
    NotificationSettings settings,
    Object? userData,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group number of the requested ADS service.
indexOffset	Type: System.UInt32 The index offset number of the requested ADS service.
dataSize	Type: System.Int32 Maximum amount of data in bytes to receive with this ADS Notification
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
cancel	Type: System.Threading.CancellationToken The Cancellation token.

Return Value

Type: [Task.ResultHandle](#) [[▶ 1136](#)].

A task that represents the asynchronous 'AddDeviceNotification' operation. The [ResultHandle](#) [[▶ 1136](#)] type parameter contains the created handle ([Handle](#) [[▶ 1138](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Remarks

The

dataSize

Parameter defines the amount of bytes, that will be attached to the [AdsNotification](#) [► 991] as value. Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [► 984] should always be called when the notification is not used anymore.

Reference

[IAdsNotifications Interface](#) [► 966]

[AddDeviceNotificationAsync Overload](#) [► 973]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsNotifications.AdsNotification](#) [► 991]

[IAdsNotifications.DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [► 984]





[AddDeviceNotification Overload](#) [► 971]

[AddDeviceNotificationAsync Overload](#) [► 973]

[TryAddDeviceNotification Overload](#) [► 985]

6.2.35.1.3 IAdsNotifications.AddDeviceNotificationEx Method

Overload List

	Name	Description
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type) [► 977]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [► 993] event.
	AddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32.) [► 977]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [► 993] event.
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type) [► 978]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [► 993] event.
	AddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32.) [► 980]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive ('AnyType') types are allowed for the parameter type.

Reference

[IAdsNotifications Interface](#) [► 966]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.35.1.3.1 IAdsNotifications.AddDeviceNotificationEx Method (String, NotificationSettings, Object, Type)

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotificationEx \[▶ 993\]](#) event.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
uint AddDeviceNotificationEx(
    string symbolPath,
    NotificationSettings settings,
    Object? userData,
    Type type
)
```

Parameters

symbolPath	Type: System.String Symbol/Instance path of the ADS variable.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
type	Type: System.Type Type of the object stored in the event argument ('AnyType')

Return Value

Type: [UInt32](#)
The notification handle.

Remarks

Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\) \[▶ 983\]](#) should always called when the notification is not used anymore.

Reference

- [IAdsNotifications Interface \[▶ 966\]](#)
- [AddDeviceNotificationEx Overload \[▶ 976\]](#)
- [TwinCAT.Ads Namespace \[▶ 179\]](#)
- [IAdsNotifications.AdsNotificationEx \[▶ 993\]](#)
- [IAdsNotifications.DeleteDeviceNotification\(UInt32\) \[▶ 983\]](#)
- [AddDeviceNotificationEx Overload \[▶ 976\]](#)
- [AddDeviceNotificationExAsync Overload \[▶ 981\]](#)
- [TryAddDeviceNotificationEx Overload \[▶ 988\]](#)

6.2.35.1.3.2 IAdsNotifications.AddDeviceNotificationEx Method (String, NotificationSettings, Object, Type, .Int32.)

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotificationEx \[▶ 993\]](#) event.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
uint AddDeviceNotificationEx(
    string symbolPath,
    NotificationSettings settings,
    Object? userData,
    Type type,
    int[]? args
)
```

Parameters

symbolPath	Type: System.String Symbol/Instance path of the ADS variable.
settings	Type: TwinCAT.Ads.NotificationSettings [► 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
type	Type: System.Type Type of the object stored in the event argument ('AnyType')
args	Type: .System.Int32 . Additional arguments (for 'AnyType')

Return Value

Type: [UInt32](#)
The notification handle.

Remarks

Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\)](#) [► 983] should always called when the notification is not used anymore.

Reference

[IAdsNotifications Interface](#) [► 966]

[AddDeviceNotificationEx Overload](#) [► 976]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsNotifications.AdsNotificationEx](#) [► 993]

[IAdsNotifications.DeleteDeviceNotification\(UInt32\)](#) [► 983]

[AddDeviceNotificationEx Overload](#) [► 976]

[AddDeviceNotificationExAsync Overload](#) [► 981]

[TryAddDeviceNotificationEx Overload](#) [► 988]

6.2.35.1.3.3 IAdsNotifications.AddDeviceNotificationEx Method (UInt32, UInt32, NotificationSettings, Object, Type)

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotificationEx](#) [► 993] event.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
uint AddDeviceNotificationEx(
    uint indexGroup,
    uint indexOffset,
    NotificationSettings settings,
    Object? userData,
    Type type
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
type	Type: System.Type Type of the object stored in the event argument ('AnyType')

Return Value

Type: [UInt32](#)
The notification handle.

Remarks

xxxxxBecause notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\)](#) [▶ 983] should always called when the notification is not used anymore. [AddDeviceNotificationEx Overload](#) [▶ 976]

Reference

- [IAdsNotifications Interface](#) [▶ 966]
- [AddDeviceNotificationEx Overload](#) [▶ 976]
- [TwinCAT.Ads Namespace](#) [▶ 179]
- [IAdsNotifications.DeleteDeviceNotification\(UInt32\)](#) [▶ 983]
- [IAdsNotifications.AdsNotificationEx](#) [▶ 993]
- [IAdsNotifications.AdsNotificationError](#) [▶ 993]
- [AddDeviceNotificationEx Overload](#) [▶ 976]
- [TryAddDeviceNotificationEx Overload](#) [▶ 988]
- [AddDeviceNotificationExAsync Overload](#) [▶ 981]

6.2.35.1.3.4 IAdsNotifications.AddDeviceNotificationEx Method (UInt32, UInt32, NotificationSettings, Object, Type, .Int32.)

Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification event. If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive ('AnyType') types are allowed for the parameter type.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
uint AddDeviceNotificationEx(
    uint indexGroup,
    uint indexOffset,
    NotificationSettings settings,
    Object? userData,
    Type type,
    int[] args
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data.
type	Type: System.Type Type of the object stored in the event argument.
args	Type: .System.Int32 . Additional arguments for 'AnyType' types.

Return Value

Type: [UInt32](#)

The notification handle.

Remarks

Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\)](#) [[▶ 983](#)] should always called when the notification is not used anymore.

Reference

[IAdsNotifications Interface](#) [[▶ 966](#)]

[AddDeviceNotificationEx Overload](#) [[▶ 976](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsNotifications.AdsNotificationEx](#) [[▶ 993](#)]



[AddDeviceNotificationEx Overload](#) [[▶ 976](#)]

[AddDeviceNotificationExAsync Overload](#) [[▶ 981](#)]

[TryAddDeviceNotificationEx Overload \[▸ 988\]](#)

6.2.35.1.4 IAdsNotifications.AddDeviceNotificationExAsync Method

Overload List

	Name	Description
	AddDeviceNotificationExAsync(String, NotificationSettings, Object, Type, .Int32., CancellationTok en) [▸ 981]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▸ 993] event.
	AddDeviceNotificationExAsync(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., CancellationTok en) [▸ 982]	Connects a variable to the ADS client asynchronously. The ADS client will be notified by the AdsNotificationEx [▸ 993] event.

Reference

[IAdsNotifications Interface \[▸ 966\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.35.1.4.1 IAdsNotifications.AddDeviceNotificationExAsync Method (String, NotificationSettings, Object, Type, .Int32., CancellationTok

Connects a variable to the ADS client asynchronously. The ADS client will be notified by the [AdsNotificationEx \[▸ 993\]](#) event.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultHandle> AddDeviceNotificationExAsync (
    string symbolPath,
    NotificationSettings settings,
    Object? userData,
    Type type,
    int[]? args,
    CancellationTok
```

Parameters

symbolPath	Type: System.String The symbol/instance path of the ADS variable.
settings	Type: TwinCAT.Ads.NotificationSettings [▸ 1106] The notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)

type	Type: System.Type Type of the object stored in the event argument ('AnyType')
args	Type: .System.Int32 . Additional arguments (for 'AnyType')
cancel	Type: System.Threading.CancellationTokens The Cancellation token.

Return Value

Type: [Task.ResultHandle](#) [[▶ 1136](#)].

A task that represents the asynchronous 'AddDeviceNotification' operation. The [ResultHandle](#) [[▶ 1136](#)] type parameter contains the created handle ([Handle](#) [[▶ 1138](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Remarks

Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotificationAsync\(UInt32, CancellationTokens\)](#) [[▶ 984](#)] should always be called when the notification is not used anymore.

Reference

[IAdsNotifications Interface](#) [[▶ 966](#)]

[AddDeviceNotificationExAsync Overload](#) [[▶ 981](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsNotifications.AdsNotificationEx](#) [[▶ 993](#)]

[IAdsNotifications.DeleteDeviceNotificationAsync\(UInt32, CancellationTokens\)](#) [[▶ 984](#)]

[AddDeviceNotificationEx Overload](#) [[▶ 976](#)]

[AddDeviceNotificationExAsync Overload](#) [[▶ 981](#)]

[TryAddDeviceNotificationEx Overload](#) [[▶ 988](#)]

6.2.35.1.4.2 IAdsNotifications.AddDeviceNotificationExAsync Method (UInt32, UInt32, NotificationSettings, Object, Type, .Int32., CancellationTokens)

Connects a variable to the ADS client asynchronously. The ADS client will be notified by the [AdsNotificationEx](#) [[▶ 993](#)] event.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultHandle> AddDeviceNotificationExAsync (
    uint indexGroup,
    uint indexOffset,
    NotificationSettings settings,
    Object? userData,
    Type type,
    int[]? args,
    CancellationTokens cancel
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The settings.
userData	Type: System.Object This object can be used to store user specific data.
type	Type: System.Type Type of the object stored in the event argument, only Primitive 'AnyTypes' allowed.
args	Type: .System.Int32 . Additional arguments (for 'AnyType')
cancel	Type: System.Threading.CancellationToken The Cancellation token.

Return Value

Type: [Task.ResultHandle](#) [[▶ 1136](#)].

A task that represents the asynchronous 'AddDeviceNotification' operation. The [ResultHandle](#) [[▶ 1136](#)] type parameter contains the created handle ([Handle](#) [[▶ 1138](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Remarks

If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive types (AnyType) are supported by this method. Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [[▶ 984](#)] should always called when the notification is not used anymore.

Reference

[IAdsNotifications Interface](#) [[▶ 966](#)]

[AddDeviceNotificationExAsync Overload](#) [[▶ 981](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsNotifications.DeleteDeviceNotificationAsync\(UInt32, CancellationToken\)](#) [[▶ 984](#)]

[IAdsNotifications.AdsNotificationEx](#) [[▶ 993](#)]

[IAdsNotifications.AdsNotificationError](#) [[▶ 993](#)]

[AddDeviceNotificationEx Overload](#) [[▶ 976](#)]

[TryAddDeviceNotificationEx Overload](#) [[▶ 988](#)]

[AddDeviceNotificationExAsync Overload](#) [[▶ 981](#)]

6.2.35.1.5 IAdsNotifications.DeleteDeviceNotification Method

Deletes a registered notification.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
void DeleteDeviceNotification(
    uint notificationHandle
)
```

Parameters

notificationHandle	Type: System.UInt32 Notification handle.
--------------------	---

Remarks

This is the complementary method to [AddDeviceNotification Overload \[▶ 971\]](#) overloads and should be called when the notification is not needed anymore the free TwinCAT realtime resources.

Reference

[IAdsNotifications Interface \[▶ 966\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

[AddDeviceNotification Overload \[▶ 971\]](#)

[IAdsNotifications.AdsNotification \[▶ 991\]](#)

[TryAddDeviceNotification Overload \[▶ 985\]](#)

[AddDeviceNotificationAsync Overload \[▶ 973\]](#)

6.2.35.1.6 IAdsNotifications.DeleteDeviceNotificationAsync Method

Deletes a registered notification asynchronously.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultAds> DeleteDeviceNotificationAsync(
    uint notificationHandle,
    CancellationToken cancel
)
```

Parameters

notificationHandle	Type: System.UInt32 Notification handle.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds \[▶ 1116\]](#).

A task that represents the asynchronous 'DeleteDeviceNotification' operation. The [ErrorCode \[▶ 1120\]](#) property contains the ADS error code after execution.

Remarks

This is the complementary method to [AddDeviceNotificationAsync Overload \[▶ 973\]](#) overloads and should be called when the notification is not needed anymore the free TwinCAT realtime resources.

Reference

[IAdsNotifications Interface \[▶ 966\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

[AddDeviceNotificationAsync Overload \[▶ 973\]](#)



[IAdsNotifications.AdsNotification \[▶ 991\]](#)

[TryAddDeviceNotification Overload \[▶ 985\]](#)

[AddDeviceNotification Overload \[▶ 971\]](#)

6.2.35.1.7 IAdsNotifications.TryAddDeviceNotification Method

Overload List

	Name	Description
	TryAddDeviceNotification(String, Int32, NotificationSettings, Object, UInt32.) [▶ 985]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event.
	TryAddDeviceNotification(UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.) [▶ 986]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotification [▶ 991] event.

Reference

[IAdsNotifications Interface \[▶ 966\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.35.1.7.1 IAdsNotifications.TryAddDeviceNotification Method (String, Int32, NotificationSettings, Object, UInt32.)

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotification \[▶ 991\]](#) event.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```

AdsErrorCode TryAddDeviceNotification(
    string symbolPath,
    int dataSize,
    NotificationSettings settings,

```

```

Object? userData,
out uint handle
)

```

Parameters

symbolPath	Type: System.String The symbol/instance path of the ADS variable.
dataSize	Type: System.Int32 Maximum amount of data in bytes to receive with this ADS Notification
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The notification settings.
userData	Type: System.Object This object can be used to store user specific data.
handle	Type: System.UInt32 . The notification handle.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS ErrorCode.

Remarks

The

dataSize

Parameter defines the amount of bytes, that will be attached to the [AdsNotification](#) [[▶ 991](#)] as value. Because notifications allocate TwinCAT system resources, a complementary call to [TryDeleteDeviceNotification\(UInt32\)](#) [[▶ 990](#)] should always be called when the notification is not used anymore.

Reference

[IAdsNotifications Interface](#) [[▶ 966](#)]

[TryAddDeviceNotification Overload](#) [[▶ 985](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsNotifications.AdsNotification](#) [[▶ 991](#)]

[IAdsNotifications.TryDeleteDeviceNotification\(UInt32\)](#) [[▶ 990](#)]

[AddDeviceNotification Overload](#) [[▶ 971](#)]

[AddDeviceNotificationAsync Overload](#) [[▶ 973](#)]

[TryAddDeviceNotification Overload](#) [[▶ 985](#)]

6.2.35.1.7.2 IAdsNotifications.TryAddDeviceNotification Method (UInt32, UInt32, Int32, NotificationSettings, Object, UInt32.)

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotification](#) [[▶ 991](#)] event.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```

AdsErrorCode TryAddDeviceNotification(
    uint indexGroup,
    uint indexOffset,
    int dataSize,
    NotificationSettings settings,
    Object? userData,
    out uint handle
)

```

Parameters

indexGroup	Type: System.UInt32 The index group number of the requested ADS service.
indexOffset	Type: System.UInt32 The index offset number of the requested ADS service.
dataSize	Type: System.Int32 Maximum amount of data in bytes to receive with this ADS Notification.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
handle	Type: System.UInt32 . The notification handle.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS error code.

Remarks

The

dataSize

Parameter defines the amount of bytes, that will be attached to the [AdsNotification](#) [[▶ 991](#)] as value. Because notifications allocate TwinCAT system resources, a complementary call to [TryDeleteDeviceNotification\(UInt32\)](#) [[▶ 990](#)] should always called when the notification is not used anymore.

Reference

[IAdsNotifications Interface](#) [[▶ 966](#)]

[TryAddDeviceNotification Overload](#) [[▶ 985](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsNotifications.TryDeleteDeviceNotification\(UInt32\)](#) [[▶ 990](#)]

[IAdsNotifications.AdsNotification](#) [[▶ 991](#)]

[IAdsNotifications.AdsNotificationError](#) [[▶ 993](#)]



[AddDeviceNotification Overload](#) [[▶ 971](#)]

[TryAddDeviceNotification Overload](#) [[▶ 985](#)]

[AddDeviceNotificationAsync Overload](#) [[▶ 973](#)]

6.2.35.1.8 IAdsNotifications.TryAddDeviceNotificationEx Method

Overload List

	Name	Description
	TryAddDeviceNotificationEx(String, NotificationSettings, Object, Type, .Int32., UInt32.) [▶ 988]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event.
	TryAddDeviceNotificationEx(UInt32, UInt32, NotificationSettings, Object, Type, .Int32., UInt32.) [▶ 989]	Connects a variable to the ADS client. The ADS client will be notified by the AdsNotificationEx [▶ 993] event.

Reference

[IAdsNotifications Interface](#) [[▶ 966](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.35.1.8.1 IAdsNotifications.TryAddDeviceNotificationEx Method (String, NotificationSettings, Object, Type, .Int32., UInt32.)

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotificationEx](#) [[▶ 993](#)] event.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryAddDeviceNotificationEx(
    string symbolPath,
    NotificationSettings settings,
    Object? userData,
    Type type,
    int[]? args,
    out uint handle
)
```

Parameters

symbolPath	Type: System.String Symbol/Instance path of the ADS variable.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
userData	Type: System.Object This object can be used to store user specific data (tag data)
type	Type: System.Type Type of the object stored in the event argument ('AnyType')
args	Type: .System.Int32 . Additional arguments (for 'AnyType')

handle	Type: System.UInt32 . The notification handle
--------	--

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS error code.

Remarks

Because notifications allocate TwinCAT system resources, a complementary call to [TryDeleteDeviceNotification\(UInt32\)](#) [[▶ 990](#)] should always called when the notification is not used anymore.

Reference

- [IAdsNotifications Interface](#) [[▶ 966](#)]
- [TryAddDeviceNotificationEx Overload](#) [[▶ 988](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]
- [IAdsNotifications.AdsNotificationEx](#) [[▶ 993](#)]
- [IAdsNotifications.DeleteDeviceNotification\(UInt32\)](#) [[▶ 983](#)]
- [AddDeviceNotificationEx Overload](#) [[▶ 976](#)]
- [AddDeviceNotificationExAsync Overload](#) [[▶ 981](#)]
- [TryAddDeviceNotificationEx Overload](#) [[▶ 988](#)]

6.2.35.1.8.2 IAdsNotifications.TryAddDeviceNotificationEx Method (UInt32, UInt32, NotificationSettings, Object, Type, .Int32., UInt32.)

Connects a variable to the ADS client. The ADS client will be notified by the [AdsNotificationEx](#) [[▶ 993](#)] event.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryAddDeviceNotificationEx(
    uint indexGroup,
    uint indexOffset,
    NotificationSettings settings,
    Object? userData,
    Type type,
    int[]? args,
    out uint handle
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.

userData	Type: System.Object This object can be used to store user specific data (tag data)
type	Type: System.Type Type of the object stored in the event argument ('AnyType')
args	Type: .System.Int32 . The 'AnyType' arguments.
handle	Type: System.UInt32 . The notification handle.

Return Value

Type: [AdsErrorCode](#) [▶ [664](#)]
The ADS Error code.

Remarks

If type is a string type, the first element of the parameter args specifies the number of characters of the string. If type is an array type, the number of elements for each dimension has to be specified in the parameter args. Only primitive types (AnyType) are supported by this method. Because notifications allocate TwinCAT system resources, a complementary call to [DeleteDeviceNotification\(UInt32\)](#) [▶ [983](#)] should always be called when the notification is not used anymore.

Reference

[IAdsNotifications Interface](#) [▶ [966](#)]

[TryAddDeviceNotificationEx Overload](#) [▶ [988](#)]

[TwinCAT.Ads Namespace](#) [▶ [179](#)]

[IAdsNotifications.DeleteDeviceNotification\(UInt32\)](#) [▶ [983](#)]

[IAdsNotifications.AdsNotificationEx](#) [▶ [993](#)]

[IAdsNotifications.AdsNotificationError](#) [▶ [993](#)]

[AddDeviceNotificationEx Overload](#) [▶ [976](#)]

[TryAddDeviceNotificationEx Overload](#) [▶ [988](#)]

[AddDeviceNotificationExAsync Overload](#) [▶ [981](#)]

6.2.35.1.9 IAdsNotifications.TryDeleteDeviceNotification Method

Deletes a registered notification.

Namespace: [TwinCAT.Ads](#) [▶ [179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryDeleteDeviceNotification(
    uint notificationHandle
)
```

Parameters

notificationHandle	Type: System.UInt32 Notification handle.
--------------------	---

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
 The ADS error code.

Remarks

This is the complementary method to [TryAddDeviceNotification Overload](#) [[▶ 985](#)] overloads and should be called when the notification is not needed anymore the free TwinCAT realtime resources.








Reference

- [IAdsNotifications Interface](#) [[▶ 966](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]
- [AddDeviceNotification Overload](#) [[▶ 971](#)]
- [IAdsNotifications.AdsNotification](#) [[▶ 991](#)]
- [TryAddDeviceNotification Overload](#) [[▶ 985](#)]
- [AddDeviceNotificationAsync Overload](#) [[▶ 973](#)]

6.2.35.2 IAdsNotifications Events

The [IAdsNotifications](#) [[▶ 966](#)] type exposes the following members.

Events

	Name	Description
 	AdsNotification [▶ 991]	Occurs when the ADS device sends a notification to the client.
	AdsNotificationError [▶ 993]	Occurs when a exception has occurred during notification management.
 	AdsNotificationEx [▶ 993]	Occurs when the ADS devices sends a notification to the client.
 	AdsSumNotification [▶ 994]	Occurs when Notifications are send (bundled notifications)

Reference

- [IAdsNotifications Interface](#) [[▶ 966](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.35.2.1 IAdsNotifications.AdsNotification Event

Occurs when the ADS device sends a notification to the client.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
event EventHandler<AdsNotificationEventArgs> AdsNotification
```

Value

Type: [System.EventHandler.AdsNotificationEventArgs](#) [► 695].

Remarks

The Event Argument contains the raw data value of the notification, not marshalled to .NET types.

Examples

Example of receiving AdsNotification events.

Trigger on changed values by ADS Notifications

```
private async Task RegisterNotificationsAsync()
{
    CancellationToken cancel = CancellationToken.None;

    using (AdsClient client = new AdsClient())
    {
        // Add the Notification event handler
        client.AdsNotification += Client_AdsNotification2;

        // Connect to target
        client.Connect(AmsNetId.Local, 851);
        uint notificationHandle = 0;

        // Notification to a DINT Type (UINT32)
        // Check for change every 200 ms

        //byte[] notificationBuffer = new byte[sizeof(UInt32)];
        int size = sizeof(UInt32);

        ResultHandle result = await client.AddDeviceNotificationAsync("MAIN.nCounter", size, new NotificationSettings(AdsTransMode.OnChange, 200, 0), null, cancel);

        if (result.Succeeded)
        {
            notificationHandle = result.Handle;
            await Task.Delay(5000); // Wait asynchronously without blocking the UI Thread.
            // Unregister the Event / Handle
            ResultAds result2 = await client.DeleteDeviceNotificationAsync(notificationHandle, cancel);
        }
        client.AdsNotification -= Client_AdsNotification2;
    }
}

private void Client_AdsNotification2(object sender, AdsNotificationEventArgs e)
{
    // Or here we know about UDINT type --> can be marshalled as UINT32
    uint nCounter = BinaryPrimitives.ReadUInt32LittleEndian(e.Data.Span);

    // If Synchronization is needed (e.g. in Windows.Forms or WPF applications)
    // we could synchronize via SynchronizationContext into the UI Thread

    /*SynchronizationContext syncContext = SynchronizationContext.Current;
    _context.Post(status => someLabel.Text = nCounter.ToString(), null); // Non-blocking post */
}
```

Reference

[IAdsNotifications Interface](#) [► 966]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.35.2.2 IAdsNotifications.AdsNotificationError Event

Occurs when a exception has occurred during notification management.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
event EventHandler<AdsNotificationErrorEventArgs> AdsNotificationError
```

Value

Type: [System.EventHandler.AdsNotificationErrorEventArgs](#) [[▶ 694](#)].

Remarks

The occurrence of this event can have two different reasons:

1. Indicates an internal error occurred during Notification management.
2. The registered notification becomes invalid on the server, eg. after a PLC Download / Online Change. If the ADS Server detects that the (still registered) Notification Sender is getting invalid, it sends an error notification so that the client will be informed about detached notifications. The event arguments contains the 'AdsInvalidNotificationException' which describes the invalid notification handle by its 'AdsInvalidNotificationException.Handle' property.

Reference

[IAdsNotifications Interface](#) [[▶ 966](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsNotifications.AdsNotification](#) [[▶ 991](#)]

[IAdsNotifications.AdsNotificationEx](#) [[▶ 993](#)]

6.2.35.2.3 IAdsNotifications.AdsNotificationEx Event

Occurs when the ADS devices sends a notification to the client.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
event EventHandler<AdsNotificationExEventArgs> AdsNotificationEx
```

Value

Type: [System.EventHandler.AdsNotificationExEventArgs](#) [[▶ 699](#)].

Remarks

The Notification event arguments marshals the data value automatically to the specified .NET Type with ANY_TYPE marshallers.

Examples

Example of receiving AdsNotificationEx events.

Trigger on changed values by ADS Notifications

```

CancellationToken cancel = CancellationTokens.None;

using (AdsClient client = new AdsClient())
{
    client.AdsNotificationEx += Client_AdsNotificationEx;
    client.Connect(AmsNetId.Local, 851);

    // Add UDINT
    ResultHandle resultHandle = await client.AddDeviceNotificationExAsync("MAIN.udint", new NotificationSettings(AdsTransMode.OnChange, 200, 200), null, typeof(uint), null, cancel);
    await Task.Delay(5000, cancel); // Wait ....
    ResultAds resultHandleDelete = await client.DeleteDeviceNotificationAsync(resultHandle.Handle, cancel); // Unregister Event
}

```

Reference

[IAdsNotifications Interface \[► 966\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.35.2.4 IAdsNotifications.AdsSumNotification Event

Occurs when Notifications are send (bundled notifications)

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
event EventHandler<AdsSumNotificationEventArgs> AdsSumNotification
```

Value

Type: [System.EventHandler.AdsSumNotificationEventArgs](#).

Remarks

As an optimization, this event receives all ADS Notifications that occurred at one point in time together. As consequence, the overhead of handler code is reduced, what can be important if notifications are triggered in a high frequency and the event has to be synchronized to the UI thread context. Because multiple notifications are bound together, less thread synchronization is necessary. The [AdsNotification \[► 991\]](#) and [AdsNotificationEx \[► 993\]](#) events shouldn't be used when SumNotifications are registered, because they have an performance side effect to this AdsSumNotification event. The full performance is reached only, when all notifications are handled on this event.

Examples

Example of receiving AdsSumNotification events.

Trigger on changed values by ADS Notifications

```

private async Task RegisterSumNotificationsAsync()
{
    CancellationTokens cancel = CancellationTokens.None;

    using (AdsClient client = new AdsClient())
    {
        // Add the Notification event handler
        client.AdsSumNotification += Client_SumNotification;

        // Connect to target
        client.Connect(AmsNetId.Local, 851);
        uint notificationHandle = 0;
    }
}

```

```
// Notification to a DINT Type (UINT32)
// Check for change every 200 ms

//byte[] notificationBuffer = new byte[sizeof(UInt32)];

ResultHandle result = await client.AddDeviceNotificationAsync("MAIN.nCounter", sizeof(UInt32), new NotificationSettings(AdsTransMode.OnChange, 200, 0), null, cancel);

if (result.Succeeded)
{
    notificationHandle = result.Handle;
    await Task.Delay(5000); // Wait asynchronously without blocking the UI Thread.
    // Unregister the Event / Handle
    ResultAds result2 = await client.DeleteDeviceNotificationAsync(notificationHandle, cancel);
}
client.AdsNotification -= Client_AdsNotification2;
}

private void Client_SumNotification(object sender, AdsSumNotificationEventArgs e)
{
    // Timestamp of the Notification List
    DateTimeOffset dateTime = e.TimeStamp;

    // List of Raw ADS Notifications
    IList<Notification> notifications = e.Notifications;

    foreach(Notification notification in notifications)
    {
        // Notifications can be handled more efficiently, because they occur together
        // handler and can be transformed/
        synchronized in one step compared to AdsClient.AdsNotification events.
    }
}
```

Reference

[IAdsNotifications Interface \[► 966\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

[IAdsNotifications.AdsNotification \[► 991\]](#)

6.2.36 IAdsReadWrite Interface

Interface for ADS Read/Write access via IndexGroup / IndexOffset

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229





Syntax



C#

```
public interface IAdsReadWrite
```

The IAdsReadWrite type exposes the following members.

Methods

	Name	Description
	ReadAsync [► 996]	Reads the data asynchronously from specified IndexGroup/IndexOffset
	ReadWriteAsync [► 997]	Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer
	TryRead [► 998]	Reads data synchronously from an ADS device and writes it to the given stream.
	TryReadWrite [► 998]	Writes data synchronously to an ADS device and reads data from that device.

	Name	Description
	TryWrite [▶ 999]	Writes data synchronously to an ADS device.
	WriteAsync [▶ 1000]	Writes the data / Value asynchronously into the specified writeBuffer.







Reference

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.36.1 IAdsReadWrite Methods

The [IAdsReadWrite \[▶ 995\]](#) type exposes the following members.

Methods

	Name	Description
	ReadAsync [▶ 996]	Reads the data asynchronously from specified IndexGroup/IndexOffset
	ReadWriteAsync [▶ 997]	Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer
	TryRead [▶ 998]	Reads data synchronously from an ADS device and writes it to the given stream.
	TryReadWrite [▶ 998]	Writes data synchronously to an ADS device and reads data from that device.
	TryWrite [▶ 999]	Writes data synchronously to an ADS device.
	WriteAsync [▶ 1000]	Writes the data / Value asynchronously into the specified writeBuffer.

Reference

[IAdsReadWrite Interface \[▶ 995\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.36.1.1 IAdsReadWrite.ReadAsync Method

Reads the data asynchronously from specified IndexGroup/IndexOffset

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultRead> ReadAsync (
    uint indexGroup,
    uint indexOffset,
    Memory<byte> buffer,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.

buffer	Type: System.Memory.Byte . The read buffer, memory area where the data is written.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultRead](#) [▶ 1143].

A task that represents the asynchronous read operation. The [ResultRead](#) [▶ 1143] parameter contains the total number of bytes read into the buffer ([ReadBytes](#) [▶ 1145]) and the [ErrorCode](#) [▶ 1120] after execution..

Reference

[IAdsReadWrite Interface](#) [▶ 995]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.36.1.2 IAdsReadWrite.ReadWriteAsync Method

Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultReadWrite> ReadWriteAsync (
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
readBuffer	Type: System.Memory.Byte . The read buffer.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultReadWrite](#) [▶ 1163].

A task that represents the asynchronous 'ReadWrite' operation. The [ResultReadWrite](#) [▶ 1163] parameter contains the total number of bytes read into the buffer ([ReadBytes](#) [▶ 1145]) and the [ErrorCode](#) [▶ 1120] after execution.

Reference

[IAdsReadWrite Interface](#) [▶ 995]

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.36.1.3 IAdsReadWrite.TryRead Method

Reads data synchronously from an ADS device and writes it to the given stream.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryRead(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> buffer,
    out int readBytes
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
buffer	Type: System.Memory.Byte . The read buffer / memory area where the read data is written.
readBytes	Type: System.Int32 . The number of read bytes.

Return Value

Type: [AdsErrorCode \[▶ 664\]](#)

The ADS error code.

Reference

[IAdsReadWrite Interface \[▶ 995\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.36.1.4 IAdsReadWrite.TryReadWrite Method

Writes data synchronously to an ADS device and reads data from that device.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryReadWrite(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer,
    out int readBytes
)
```

Parameters

indexGroup	Type: System.UInt32 The index group number of the requested ADS service.
indexOffset	Type: System.UInt32 The index offset number of the requested ADS service.
readBuffer	Type: System.Memory.Byte . The read buffer.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.
readBytes	Type: System.Int32 . The read bytes.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS Error code.

Reference

[IAdsReadWrite Interface](#) [[▶ 995](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.36.1.5 IAdsReadWrite.TryWrite Method

Writes data synchronously to an ADS device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
AdsErrorCode TryWrite(
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The data buffer to be written.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS error code.

Reference

[IAdsReadWrite Interface](#) [[▶ 995](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.36.1.6 IAdsReadWrite.WriteAsync Method

Writes the data / Value asynchronously into the specified writeBuffer.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultWrite> WriteAsync(
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory<byte> writeBuffer,
    CancellationToken cancel
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite](#) [[▶ 1187](#)].

A task that represents the asynchronous 'Write' operation. The [ResultWrite](#) [[▶ 1187](#)] parameter contains the [ErrorCode](#) [[▶ 1120](#)] after execution.

Reference

[IAdsReadWrite Interface](#) [[▶ 995](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.37 IAdsReadWrite2 Interface

Interface for ADS Read/Write access via IndexGroup / IndexOffset

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax










C#

```
public interface IAdsReadWrite2 : IAdsReadWrite
```

The IAdsReadWrite2 type exposes the following members.

Methods

	Name	Description
	Read [▶ 1002]	Reads data synchronously from an ADS device and writes it to the given readBuffer

	Name	Description
	ReadAsync [▸ 996]	Reads the data asynchronously from specified IndexGroup/IndexOffset (Inherited from IAdsReadWrite [▸ 995].)
	ReadWrite [▸ 1002]	Writes data synchronously to an ADS device and then Reads data from this device into the readBuffer
	ReadWriteAsync [▸ 997]	Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer (Inherited from IAdsReadWrite [▸ 995].)
	TryRead [▸ 998]	Reads data synchronously from an ADS device and writes it to the given stream. (Inherited from IAdsReadWrite [▸ 995].)
	TryReadWrite [▸ 998]	Writes data synchronously to an ADS device and reads data from that device. (Inherited from IAdsReadWrite [▸ 995].)
	TryWrite [▸ 999]	Writes data synchronously to an ADS device. (Inherited from IAdsReadWrite [▸ 995].)
	Write(UInt32, UInt32) [▸ 1003]	Triggers a 'Write' call to the ADS device at the specified address.
	Write(UInt32, UInt32, ReadOnlyMemory.BYTE) [▸ 1004]	Writes data synchronously to an ADS device.
	WriteAsync [▸ 1000]	Writes the data / Value asynchronously into the specified writeBuffer. (Inherited from IAdsReadWrite [▸ 995].)









Reference



[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.37.1 IAdsReadWrite2 Methods

The [IAdsReadWrite2 \[▸ 1000\]](#) type exposes the following members.

Methods

	Name	Description
	Read [▸ 1002]	Reads data synchronously from an ADS device and writes it to the given readBuffer
	ReadAsync [▸ 996]	Reads the data asynchronously from specified IndexGroup/IndexOffset (Inherited from IAdsReadWrite [▸ 995].)
	ReadWrite [▸ 1002]	Writes data synchronously to an ADS device and then Reads data from this device into the readBuffer
	ReadWriteAsync [▸ 997]	Read/Writes data asynchronously to/from the specified writeBuffer, readBuffer (Inherited from IAdsReadWrite [▸ 995].)
	TryRead [▸ 998]	Reads data synchronously from an ADS device and writes it to the given stream. (Inherited from IAdsReadWrite [▸ 995].)
	TryReadWrite [▸ 998]	Writes data synchronously to an ADS device and reads data from that device. (Inherited from IAdsReadWrite [▸ 995].)
	TryWrite [▸ 999]	Writes data synchronously to an ADS device. (Inherited from IAdsReadWrite [▸ 995].)
	Write(UInt32, UInt32) [▸ 1003]	Triggers a 'Write' call to the ADS device at the specified address.

	Name	Description
	Write(UInt32, UInt32, ReadOnlyMemory<Byte>.) [▶ 1004]	Writes data synchronously to an ADS device.
	WriteAsync [▶ 1000]	Writes the data / Value asynchronously into the specified writeBuffer. (Inherited from IAdsReadWrite [▶ 995].)

Reference

[IAdsReadWrite2 Interface](#) [[▶ 1000](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.37.1.1 IAdsReadWrite2.Read Method

Reads data synchronously from an ADS device and writes it to the given readBuffer

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Read(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
readBuffer	Type: System.Memory.Byte . Memory location, where to read the data.

Return Value

Type: [Int32](#)

Number of successfully returned (read) data bytes.

Reference

[IAdsReadWrite2 Interface](#) [[▶ 1000](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.37.1.2 IAdsReadWrite2.ReadWrite Method

Writes data synchronously to an ADS device and then Reads data from this device into the readBuffer

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int ReadWrite (
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

indexGroup	Type: System.UInt32 The index group number of the requested ADS service.
indexOffset	Type: System.UInt32 The index offset number of the requested ADS service.
readBuffer	Type: System.Memory.Byte . The read buffer.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.

Return Value

Type: [Int32](#)
Number of successfully returned (read) data bytes.



Reference

[IAdsReadWrite2 Interface](#) [► 1000]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.37.1.3 IAdsReadWrite2.Write Method

Overload List

	Name	Description
	Write(UInt32, UInt32) [► 1003]	Triggers a 'Write' call to the ADS device at the specified address.
	Write(UInt32, UInt32, ReadOnlyMemory.Byte) [► 1004]	Writes data synchronously to an ADS device.

Reference

[IAdsReadWrite2 Interface](#) [► 1000]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.37.1.3.1 IAdsReadWrite2.Write Method (UInt32, UInt32)

Triggers a 'Write' call to the ADS device at the specified address.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void Write(
    uint indexGroup,
    uint indexOffset
)
```

Parameters

indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.

Reference

[IAdsReadWrite2 Interface](#) [► 1000]

[Write Overload](#) [► 1003]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.37.1.3.2 IAdsReadWrite2.Write Method (UInt32, UInt32, ReadOnlyMemory`1)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
void Write(
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory writeBuffer
)
```

Parameters

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
writeBuffer	Type: ReadOnlyMemory

Reference

[IAdsReadWrite2 Interface](#) [► 1000]

[Write Overload](#) [► 1003]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.37.1.3.3 IAdsReadWrite2.Write Method (UInt32, UInt32, ReadOnlyMemory.Byte.)

Writes data synchronously to an ADS device.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void Write(
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The data to write.

Reference

[IAdsReadWrite2 Interface \[▸ 1000\]](#)

[Write Overload \[▸ 1003\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.38 IAdsReadWriteTimeoutAccess Interface

Interface IAdsReadWriteTimeoutAccess

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229








Syntax


C#

```
public interface IAdsReadWriteTimeoutAccess
```

The IAdsReadWriteTimeoutAccess type exposes the following members.

Methods

	Name	Description
	Read [▸ 1006]	Reads data synchronously from an ADS device and writes it to the given stream.
	ReadAny [▸ 1007]	Reads data synchronously from an ADS device and writes it to an object.
	ReadWrite [▸ 1008]	Writes data synchronously to an ADS device and then Reads data from this device.
	TryRead [▸ 1008]	Reads data synchronously from an ADS device and writes it to the given stream.
	TryReadWrite [▸ 1009]	Writes data synchronously to an ADS device and then Reads data from this device.
	TryWrite [▸ 1010]	Writes data synchronously to an ADS device.
	Write [▸ 1011]	Writes data synchronously to an ADS device.

	Name	Description
	WriteAny [▸ 1011]	Writes an object synchronously to an ADS device.

Remarks

For internal use only.









Reference

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.38.1 IAdsReadWriteTimeoutAccess Methods

The [IAdsReadWriteTimeoutAccess \[▸ 1005\]](#) type exposes the following members.

Methods

	Name	Description
	Read [▸ 1006]	Reads data synchronously from an ADS device and writes it to the given stream.
	ReadAny [▸ 1007]	Reads data synchronously from an ADS device and writes it to an object.
	ReadWrite [▸ 1008]	Writes data synchronously to an ADS device and then Reads data from this device.
	TryRead [▸ 1008]	Reads data synchronously from an ADS device and writes it to the given stream.
	TryReadWrite [▸ 1009]	Writes data synchronously to an ADS device and then Reads data from this device.
	TryWrite [▸ 1010]	Writes data synchronously to an ADS device.
	Write [▸ 1011]	Writes data synchronously to an ADS device.
	WriteAny [▸ 1011]	Writes an object synchronously to an ADS device.

Reference

[IAdsReadWriteTimeoutAccess Interface \[▸ 1005\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.38.1.1 IAdsReadWriteTimeoutAccess.Read Method

Reads data synchronously from an ADS device and writes it to the given stream.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Read(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    int timeout
)
```

Parameters

indexGroup	Type: System.UInt32 The index group number of the requested ADS service.
indexOffset	Type: System.UInt32 The index offset number of the requested ADS service.
readBuffer	Type: System.Memory.Byte . The read buffer.
timeout	Type: System.Int32 The timeout.

Return Value

Type: [Int32](#)
Number of successfully returned data bytes.

Reference

[IAdsReadWriteTimeoutAccess Interface \[▶ 1005\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.38.1.2 IAdsReadWriteTimeoutAccess.ReadAny Method

Reads data synchronously from an ADS device and writes it to an object.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object ReadAny(
    uint indexGroup,
    uint indexOffset,
    Type type,
    int[] args,
    int timeout
)
```

Parameters

indexGroup	Type: System.UInt32 Index group of the ADS variable.
indexOffset	Type: System.UInt32 Index offset of the ADS variable.
type	Type: System.Type Type of the object to be read.
args	Type: .System.Int32 . Additional arguments.
timeout	Type: System.Int32 The timeout.

Return Value

Type: [Object](#)
The Value of the data marshalled to the specified type.

Remarks

If the Type of the object to be read is a string type, the first element of the parameter args specifies the number of characters of the string. If the Type of the object to be read is an array type, the number of elements for each dimension has to be specified in the parameter args. The type is limited to Primitive types ('AnyType').

Reference

[IAdsReadWriteTimeoutAccess Interface \[► 1005\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.38.1.3 IAdsReadWriteTimeoutAccess.ReadWrite Method

Writes data synchronously to an ADS device and then Reads data from this device.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
int ReadWrite (
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer,
    int timeout
)
```

Parameters

indexGroup	Type: System.UInt32 The index group number of the requested ADS service.
indexOffset	Type: System.UInt32 The index offset number of the requested ADS service.
readBuffer	Type: System.Memory.Byte . The read buffer.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.
timeout	Type: System.Int32 The timeout.

Return Value

Type: [Int32](#)

Number of successfully returned data bytes.

Reference

[IAdsReadWriteTimeoutAccess Interface \[► 1005\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.38.1.4 IAdsReadWriteTimeoutAccess.TryRead Method

Reads data synchronously from an ADS device and writes it to the given stream.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```

AdsErrorCode TryRead(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    int timeout,
    out int readBytes
)
    
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
readBuffer	Type: System.Memory.Byte . The read buffer.
timeout	Type: System.Int32 The timeout.
readBytes	Type: System.Int32 . The read bytes.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
The ADS Error code.

Reference

[IAdsReadWriteTimeoutAccess Interface](#) [[▶ 1005](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.38.1.5 IAdsReadWriteTimeoutAccess.TryReadWrite Method

Writes data synchronously to an ADS device and then Reads data from this device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```

AdsErrorCode TryReadWrite(
    uint indexGroup,
    uint indexOffset,
    Memory<byte> readBuffer,
    ReadOnlyMemory<byte> writeBuffer,
    int timeout,
    out int readBytes
)
    
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.

readBuffer	Type: System.Memory.Byte . The read stream.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write stream.
timeout	Type: System.Int32 The timeout.
readBytes	Type: System.Int32 . The read bytes.

Return Value

Type: [AdsErrorCode](#) [▶ [664](#)]
The ADS Error code.

Reference

[IAdsReadWriteTimeoutAccess Interface](#) [▶ [1005](#)]

[TwinCAT.Ads Namespace](#) [▶ [179](#)]

6.2.38.1.6 IAdsReadWriteTimeoutAccess.TryWrite Method

Writes data synchronously to an ADS device.

Namespace: [TwinCAT.Ads](#) [▶ [179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
AdsErrorCode TryWrite(
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory<byte> writeBuffer,
    int timeout
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.
timeout	Type: System.Int32 The timeout.

Return Value

Type: [AdsErrorCode](#) [▶ [664](#)]
The ADS Error code.

Reference

[IAdsReadWriteTimeoutAccess Interface](#) [▶ [1005](#)]

[TwinCAT.Ads Namespace](#) [▶ [179](#)]

6.2.38.1.7 IAdsReadWriteTimeoutAccess.Write Method

Writes data synchronously to an ADS device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void Write(
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory<byte> writeBuffer,
    int timeout
)
```

Parameters

indexGroup	Type: System.UInt32 The index group number of the requested ADS service.
indexOffset	Type: System.UInt32 The index offset number of the requested ADS service.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer.
timeout	Type: System.Int32 The timeout.

Reference

[IAdsReadWriteTimeoutAccess Interface](#) [[▶ 1005](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.38.1.8 IAdsReadWriteTimeoutAccess.WriteAny Method

Writes an object synchronously to an ADS device.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void WriteAny(
    uint indexGroup,
    uint indexOffset,
    Object value,
    int[] args,
    int timeout
)
```

Parameters

indexGroup	Type: System.UInt32 Contains the index group number of the requested ADS service.
indexOffset	Type: System.UInt32 Contains the index offset number of the requested ADS service.
value	Type: System.Object Object to write to the ADS device (Primitive type, 'AnyType')

args	Type: <u>System.Int32</u> . Additional arguments.
timeout	Type: <u>System.Int32</u> The timeout.

Return Value

Type:
System.Int32.

Remarks

If the Type of the object to be written is a string type, the first element of parameter args specifies the number of characters of the string. If the Type of the object to be written is an array type, the number of elements for each dimension has to be specified in the parameter args. The value is limited to Primitive types ('AnyType').

Reference

[IAdsReadWriteTimeoutAccess Interface \[► 1005\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.39 IAdsRpcInvoke Interface

Interface IAdsRpcInvoke

Namespace: [TwinCAT.Ads \[► 179\]](#)




Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229









Syntax

C#

```
public interface IAdsRpcInvoke
```

Methods

	Name	Description
	InvokeRpcMethodAsync(String, String, Object, CancellationToken) [► 1021]	Invokes the specified RPC Method asynchronously
	InvokeRpcMethodAsync(IRpcCallableInstance, IRpcMethod, Object, AnyTypeSpecifier, AnyTypeSpecifier, CancellationToken) [► 1024]	Invokes the specified RPC Method asynchronously
	InvokeRpcMethodAsync(String, String, Object, Any	Invokes the specified RPC Method asynchronously

	Name	Description
	TypeSpecifier, AnyTypeSpecifier, CancellationToken) [▶ 1022]	
 	TryInvokeRpcMethod(String, String, Object, Object.) [▶ 1026]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(String, String, Object, Object.) [▶ 1028]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(IRpcCallableInstance, IRpcMethod, Object, AnyTypeSpecifier, AnyTypeSpecifier, Object, Object.) [▶ 1031]	Invokes the rpc method.
 	TryInvokeRpcMethod(String, String, Object, AnyTypeSpecifier, AnyTypeSpecifier, Object, Object.) [▶ 1030]	Invokes the rpc method.

Remarks



This interface is used to invoke ADS RPC Method calls. This can be done 'symbolic' via the Method name or the MethodID of the method on the specified symbol. To activate the RPC Access within the PLC environment, its dataType (Structure, FB) must be marked with the PlcAttribute 'TcRpcEnable' to enable RpcMethods.





Reference

TwinCAT.Ads Namespace [▶ 179]

6.2.39.1 IAdsRpcInvoke Methods

Methods

	Name	Description
 	InvokeRpcMethodAsync(String,	Invokes the specified RPC Method asynchronously

	Name	Description
	String, .Object, CancellationTok en) [▶ 1021]	
 	InvokeRpcMethodA sync(IRpcCallableIns tance, IRpcMethod, .Objec t, .AnyTypeSpecifier , .AnyTypeSpecifier, CancellationTok en) [▶ 1024]	Invokes the specified RPC Method asynchronously
 	InvokeRpcMethodA sync(String, String, .Object, .Any TypeSpecifier, AnyTypeSpecifier, CancellationTok en) [▶ 1022]	Invokes the specified RPC Method asynchronously
 	TryInvokeRpcMetho d(String, String, .Object, Object.) [▶ 1026]	Invokes the specified RPC Method
 	TryInvokeRpcMetho d(String, String, .Object, .Obj ect., Object.) [▶ 1028]	Invokes the specified RPC Method
 	TryInvokeRpcMetho d(IRpcCallableInstan ce, IRpcMethod, .Objec t, .AnyTypeSpecifier , AnyTypeSpecifier, . Object., Object.) [▶ 1031]	Invokes the rpc method.
 	TryInvokeRpcMetho d(String, String, .Object, .Any TypeSpecifier, AnyTypeSpecifier, . Object., Object.) [▶ 1030]	Invokes the rpc method.




Reference

[IAdsRpcInvoke Interface \[▶ 1012\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.39.1.1 IAdsRpcInvoke.InvokeRpcMethod Method

Overload List

	Name	Description
	InvokeRpcMethod(String, .Object.) [▶ 1015]	Invokes the specified RPC Method
	InvokeRpcMethod(String, .Object., .Object.) [▶ 1017]	Invokes the specified RPC Method
	InvokeRpcMethod(String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier., .Object.) [▶ 1018]	Invokes the specified RPC Method

Reference

[IAdsRpcInvoke Interface \[▶ 1012\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.39.1.1.1 IAdsRpcInvoke.InvokeRpcMethod Method (String, String, .Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
Object InvokeRpcMethod(
    string symbolPath,
    string methodName,
    Object[] inParameters
)
```

Parameters

- symbolPath Type: [System.String](#)
The symbol path.
- methodName Type: [System.String](#)
The method name.
- inParameters Type: [.System.Object.](#)
The input parameters or NULL

Return Value

Type: [Object](#)

The return value of the Method (as object).

Remarks

This method only supports primitive data types as inParameters. Any available outparameters will be ignored. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] { (short) 3, (short) 4 });

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```

Reference

[IAdsRpcInvoke Interface \[► 1012\]](#)

[InvokeRpcMethod Overload \[▸ 1015\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.39.1.1.2 IAdsRpcInvoke.InvokeRpcMethod Method (String, String, .Object., .Object..)

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
Object InvokeRpcMethod(  
    string symbolPath,  
    string methodName,  
    Object[] inParameters,  
    out Object[] outParameters  
)
```

Parameters

symbolPath	Type: System.String The symbol path.
methodName	Type: System.String The method name.
inParameters	Type: .System.Object. The input parameters or NULL
outParameters	Type: .System.Object.. The output parameters.

Return Value

Type: [Object](#)
The return value of the Method (as object).

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier \[▸ 2094\]](#) specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram  
{  
    /// <summary>  
    /// Defines the entry point of the application.  
    /// </summary>  
    /// <param name="args">The arguments.</param>  
    static void Main(string[] args)  
    {  
        // Get the AdsAddress from command-line arguments  
        AmsAddress address = ArgParser.Parse(args);  
  
        using (AdsClient client = new AdsClient())  
        {  
            //client.Synchronize = false;
```

```

// Connect to the target device
client.Connect(address);

SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.VirtualTree);
ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

// Get the Symbols (Dynamic Symbols)

IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

// Call a Method that has the following signature (within MAIN Program)
/* {attribute 'TcRpcEnable'}
METHOD PUBLIC M_Add : INT
VAR_INPUT
    i1 : INT := 0;
    i2 : INT := 0;
END_VAR
*/

short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

// Call a Method that has no parameter and returns VOID
main.InvokeRpcMethod("M_Method1", new object[] {});

//Browsing RpcMethods
foreach(IRpcMethod method in main.RpcMethods)
{
    string methodName = method.Name;

    foreach(IRpcMethodParameter parameter in method.Parameters)
    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
}

```

Reference

[IAdsRpcInvoke Interface \[► 1012\]](#)

[InvokeRpcMethod Overload \[► 1015\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.39.1.1.3 IAdsRpcInvoke.InvokeRpcMethod Method (String, String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object..)

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```

Object InvokeRpcMethod(
    string symbolPath,
    string methodName,
    Object[] inParameters,
    AnyTypeSpecifier[] outSpecifiers,
    AnyTypeSpecifier retSpecifier,
    out Object[] outParameters
)

```

Parameters

symbolPath	Type: System.String The symbol path.
methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094]. The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094] The ret specifier (specifying the return value) or NULL.
outParameters	Type: .System.Object . The out parameters.

Return Value

Type: [Object](#)
The return value of the Method (as object).

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});
        }
    }
}
```

```

// Call a Method that has no parameter and returns VOID
main.InvokeRpcMethod("M_Method1", new object[] {});

//Browsing RpcMethods
foreach(IRpcMethod method in main.RpcMethods)
{
    string methodName = method.Name;

    foreach(IRpcMethodParameter parameter in method.Parameters)
    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
}

```

Reference




[IAdsRpcInvoke Interface \[► 1012\]](#)

[InvokeRpcMethod Overload \[► 1015\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.39.1.2 IAdsRpcInvoke.InvokeRpcMethodAsync Method

Overload List

	Name	Description
	InvokeRpcMethodAsync(String, String, Object, CancellationToken) [► 1021]	Invokes the specified RPC Method asynchronously
	InvokeRpcMethodAsync(IRpcCallableInstance, IRpcMethod, Object, AnyTypeSpecifier, AnyTypeSpecifier, CancellationToken) [► 1024]	Invokes the specified RPC Method asynchronously
	InvokeRpcMethodAsync(String, String, Object, AnyTypeSpecifier, AnyTypeSpecifier, CancellationToken) [► 1022]	Invokes the specified RPC Method asynchronously

Reference

[IAdsRpcInvoke Interface \[► 1012\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.39.1.2.1 IAdsRpcInvoke.InvokeRpcMethodAsync Method (String, String, .Object., CancellationToken)

Invokes the specified RPC Method asynchronously

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultRpcMethod> InvokeRpcMethodAsync (
    string symbolPath,
    string methodName,
    Object[]? inParameters,
    CancellationToken cancel
)
```

Parameters

symbolPath	Type: System.String The symbol/Instance path of the symbol.
methodName	Type: System.String The method name.
inParameters	Type: .System.Object. The parameters.
cancel	Type: System.Threading.CancellationToken The cancellation token

Return Value

Type: [Task.ResultRpcMethod](#) [[▶ 1176](#)].

A task that represents the asynchronous 'InvokeRpcMethod' operation. The [ResultRpcMethod](#) [[▶ 1176](#)] results contains the return value together with the output parameters.

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [[▶ 2094](#)] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);
        }
    }
}
```

```

        // Get the Symbols (Dynamic Symbols)

        IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
the PLC Program

        // Call a Method that has the following signature (within MAIN Program)
        /* {attribute 'TcRpcEnable'}
        METHOD PUBLIC M_Add : INT
        VAR_INPUT
            i1 : INT := 0;
            i2 : INT := 0;
        END_VAR
        */

        short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

        // Call a Method that has no parameter and returns VOID
        main.InvokeRpcMethod("M_Method1", new object[] {});

        //Browsing RpcMethods
        foreach(IRpcMethod method in main.RpcMethods)
        {
            string methodName = method.Name;

            foreach(IRpcMethodParameter parameter in method.Parameters)
            {
                string parameterName = parameter.Name;
                string parameterType = parameter.TypeName;
            }
        }
    }
}

```

Reference

[IAdsRpcInvoke Interface \[► 1012\]](#)

[InvokeRpcMethodAsync Overload \[► 1020\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.39.1.2.2 IAdsRpcInvoke.InvokeRpcMethodAsync Method (String, String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, CancellationToken)

Invokes the specified RPC Method asynchronously

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```

Task<ResultRpcMethod> InvokeRpcMethodAsync (
    string symbolPath,
    string methodName,
    Object[]? inParameters,
    AnyTypeSpecifier[]? outSpecifiers,
    AnyTypeSpecifier? retSpecifier,
    CancellationToken cancel
)

```

Parameters

symbolPath	Type: System.String The symbol/Instance path of the symbol.
------------	--

methodName	Type: System.String The method name.
inParameters	Type: System.Object . The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094]. The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094] The ret specifier (specifying the return value) or NULL.
cancel	Type: System.Threading.CancellationToken The cancellation token

Return Value

Type: [Task.ResultRpcMethod](#) [[▶ 1176](#)].

A task that represents the asynchronous 'InvokeRpcMethod' operation. The [ResultRpcMethod](#) [[▶ 1176](#)] results contains the return value together with the output parameters.

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set. [ReturnValue](#) [[▶ 1179](#)] and the [ErrorCode](#) [[▶ 1120](#)] of the ADS communication after execution.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});
        }
    }
}
```

```

//Browsing RpcMethods
foreach(IRpcMethod method in main.RpcMethods)
{
    string methodName = method.Name;

    foreach(IRpcMethodParameter parameter in method.Parameters)
    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
}
}

```

Reference

[IAdsRpcInvoke Interface \[► 1012\]](#)

[InvokeRpcMethodAsync Overload \[► 1020\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.39.1.2.3 IAdsRpcInvoke.InvokeRpcMethodAsync Method (IRpcCallableInstance, IRpcMethod, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, CancellationToken)

Invokes the specified RPC Method asynchronously

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```

Task<ResultRpcMethod> InvokeRpcMethodAsync (
    IRpcCallableInstance symbol,
    IRpcMethod method,
    Object[]? inParameters,
    AnyTypeSpecifier[]? outSpecifiers,
    AnyTypeSpecifier? retSpecifier,
    CancellationToken cancel
)

```

Parameters

symbol	Type: TwinCAT.TypeSystem.IRpcCallableInstance [► 2606] The RPC callable symbol."
method	Type: TwinCAT.TypeSystem.IRpcMethod [► 2625] The method.
inParameters	Type: .System.Object. The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [► 2094] . The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [► 2094] The ret specifier (specifying the return value) or NULL.
cancel	Type: System.Threading.CancellationToken The cancellation token

Return Value

Type: [Task.ResultRpcMethod](#) [► 1176].

A task that represents the asynchronous 'InvokeRpcMethod' operation. The [ResultRpcMethod](#) [► 1176] results contains the return value together with the output parameters.

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set. [ReturnValue](#) [► 1179] and the [ErrorCode](#) [► 1120] of the ADS communication after execution.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */









            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```

Reference[IAdsRpcInvoke Interface \[► 1012\]](#)[InvokeRpcMethodAsync Overload \[► 1020\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.39.1.3 IAdsRpcInvoke.TryInvokeRpcMethod Method****Overload List**

	Name	Description
 	TryInvokeRpcMethod(String, String, .Object., Object.) [► 1026]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(String, String, .Object., .Object., Object.) [► 1028]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(IRpcCallableInstance, IRpcMethod, .Object., .AnyTypeSpecifier, .AnyTypeSpecifier, .Object., Object.) [► 1031]	Invokes the rpc method.
 	TryInvokeRpcMethod(String, String, .Object., .AnyTypeSpecifier, .AnyTypeSpecifier, .Object., Object.) [► 1030]	Invokes the rpc method.

Reference[IAdsRpcInvoke Interface \[► 1012\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.39.1.3.1 IAdsRpcInvoke.TryInvokeRpcMethod Method (String, String, .Object., Object.)**

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```

AdsErrorCode TryInvokeRpcMethod(
    string symbolPath,
    string methodName,
    Object[]? inParameters,
    out Object?? retValue
)
    
```

Parameters

symbolPath	Type: <u>System.String</u> The symbol path.
methodName	Type: <u>System.String</u> The method name.
inParameters	Type: <u>.System.Object</u> . The parameters.
retValue	Type: <u>System.Object</u> . The return value of the RPC method as object.

Return Value

Type: AdsErrorCode [▶ 664]
The ADS Error Code.

Remarks

Because this overload doesn't provide any AnyTypeSpecifier [▶ 2094] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```

class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            */
        }
    }
}
    
```

```

    END_VAR
    */

    short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

    // Call a Method that has no parameter and returns VOID
    main.InvokeRpcMethod("M_Method1", new object[] {});

    //Browsing RpcMethods
    foreach(IRpcMethod method in main.RpcMethods)
    {
        string methodName = method.Name;

        foreach(IRpcMethodParameter parameter in method.Parameters)
        {
            string parameterName = parameter.Name;
            string parameterType = parameter.TypeName;
        }
    }
}
}
}

```

Reference

[IAdsRpcInvoke Interface](#) [► 1012]

[TryInvokeRpcMethod Overload](#) [► 1026]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.39.1.3.2 IAdsRpcInvoke.TryInvokeRpcMethod Method (String, String, .Object., .Object., Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```

AdsErrorCode TryInvokeRpcMethod(
    string symbolPath,
    string methodName,
    Object[]? inParameters,
    out Object[]? outParameters,
    out Object?? retValue
)

```

Parameters

symbolPath	Type: System.String The symbol path.
methodName	Type: System.String The method name.
inParameters	Type: .System.Object. The parameters.
outParameters	Type: .System.Object.. The out parameters.
retValue	Type: System.Object. The return value of the RPC method as object.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The ADS Error Code.

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [[▶ 2094](#)] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```

Reference

[IAdsRpcInvoke Interface](#) [[▶ 1012](#)]

[TryInvokeRpcMethod Overload \[▶ 1026\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.39.1.3.3 IAdsRpcInvoke.TryInvokeRpcMethod Method (String, String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object., Object.)

Invokes the rpc method.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryInvokeRpcMethod(
    string symbolPath,
    string methodName,
    Object[]? inParameters,
    AnyTypeSpecifier[]? outSpecifiers,
    AnyTypeSpecifier? retSpecifier,
    out Object[]? outParameters,
    out Object?? retValue
)
```

Parameters

symbolPath	Type: System.String The symbol.
methodName	Type: System.String Name of the method.
inParameters	Type: .System.Object. The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094] . The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094] The ret specifier (specifying the return value) or NULL.
outParameters	Type: .System.Object.. The out parameters.
retValue	Type: System.Object. The return value of the RPC method./>

Return Value

Type: [AdsErrorCode \[▶ 664\]](#)

AdsErrorCode.

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```

class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}

```

Reference

[IAdsRpcInvoke Interface \[► 1012\]](#)

[TryInvokeRpcMethod Overload \[► 1026\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.39.1.3.4 IAdsRpcInvoke.TryInvokeRpcMethod Method (IRpcCallableInstance, IRpcMethod, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object., Object.)

Invokes the rpc method.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```

AdsErrorCode TryInvokeRpcMethod(
    IRpcCallableInstance symbol,
    IRpcMethod method,
    Object[]? inParameters,
    AnyTypeSpecifier[]? outSpecifiers,
    AnyTypeSpecifier? retSpecifier,
    out Object[]? outParameters,
    out Object?? retValue
)

```

Parameters

symbol	Type: TwinCAT.TypeSystem.IRpcCallableInstance [▶ 2606] The RPC callable symbol
method	Type: TwinCAT.TypeSystem.IRpcMethod [▶ 2625] The method.
inParameters	Type: .System.Object . The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094]. The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094] The ret specifier (specifying the return value) or NULL.
outParameters	Type: .System.Object . The out parameters.
retValue	Type: System.Object . The return value of the RPC method./>

Return Value

Type: [AdsErrorCode](#) [▶ 664]
AdsErrorCode.

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outParameters, are allowed to be empty or NULL.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```

class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);
        }
    }
}

```



```

// Get the Symbols (Dynamic Symbols)

IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
the PLC Program

// Call a Method that has the following signature (within MAIN Program)
/* {attribute 'TcRpcEnable'}
METHOD PUBLIC M_Add : INT
VAR_INPUT
    i1 : INT := 0;
    i2 : INT := 0;
END_VAR
*/

short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

// Call a Method that has no parameter and returns VOID
main.InvokeRpcMethod("M_Method1", new object[] {});

//Browsing RpcMethods
foreach(IRpcMethod method in main.RpcMethods)
{
    string methodName = method.Name;

    foreach(IRpcMethodParameter parameter in method.Parameters)
    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
}

```

Reference

[IAdsRpcInvoke Interface \[▶ 1012\]](#)

[TryInvokeRpcMethod Overload \[▶ 1026\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.40 IAdsSession Interface

Interface IAdsSession

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#




```







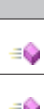


public interface IAdsSession : ISession,
    IConnectionStateProvider, ISymbolServerProvider

```






The IAdsSession type exposes the following members.

Properties


	Name	Description
	Address [▶ 1035]	Gets the Address specifier of the Session / connection
	AddressSpecifier [▶ 94]	Gets the communication endpoint address string representation. (Inherited from ISession [▶ 92].)
	Connection [▶ 94]	Gets the Connection object. (Inherited from ISession [▶ 92].)

	Name	Description
	ConnectionState [▶ 90]	Gets the current Connection state of the IConnectionStateProvider [▶ 89] (Inherited from IConnectionStateProvider [▶ 89].)
	EstablishedAt [▶ 95]	Gets the UTC time when the session was established. (Inherited from ISession [▶ 92].)
	Id [▶ 95]	Gets the Session Id (Inherited from ISession [▶ 92].)
	IsConnected [▶ 96]	Gets a value indicating whether the session is connected. (Inherited from ISession [▶ 92].)
	NetId [▶ 1036]	Gets the NetId of the Session
	Owner [▶ 1036]	Gets the Session owner.
	Port [▶ 1036]	Gets the Ams Port of the Session
	Provider [▶ 96]	Gets the Session Provider (Inherited from ISession [▶ 92].)
	SymbolServer [▶ 106]	Gets the symbol server. (Inherited from ISymbolServerProvider [▶ 106].)

Methods

	Name	Description
	Close [▶ 97]	Closes this ISession [▶ 92] (Inherited from ISession [▶ 92].)
	Connect [▶ 97]	Connects the session and returns the established IConnection [▶ 79] object. (Inherited from ISession [▶ 92].)
	ConnectAsync [▶ 99]	Connects the session and returns the established IConnection [▶ 79] object. (Inherited from ISession [▶ 92].)
	Disconnect [▶ 98]	Disconnects the ISession [▶ 92] (Inherited from ISession [▶ 92].)
	EnsureConnection [▶ 98]	Ensures, that the ISession [▶ 92] is connected and returns the IConnection [▶ 79] object. (Inherited from ISession [▶ 92].)

Events

	Name	Description
	ConnectionStateChanged [▶ 91]	Occurs when connection status of the IConnectionStateProvider [▶ 89] has been changed. (Inherited from IConnectionStateProvider [▶ 89].)

Reference














[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[TwinCAT.ISession](#) [[▶ 92](#)]

6.2.40.1 IAdsSession Properties

The [IAdsSession](#) [[▶ 1033](#)] type exposes the following members.

Properties

	Name	Description
	Address [▶ 1035]	Gets the Address specifier of the Session / connection
	AddressSpecifier [▶ 94]	Gets the communication endpoint address string representation. (Inherited from ISession [▶ 92].)
	Connection [▶ 94]	Gets the Connection object. (Inherited from ISession [▶ 92].)
 	ConnectionState [▶ 90]	Gets the current Connection state of the IConnectionStateProvider [▶ 89] (Inherited from IConnectionStateProvider [▶ 89].)
	EstablishedAt [▶ 95]	Gets the UTC time when the session was established. (Inherited from ISession [▶ 92].)
	Id [▶ 95]	Gets the Session Id (Inherited from ISession [▶ 92].)
	IsConnected [▶ 96]	Gets a value indicating whether the session is connected. (Inherited from ISession [▶ 92].)
	NetId [▶ 1036]	Gets the NetId of the Session
	Owner [▶ 1036]	Gets the Session owner.
	Port [▶ 1036]	Gets the Ams Port of the Session
	Provider [▶ 96]	Gets the Session Provider (Inherited from ISession [▶ 92].)
	SymbolServer [▶ 106]	Gets the symbol server. (Inherited from ISymbolServerProvider [▶ 106].)

Reference

[IAdsSession Interface](#) [[▶ 1033](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.40.1.1 IAdsSession.Address Property

Gets the Address specifier of the Session / connection

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AmsAddress Address { get; }
```

Property Value

Type: [AmsAddress](#) [[▶ 752](#)]

The address.

Reference

[IAdsSession Interface](#) [[▶ 1033](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.40.1.2 IAdsSession.NetId Property

Gets the NetId of the Session

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AmsNetId NetId { get; }
```

Property Value

Type: [AmsNetId](#) [[▶ 767](#)]

The net identifier.

Reference

[IAdsSession Interface](#) [[▶ 1033](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.40.1.3 IAdsSession.Owner Property

Gets the Session owner.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object? Owner { get; }
```

Property Value

Type: [Object](#)

The owner or NULL

Reference

[IAdsSession Interface](#) [[▶ 1033](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.40.1.4 IAdsSession.Port Property

Gets the Ams Port of the Session

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Port { get; }
```

Property Value

Type: [Int32](#)
The port.

Reference






[IAdsSession Interface](#) [[▶ 1033](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.40.2 IAdsSession Methods

The [IAdsSession](#) [[▶ 1033](#)] type exposes the following members.

Methods

	Name	Description
	Close [▶ 97]	Closes this ISession [▶ 92] (Inherited from ISession [▶ 92].)
	Connect [▶ 97]	Connects the session and returns the established IConnection [▶ 79] object. (Inherited from ISession [▶ 92].)
	ConnectAsync [▶ 99]	Connects the session and returns the established IConnection [▶ 79] object. (Inherited from ISession [▶ 92].)
	Disconnect [▶ 98]	Disconnects the ISession [▶ 92] (Inherited from ISession [▶ 92].)
	EnsureConnection [▶ 98]	Ensures, that the ISession [▶ 92] is connected and returns the IConnection [▶ 79] object. (Inherited from ISession [▶ 92].)

Reference


[IAdsSession Interface](#) [[▶ 1033](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.40.3 IAdsSession Events

The [IAdsSession](#) [[▶ 1033](#)] type exposes the following members.

Events

	Name	Description
	ConnectionStateChanged [▶ 91]	Occurs when connection status of the IConnectionStateProvider [▶ 89] has been changed. (Inherited from IConnectionStateProvider [▶ 89].)

Reference

[IAdsSession Interface](#) [[▶ 1033](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.41 IAdsSessionSettings Interface

Interface for ADS Session Settings

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229




Syntax

C#

```
public interface IAdsSessionSettings
```

The IAdsSessionSettings type exposes the following members.

Properties

	Name	Description
	ResurrectionTime [▶ 1038]	Gets or sets the resurrection time.
	SymbolLoader [▶ 1039]	Gets or sets the symbol loader settings
	Timeout [▶ 1039]	Gets the ADS timeout in milliseconds.

Reference




[TwinCAT.Ads Namespace](#) [▶ 179]

IAdsSessionSettings

6.2.41.1 IAdsSessionSettings Properties

The IAdsSessionSettings [▶ 1037] type exposes the following members.

Properties

	Name	Description
	ResurrectionTime [▶ 1038]	Gets or sets the resurrection time.
	SymbolLoader [▶ 1039]	Gets or sets the symbol loader settings
	Timeout [▶ 1039]	Gets the ADS timeout in milliseconds.

Reference

[IAdsSessionSettings Interface](#) [▶ 1037]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.41.1.1 IAdsSessionSettings.ResurrectionTime Property

Gets or sets the resurrection time.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
TimeSpan ResurrectionTime { get; set; }
```

Property Value

Type: [TimeSpan](#)
The resurrection time.

Reference

[IAdsSessionSettings Interface](#) [► 1037]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.41.1.2 IAdsSessionSettings.SymbolLoader Property

Gets or sets the symbol loader settings

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
SymbolLoaderSettings SymbolLoader { get; set; }
```

Property Value

Type: [SymbolLoaderSettings](#) [► 148]
The symbol loader.

Reference

[IAdsSessionSettings Interface](#) [► 1037]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.41.1.3 IAdsSessionSettings.Timeout Property

Gets the ADS timeout in milliseconds.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Timeout { get; set; }
```

Property Value

Type: [Int32](#)
The timeout.

Reference

[IAdsSessionSettings Interface](#) [► 1037]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.42 IAdsStateControl Interface

Interface for reading and controlling the ADS state.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]







Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public interface IAdsStateControl
```

Methods


	Name	Description
	TryWriteControl(StateInfo) [▶ 1042]	Changes the ADS status and the device status of an ADS server.
	TryWriteControl(StateInfo, ReadOnlyMemory.BYTE) [▶ 1043]	Changes the ADS status and the device status of an ADS server.
	WriteControl(StateInfo) [▶ 1044]	Changes the ADS status and the device status of an ADS server.
	WriteControl(StateInfo, ReadOnlyMemory.BYTE) [▶ 1045]	Changes the ADS status and the device status of an ADS server.
	WriteControlAsync(AdsState, UInt16, CancellationToken) [▶ 1046]	Changes the ADS status and device status of the ADS server asynchronously.
	WriteControlAsync(AdsState, UInt16, ReadOnlyMemory.BYTE, CancellationToken) [▶ 1047]	Changes the ADS status and device status of the ADS server asynchronously.






Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.42.1 IAdsStateControl Methods

Methods

	Name	Description
	TryWriteControl(StateInfo) [▶ 1042]	Changes the ADS status and the device status of an ADS server.

	Name	Description
	TryWriteControl(StateInfo, ReadOnlyMemory.Byte.) [▶ 1043]	Changes the ADS status and the device status of an ADS server.
	WriteControl(StateInfo) [▶ 1044]	Changes the ADS status and the device status of an ADS server.
	WriteControl(StateInfo, ReadOnlyMemory.Byte.) [▶ 1045]	Changes the ADS status and the device status of an ADS server.
	WriteControlAsync(AdsState, UInt16, CancellationToken) [▶ 1046]	Changes the ADS status and device status of the ADS server asynchronously.
	WriteControlAsync(AdsState, UInt16, ReadOnlyMemory.Byte., CancellationToken) [▶ 1047]	Changes the ADS status and device status of the ADS server asynchronously.



Reference

[IAdsStateControl Interface](#) [[▶ 1039](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.42.1.1 IAdsStateControl.TryWriteControl Method

Overload List

	Name	Description
	TryWriteControl(StateInfo) [▶ 1042]	Changes the ADS status and the device status of an ADS server.
	TryWriteControl(StateInfo, ReadOnlyMemory.Byte.) [▶ 1043]	Changes the ADS status and the device status of an ADS server.

Reference

[IAdsStateControl Interface](#) [[▶ 1039](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.42.1.1 IAdsStateControl.TryWriteControl Method (StateInfo)

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryWriteControl(  
    StateInfo stateInfo  
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200] New ADS status and device status.
-----------	---

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

AdsErrorCode.

Reference

[IAdsStateControl Interface](#) [[▶ 1039](#)]

[TryWriteControl Overload](#) [[▶ 1041](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.42.1.2 IAdsStateControl.TryWriteControl Method (StateInfo, ReadOnlyMemory`1)

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
AdsErrorCode TryWriteControl(  
    StateInfo stateInfo,  
    ReadOnlyMemory writeBuffer  
)
```

Parameters

stateInfo Type: [TwinCAT.Ads.StateInfo](#) [[▶ 1200](#)]

writeBuffer Type: [ReadOnlyMemory](#)

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

Reference

[IAdsStateControl Interface \[▸ 1039\]](#)

[TryWriteControl Overload \[▸ 1041\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.42.1.1.3 IAdsStateControl.TryWriteControl Method (StateInfo, ReadOnlyMemory.Byte.)

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```

AdsErrorCode TryWriteControl(
    StateInfo stateInfo,
    ReadOnlyMemory<byte> writeBuffer
)
    
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▸ 1200] New ADS status and device status.
writeBuffer	Type: System.ReadOnlyMemory.Byte. The write buffer.

Return Value

Type: [AdsErrorCode \[▸ 664\]](#)

AdsErrorCode.

Reference



[IAdsStateControl Interface \[▸ 1039\]](#)

[TryWriteControl Overload \[▸ 1041\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.42.1.2 IAdsStateControl.WriteControl Method

Overload List

	Name	Description
	WriteControl(StateInfo) [▸ 1044]	Changes the ADS status and the device status of an ADS server.
	WriteControl(StateInfo, ReadOnlyMemory.Byte.) [▸ 1045]	Changes the ADS status and the device status of an ADS server.

Reference

[IAdsStateControl Interface](#) [► 1039]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.42.1.2.1 IAdsStateControl.WriteControl Method (StateInfo)

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void WriteControl(  
    StateInfo stateInfo  
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [► 1200] New ADS status and device status.
-----------	---

Reference

[IAdsStateControl Interface](#) [► 1039]

[WriteControl Overload](#) [► 1043]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.42.1.2.2 IAdsStateControl.WriteControl Method (StateInfo, ReadOnlyMemory`1)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934 addedca3e72bc0ea15da1c14

Syntax

C#

```
void WriteControl(  
    StateInfo stateInfo,  
    ReadOnlyMemory writeBuffer  
)
```

Parameters

stateInfo Type: [TwinCAT.Ads.StateInfo](#) [► 1200]

writeBuffer Type: [ReadOnlyMemory](#)

Reference

[IAdsStateControl Interface](#) [► 1039]

[WriteControl Overload](#) [► 1043]

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.42.1.2.3 IAdsStateControl.WriteControl Method (StateInfo, ReadOnlyMemory.Byte.)

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void WriteControl(
    StateInfo stateInfo,
    ReadOnlyMemory<byte> writeBuffer
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200] New ADS status and device status.
writeBuffer	Type: System.ReadOnlyMemory.Byte. The write buffer.

Reference



[IAdsStateControl Interface \[▶ 1039\]](#)

[WriteControl Overload \[▶ 1043\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.42.1.3 IAdsStateControl.WriteControlAsync Method

Overload List

	Name	Description
	WriteControlAsync(AdsState, UInt16, CancellationToken) [▶ 1046]	Changes the ADS status and device status of the ADS server asynchronously.
	WriteControlAsync(AdsState, UInt16, ReadOnlyMemory.Byte, CancellationToken) [▶ 1047]	Changes the ADS status and device status of the ADS server asynchronously.

Reference

[IAdsStateControl Interface \[▶ 1039\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.42.1.3.1 IAdsStateControl.WriteControlAsync Method (AdsState, UInt16, CancellationToken)

Changes the ADS status and device status of the ADS server asynchronously.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultAds> WriteControlAsync (
    AdsState state,
    ushort deviceState,
    CancellationToken cancel
)
```

Parameters

state	Type: TwinCAT.Ads.AdsState [► 729] The ADS state.
deviceState	Type: System.UInt16 The device state.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds](#) [► 1116].

A task that represents the asynchronous 'WriteControl' operation. The [ResultAds](#) [► 1116] parameter contains the state the [ErrorCode](#) [► 1120] of the ADS communication after execution.

Reference

[IAdsStateControl Interface](#) [► 1039]

[WriteControlAsync Overload](#) [► 1045]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.42.1.3.2 IAdsStateControl.WriteControlAsync Method (AdsState, UInt16, ReadOnlyMemory<T>, Void)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
Task<ResultAds> WriteControlAsync (
    AdsState state,
    ushort deviceState,
    ReadOnlyMemory data,
    void cancel
)
```

Parameters

state Type: [TwinCAT.Ads.AdsState](#) [► 729]

deviceState Type: [System.UInt16](#)

data Type: [ReadOnlyMemory](#)

cancel Type: [System.Void](#)

Return Value

Type: [Task.ResultAds](#) [[▶ 1116](#)].

Reference

- [IAdsStateControl Interface](#) [[▶ 1039](#)]
- [WriteControlAsync Overload](#) [[▶ 1045](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.42.1.3.3 IAdsStateControl.WriteControlAsync Method (AdsState, UInt16, ReadOnlyMemory.Byte., CancellationToken)

Changes the ADS status and device status of the ADS server asynchronously.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultAds> WriteControlAsync (
    AdsState state,
    ushort deviceState,
    ReadOnlyMemory<byte> data,
    CancellationToken cancel
)
```

Parameters

state	Type: TwinCAT.Ads.AdsState [▶ 729] The ADS state.
deviceState	Type: System.UInt16 The device state.
data	Type: System.ReadOnlyMemory.Byte. Additional data.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds](#) [[▶ 1116](#)].

A task that represents the asynchronous 'WriteControl' operation. The [ResultAds](#) [[▶ 1116](#)] parameter contains the state the [ErrorCode](#) [[▶ 1120](#)] of the ADS communication after execution.

Reference

- [IAdsStateControl Interface](#) [[▶ 1039](#)]
- [WriteControlAsync Overload](#) [[▶ 1045](#)]

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.43 IAdsStateControlTimeout Interface

Interface IAdsStateControlTimeout

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229







Syntax

C#

```
public interface IAdsStateControlTimeout
```

The IAdsStateControlTimeout type exposes the following members.

Methods

	Name	Description
	ReadState [▸ 1049]	Reads the ADS status and the device status from an ADS server.
	TryReadState [▸ 1050]	Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful.
	TryWriteControl(StateInfo, Int32) [▸ 1051]	Changes the ADS status and the device status of an ADS server.
	TryWriteControl(StateInfo, ReadOnlyMemory<Byte>, Int32) [▸ 1052]	Changes the ADS status and the device status of an ADS server.
	WriteControl(StateInfo, Int32) [▸ 1053]	Changes the ADS status and the device status of an ADS server.
	WriteControl(StateInfo, ReadOnlyMemory<Byte>, Int32) [▸ 1054]	Changes the ADS status and the device status of an ADS server.

Remarks

For internal use only.







Reference

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.43.1 IAdsStateControlTimeout Methods

The [IAdsStateControlTimeout \[▸ 1048\]](#) type exposes the following members.

Methods

	Name	Description
	ReadState [▶ 1049]	Reads the ADS status and the device status from an ADS server.
	TryReadState [▶ 1050]	Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful.
	TryWriteControl(StateInfo, Int32) [▶ 1051]	Changes the ADS status and the device status of an ADS server.
	TryWriteControl(StateInfo, ReadOnlyMemory<Byte>, Int32) [▶ 1052]	Changes the ADS status and the device status of an ADS server.
	WriteControl(StateInfo, Int32) [▶ 1053]	Changes the ADS status and the device status of an ADS server.
	WriteControl(StateInfo, ReadOnlyMemory<Byte>, Int32) [▶ 1054]	Changes the ADS status and the device status of an ADS server.

Reference

[IAdsStateControlTimeout Interface](#) [[▶ 1048](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.43.1.1 IAdsStateControlTimeout.ReadState Method

Reads the ADS status and the device status from an ADS server.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
StateInfo ReadState(
    int timeout
)
```

Parameters

timeout	Type: System.Int32 The timeout.
---------	--

Return Value

Type: [StateInfo](#) [[▶ 1200](#)]

The ADS statue and device status.

Reference

[IAdsStateControlTimeout Interface](#) [[▶ 1048](#)]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.43.1.2 IAdsStateControlTimeout.TryReadState Method

Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryReadState(
    int timeout,
    out StateInfo stateInfo
)
```

Parameters

timeout	Type: System.Int32 The timeout.
stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200]. The ADS statue and device status.

Return Value

Type: [AdsErrorCode](#) [▶ 664]

AdsErrorCode of the ADS read state call. Check for AdsErrorCode.NoError to see if call was successful.



Reference

[IAdsStateControlTimeout Interface](#) [▶ 1048]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.43.1.3 IAdsStateControlTimeout.TryWriteControl Method

Overload List

	Name	Description
	TryWriteControl(StateInfo, Int32) [▶ 1051]	Changes the ADS status and the device status of an ADS server.
	TryWriteControl(StateInfo, ReadOnlyMemory<Byte>, Int32) [▶ 1052]	Changes the ADS status and the device status of an ADS server.

Reference

[IAdsStateControlTimeout Interface](#) [▶ 1048]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.43.1.3.1 IAdsStateControlTimeout.TryWriteControl Method (StateInfo, Int32)

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryWriteControl(  
    StateInfo stateInfo,  
    int timeout  
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200] New ADS status and device status.
timeout	Type: System.Int32 The timeout.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
AdsErrorCode.

Reference

[IAdsStateControlTimeout Interface](#) [[▶ 1048](#)]

[TryWriteControl Overload](#) [[▶ 1050](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.43.1.3.2 IAdsStateControlTimeout.TryWriteControl Method (StateInfo, ReadOnlyMemory<T>, Void)

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
AdsErrorCode TryWriteControl(  
    StateInfo stateInfo,  
    ReadOnlyMemory writeBuffer,  
    void timeout  
)
```

Parameters

stateInfo Type: [TwinCAT.Ads.StateInfo](#) [[▶ 1200](#)]

writeBuffer Type: [ReadOnlyMemory](#)

timeout Type: [System.Void](#)

Return ValueType: [AdsErrorCode](#) [▶ 664]**Reference**[IAdsStateControlTimeout Interface](#) [▶ 1048][TryWriteControl Overload](#) [▶ 1050][TwinCAT.Ads Namespace](#) [▶ 179]**6.2.43.1.3.3 IAdsStateControlTimeout.TryWriteControl Method (StateInfo, ReadOnlyMemory.Byte., Int32)**

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads](#) [▶ 179]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```

AdsErrorCode TryWriteControl(
    StateInfo stateInfo,
    ReadOnlyMemory<byte> writeBuffer,
    int timeout
)

```


Parameters


stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200] New ADS status and device status.
writeBuffer	Type: System.ReadOnlyMemory.Byte. The write buffer.
timeout	Type: System.Int32 The timeout.

Return ValueType: [AdsErrorCode](#) [▶ 664]

AdsErrorCode.

Reference[IAdsStateControlTimeout Interface](#) [▶ 1048][TryWriteControl Overload](#) [▶ 1050][TwinCAT.Ads Namespace](#) [▶ 179]**6.2.43.1.4 IAdsStateControlTimeout.WriteControl Method****Overload List**

	Name	Description
	WriteControl(StateInfo, Int32) [▶ 1053]	Changes the ADS status and the device status of an ADS server.

	Name	Description
	WriteControl(StateInfo, ReadOnlyMemory<Byte>, Int32) [▶ 1054]	Changes the ADS status and the device status of an ADS server.

Reference

[IAdsStateControlTimeout Interface](#) [[▶ 1048](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.43.1.4.1 IAdsStateControlTimeout.WriteControl Method (StateInfo, Int32)

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void WriteControl(
    StateInfo stateInfo,
    int timeout
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200] New ADS status and device status.
timeout	Type: System.Int32 The timeout.

Reference

[IAdsStateControlTimeout Interface](#) [[▶ 1048](#)]

[WriteControl Overload](#) [[▶ 1052](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.43.1.4.2 IAdsStateControlTimeout.WriteControl Method (StateInfo, ReadOnlyMemory<Byte>, Int32)

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
void WriteControl(
    StateInfo stateInfo,
    ReadOnlyMemory<Byte> writeBuffer,
    void timeout
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200]
writeBuffer	Type: ReadOnlyMemory
timeout	Type: System.Void

Reference

[IAdsStateControlTimeout Interface](#) [▶ 1048]

[WriteControl Overload](#) [▶ 1052]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.43.1.4.3 IAdsStateControlTimeout.WriteControl Method (StateInfo, ReadOnlyMemory.Byte., Int32)

Changes the ADS status and the device status of an ADS server.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax**C#**

```
void WriteControl(
    StateInfo stateInfo,
    ReadOnlyMemory<byte> writeBuffer,
    int timeout
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [▶ 1200] New ADS status and device status.
writeBuffer	Type: System.ReadOnlyMemory.Byte. The write buffer.
timeout	Type: System.Int32 The timeout.

Reference

[IAdsStateControlTimeout Interface](#) [▶ 1048]

[WriteControl Overload](#) [▶ 1052]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.44 IAdsStateObserver Interface

Interface for an AdsState observer

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229


Syntax

C#


```
public interface IAdsStateObserver
```

The IAdsStateObserver type exposes the following members.

Properties

	Name	Description
	StateInfo [▶ 1055]	Gets the current state of the connected ADS Server.

Events

	Name	Description
	AdsStateChanged [▶ 1056]	Occurs when the ads state has been changed.


Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.44.1 IAdsStateObserver Properties

The [IAdsStateObserver](#) [[▶ 1054](#)] type exposes the following members.

Properties

	Name	Description
	StateInfo [▶ 1055]	Gets the current state of the connected ADS Server.

Reference

[IAdsStateObserver Interface](#) [[▶ 1054](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.44.1.1 IAdsStateObserver.StateInfo Property

Gets the current state of the connected ADS Server.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
StateInfo StateInfo { get; }
```

Property Value

Type: [StateInfo](#) [[▶ 1200](#)]

ADS state

Reference


[IAdsStateObserver Interface](#) [[▶ 1054](#)]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.44.2 IAdsStateObserver Events

The [IAdsStateObserver](#) [▶ 1054] type exposes the following members.

Events

	Name	Description
	AdsStateChanged [▶ 1056]	Occurs when the ads state has been changed.

Reference

[IAdsStateObserver Interface](#) [▶ 1054]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.44.2.1 IAdsStateObserver.AdsStateChanged Event

Occurs when the ads state has been changed.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
event EventHandler<AdsStateChangedEventArgs2> AdsStateChanged
```

Value

Type: [System.EventHandler.AdsStateChangedEventArgs2](#) [▶ 732].

Reference

[IAdsStateObserver Interface](#) [▶ 1054]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.45 IAdsStateProvider Interface

Interface IAdsStateProvider

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax





C#

```
public interface IAdsStateProvider
```


The IAdsStateProvider type exposes the following members.

Methods

	Name	Description
	ReadState [▶ 1058]	Reads the ADS status and the device status from an ADS server.

	Name	Description
	ReadStateAsync [▶ 1058]	Reads the ADS status and the device status from an ADS server.
	RegisterAdsStateCh angedAsync [▶ 1059]	Registers for AdsStateChanged [▶ 1061] events as an asynchronous operation.
	TryReadState [▶ 1059]	Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful.
	UnregisterAdsState ChangedAsync [▶ 1060]	Registers for AdsStateChanged [▶ 1061] events as an asynchronous operation.

Events

	Name	Description
	AdsStateChanged [▶ 1061]	Occurs when the AdsState of the target system has been changed.






Reference

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.45.1 IAdsStateProvider Methods

The [IAdsStateProvider](#) [▶ 1056] type exposes the following members.

Methods

	Name	Description
	ReadState [▶ 1058]	Reads the ADS status and the device status from an ADS server.
	ReadStateAsync [▶ 1058]	Reads the ADS status and the device status from an ADS server.
	RegisterAdsStateCh angedAsync [▶ 1059]	Registers for AdsStateChanged [▶ 1061] events as an asynchronous operation.
	TryReadState [▶ 1059]	Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful.
	UnregisterAdsState ChangedAsync [▶ 1060]	Registers for AdsStateChanged [▶ 1061] events as an asynchronous operation.

Reference

[IAdsStateProvider Interface](#) [▶ 1056]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.45.1.1 IAdsStateProvider.ReadState Method

Reads the ADS status and the device status from an ADS server.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
StateInfo ReadState()
```

Return Value

Type: [StateInfo](#) [► 1200]

The ADS status and device status or an Exception with ErrorCode: [DeviceServiceNotSupported](#) [► 664].

Remarks

Not all ADS Servers support the State ADS Request.

Reference

[IAdsStateProvider Interface](#) [► 1056]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.45.1.2 IAdsStateProvider.ReadStateAsync Method

Reads the ADS status and the device status from an ADS server.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultReadDeviceState> ReadStateAsync (
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token
--------	--

Return Value

Type: [Task.ResultReadDeviceState](#) [► 1157].

A task that represents the asynchronous 'ReadState' operation. The [ResultReadDeviceState](#) [► 1157] parameter contains the state ([State](#) [► 1159]) as long as the [ErrorCode](#) [► 1120] of the ADS communication after execution.

Remarks

Not all ADS Servers support the State ADS Request

Reference

[IAdsStateProvider Interface](#) [► 1056]

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.45.1.3 IAdsStateProvider.RegisterAdsStateChangedAsync Method

Registers for [AdsStateChanged \[► 1061\]](#) events as an asynchronous operation.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultAds> RegisterAdsStateChangedAsync (
    EventHandler<AdsStateChangedEventArgs> handler,
    CancellationToken cancel
)
```

Parameters

handler	Type: System.EventHandler.AdsStateChangedEventArgs [► 729] . The handler function to be registered for AdsStateChanged calls.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds \[► 1116\]](#).

A task that represents the asynchronous 'RegisterAdsStateChanged' operation. The [ResultAds \[► 1116\]](#) parameter contains the state the [ErrorCode \[► 1120\]](#) of the ADS communication after execution.

Remarks

Not all ADS Servers support the State ADS Request.

Reference

[IAdsStateProvider Interface \[► 1056\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.45.1.4 IAdsStateProvider.TryReadState Method

Reads the ADS status and the device status from an ADS server. Unlike the ReadState method this method does not call an exception on failure. Instead an AdsErrorCode is returned. If the return value is equal to AdsErrorCode.NoError the call was successful.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryReadState (
    out StateInfo stateInfo
)
```

Parameters

stateInfo	Type: TwinCAT.Ads.StateInfo [► 1200] . The ADS statue and device status.
-----------	---

Return Value

Type: [AdsErrorCode](#) [▶ 664]

[AdsErrorCode](#) [▶ 664] of the ADS read state call. Check for [NoError](#) [▶ 664] to see if call was successful.

Remarks

Not all ADS Servers support the State ADS Request

Reference

[IAdsStateProvider Interface](#) [▶ 1056]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.45.1.5 IAdsStateProvider.UnregisterAdsStateChangedAsync Method

Registers for [AdsStateChanged](#) [▶ 1061] events as an asynchronous operation.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultAds> UnregisterAdsStateChangedAsync (
    EventHandler<AdsStateChangedEventArgs> handler,
    CancellationToken cancel
)
```

Parameters

handler	Type: System.EventHandler.AdsStateChangedEventArgs [▶ 729]. The handler function to be unregistered.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds](#) [▶ 1116].

A task that represents the asynchronous 'UnregisterAdsStateChanged' operation. The [ResultAds](#) [▶ 1116] parameter contains the state the [ErrorCode](#) [▶ 1120] of the ADS communication after execution.

Reference


[IAdsStateProvider Interface](#) [▶ 1056]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.45.2 IAdsStateProvider Events

The [IAdsStateProvider](#) [▶ 1056] type exposes the following members.

Events

	Name	Description
	AdsStateChanged [▶ 1061]	Occurs when the AdsState of the target system has been changed.

Reference

[IAdsStateProvider Interface \[▶ 1056\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.45.2.1 IAdsStateProvider.AdsStateChanged Event

Occurs when the AdsState of the target system has been changed.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
event EventHandler<AdsStateChangedEventArgs> AdsStateChanged
```

Value

Type: [System.EventHandler.AdsStateChangedEventArgs \[▶ 729\]](#).

Remarks

Not all ADS Servers support the State ADS Request. This event occurs asynchronously in an Background Thread. Be aware about Synchronization contexts for possible synchronizations into the UI Synchronization context.

Reference

[IAdsStateProvider Interface \[▶ 1056\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.46 IAdsSymbolChangedProvider Interface

Interface IAdsSymbolChangedProvider

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax



C#

```
public interface IAdsSymbolChangedProvider
```


The IAdsSymbolChangedProvider type exposes the following members.

Methods

	Name	Description
	RegisterSymbolVersionChanged [▶ 1064]	Registers the symbol version changed.
	RegisterSymbolVersionChangedAsync [▶ 1062]	Registers the symbol version changed asynchronously.

	Name	Description
	UnregisterSymbolVersionChanged [▶ 1064]	Unregisters the symbol version changed.
	UnregisterSymbolVersionChangedAsync [▶ 1063]	Unregisters the symbol version changed asynchronous.

Events

	Name	Description
	AdsSymbolVersionChanged [▶ 1065]	Occurs when the ADS Symbol Version changed.





Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.46.1 IAdsSymbolChangedProvider Methods

The [IAdsSymbolChangedProvider](#) [[▶ 1061](#)] type exposes the following members.

Methods

	Name	Description
	RegisterSymbolVersionChanged [▶ 1064]	Registers the symbol version changed.
	RegisterSymbolVersionChangedAsync [▶ 1062]	Registers the symbol version changed asynchronously.
	UnregisterSymbolVersionChanged [▶ 1064]	Unregisters the symbol version changed.
	UnregisterSymbolVersionChangedAsync [▶ 1063]	Unregisters the symbol version changed asynchronous.

Reference

[IAdsSymbolChangedProvider Interface](#) [[▶ 1061](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.46.1.1 IAdsSymbolChangedProvider.RegisterSymbolVersionChangedAsync Method

Registers the symbol version changed asynchronously.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultAds> RegisterSymbolVersionChangedAsync (
    EventHandler<AdsSymbolVersionChangedEventArgs> handler,
    CancellationToken cancel
)
```

Parameters

handler	Type: System.EventHandler.AdsSymbolVersionChangedEventArgs [▶ 739] . The handler function to register.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds \[▶ 1116\]](#).

A task that represents the asynchronous 'RegisterSymbolVersionChanged' operation. The [ResultAds \[▶ 1116\]](#) parameter contains the value [ErrorCode \[▶ 1120\]](#) of the ADS communication after execution.

Reference

[IAdsSymbolChangedProvider Interface \[▶ 1061\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.46.1.2 IAdsSymbolChangedProvider.UnregisterSymbolVersionChangedAsync Method

Unregisters the symbol version changed asynchronous.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultAds> UnregisterSymbolVersionChangedAsync (
    EventHandler<AdsSymbolVersionChangedEventArgs> handler,
    CancellationToken cancel
)
```

Parameters

handler	Type: System.EventHandler.AdsSymbolVersionChangedEventArgs [▶ 739] . The handler function to unregister.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds \[▶ 1116\]](#).

A task that represents the asynchronous 'UnregisterSymbolVersionChangedAsync' operation. The [ResultAds \[▶ 1116\]](#) parameter contains the value [ErrorCode \[▶ 1120\]](#) of the ADS communication after execution.

Reference[IAdsSymbolChangedProvider Interface \[► 1061\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.46.1.3 IAdsSymbolChangedProvider.RegisterSymbolVersionChanged Method**

Registers the symbol version changed.

Namespace: [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
AdsErrorCode RegisterSymbolVersionChanged (
    EventHandler<AdsSymbolVersionChangedEventArgs> handler
)
```

Parameters

handler	Type: System.EventHandler.AdsSymbolVersionChangedEventArgs [► 739] . The handler function to register.
---------	---

Return ValueType: [AdsErrorCode \[► 664\]](#)

AdsErrorCode.

Reference[IAdsSymbolChangedProvider Interface \[► 1061\]](#)[TwinCAT.Ads Namespace \[► 179\]](#)**6.2.46.1.4 IAdsSymbolChangedProvider.UnregisterSymbolVersionChanged Method**

Unregisters the symbol version changed.

Namespace: [TwinCAT.Ads \[► 179\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
AdsErrorCode UnregisterSymbolVersionChanged (
    EventHandler<AdsSymbolVersionChangedEventArgs> handler
)
```

Parameters

handler	Type: System.EventHandler.AdsSymbolVersionChangedEventArgs [► 739] . The handler function to unregister.
---------	---

Return Value

Type: [AdsErrorCode](#) [▶ 664]
 AdsErrorCode.

Reference


[IAdsSymbolChangedProvider Interface](#) [▶ 1061]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.46.2 IAdsSymbolChangedProvider Events

The [IAdsSymbolChangedProvider](#) [▶ 1061] type exposes the following members.

Events

	Name	Description
	AdsSymbolVersionC hanged [▶ 1065]	Occurs when the ADS Symbol Version changed.

Reference

[IAdsSymbolChangedProvider Interface](#) [▶ 1061]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.46.2.1 IAdsSymbolChangedProvider.AdsSymbolVersionChanged Event

Occurs when the ADS Symbol Version changed.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
event EventHandler<AdsSymbolVersionChangedEventArgs> AdsSymbolVersionChanged
```

Value

Type: [System.EventHandler.AdsSymbolVersionChangedEventArgs](#) [▶ 739].

Reference

[IAdsSymbolChangedProvider Interface](#) [▶ 1061]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.47 IAdsSymbolicAccess Interface

Interface for symbolic ads access.

Namespace: [TwinCAT.Ads](#) [▶ 179]
















Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229
















Syntax**C#**

```
public interface IAdsSymbolicAccess
```

The IAdsSymbolicAccess type exposes the following members.

Methods

	Name	Description
	CleanupSymbolTable [▶ 1070]	Clears the internal symbol / DataTypes cache.
	ReadDataType [▶ 1070]	Call this method to obtain information about the specified data type.
	ReadDataTypeAsync [▶ 1070]	Call this method to obtain information about the specified data type.
	ReadSymbol [▶ 1071]	Call this method to obtain information about the individual symbols (variables) in ADS devices.
	ReadSymbolAsync [▶ 1072]	Call this method to obtain information about the individual symbols (variables) in ADS devices.
	ReadValue(ISymbol) [▶ 1073]	Reads the value of a symbol and returns it as an object.
	ReadValue(String, Type) [▶ 1075]	Reads the value of a symbol and returns the value as object. The parameter type must have the same binary layout as the ADS symbol.
	ReadValue.T.(ISymbol) [▶ 1074]	Reads the value of a symbol and returns it as an object.
	ReadValue.T.(String) [▶ 1073]	Reads the value of a symbol and returns the value. The parameter type must have the same binary layout as the ADS symbol.
	ReadValueAsync(ISymbol, CancellationToken) [▶ 1077]	Reads the value of a symbol asynchronously and returns it as an object.
	ReadValueAsync(String, Type, CancellationToken) [▶ 1079]	Reads the value of a symbol asynchronously.
	ReadValueAsync.T.(ISymbol, CancellationToken) [▶ 1078]	Reads the value of a symbol asynchronously and returns it as an object.
	ReadValueAsync.T.(String, CancellationToken) [▶ 1076]	Reads the value of a symbol asynchronously.
	TryReadDataType [▶ 1079]	Call this method to obtain information about the specified data type.
	TryReadSymbol [▶ 1080]	Call this method to obtain information about the individual symbols (variables) in ADS devices.

	Name	Description
	TryReadValue(ISymbol, Object.) [▶ 1082]	Reads the value of a symbol and returns it as an object.
	TryReadValue(String, Type, Object.) [▶ 1083]	Reads the value of a symbol and returns the value as object.
	TryReadValue.T. (ISymbol, T.) [▶ 1083]	Reads the value of a symbol and returns it as an object.
	TryReadValue.T. (String, T.) [▶ 1081]	Reads the value of a symbol and returns the value as object.
	TryWriteValue(ISymbol, Object) [▶ 1086]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	TryWriteValue(String, Object) [▶ 1085]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	TryWriteValue.T. (ISymbol, T) [▶ 1087]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	TryWriteValue.T. (String, T) [▶ 1085]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol.
	WriteValue(ISymbol, Object) [▶ 1089]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValue(String, Object) [▶ 1088]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol.
	WriteValue.T. (ISymbol, T) [▶ 1090]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValue.T.(String, T) [▶ 1089]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol.
	WriteValueAsync(ISymbol, Object, CancellationToken) [▶ 1092]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValueAsync.T. (ISymbol, T, CancellationToken) [▶ 1093]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValueAsync.T. (String, T, CancellationToken) [▶ 1091]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol.

















Reference















[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.47.1 IAdsSymbolicAccess Methods

The `IAdsSymbolicAccess` [▶ 1065] type exposes the following members.

Methods

	Name	Description
	<code>CleanupSymbolTable</code> [▶ 1070]	Clears the internal symbol / DataTypes cache.
	<code>ReadDataType</code> [▶ 1070]	Call this method to obtain information about the specified data type.
	<code>ReadDataTypeAsync</code> [▶ 1070]	Call this method to obtain information about the specified data type.
	<code>ReadSymbol</code> [▶ 1071]	Call this method to obtain information about the individual symbols (variables) in ADS devices.
	<code>ReadSymbolAsync</code> [▶ 1072]	Call this method to obtain information about the individual symbols (variables) in ADS devices.
	<code>ReadValue(ISymbol)</code> [▶ 1073]	Reads the value of a symbol and returns it as an object.
	<code>ReadValue(String, Type)</code> [▶ 1075]	Reads the value of a symbol and returns the value as object. The parameter type must have the same binary layout as the ADS symbol.
	<code>ReadValue.T.(ISymbol)</code> [▶ 1074]	Reads the value of a symbol and returns it as an object.
	<code>ReadValue.T.(String)</code> [▶ 1073]	Reads the value of a symbol and returns the value. The parameter type must have the same binary layout as the ADS symbol.
	<code>ReadValueAsync(ISymbol, CancellationToken)</code> [▶ 1077]	Reads the value of a symbol asynchronously and returns it as an object.
	<code>ReadValueAsync(String, Type, CancellationToken)</code> [▶ 1079]	Reads the value of a symbol asynchronously.
	<code>ReadValueAsync.T.(ISymbol, CancellationToken)</code> [▶ 1078]	Reads the value of a symbol asynchronously and returns it as an object.
	<code>ReadValueAsync.T.(String, CancellationToken)</code> [▶ 1076]	Reads the value of a symbol asynchronously.
	<code>TryReadDataType</code> [▶ 1079]	Call this method to obtain information about the specified data type.
	<code>TryReadSymbol</code> [▶ 1080]	Call this method to obtain information about the individual symbols (variables) in ADS devices.
	<code>TryReadValue(ISymbol, Object.)</code> [▶ 1082]	Reads the value of a symbol and returns it as an object.

	Name	Description
	TryReadValue(String, Type, Object.) [▶ 1083]	Reads the value of a symbol and returns the value as object.
	TryReadValue.T. (ISymbol, T.) [▶ 1083]	Reads the value of a symbol and returns it as an object.
	TryReadValue.T. (String, T.) [▶ 1081]	Reads the value of a symbol and returns the value as object.
	TryWriteValue(ISymbol, Object) [▶ 1086]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	TryWriteValue(String, Object) [▶ 1085]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	TryWriteValue.T. (ISymbol, T) [▶ 1087]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	TryWriteValue.T. (String, T) [▶ 1085]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol.
	WriteValue(ISymbol, Object) [▶ 1089]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValue(String, Object) [▶ 1088]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol.
	WriteValue.T. (ISymbol, T) [▶ 1090]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValue.T.(String, T) [▶ 1089]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol.
	WriteValueAsync(ISymbol, Object, CancellationToken) [▶ 1092]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValueAsync.T. (ISymbol, T, CancellationToken) [▶ 1093]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValueAsync.T. (String, T, CancellationToken) [▶ 1091]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol.

Reference

[IAdsSymbolicAccess Interface](#) [▶ 1065]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.47.1.1 IAdsSymbolicAccess.CleanupSymbolTable Method

Clears the internal symbol / DataTypes cache.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void CleanupSymbolTable()
```

Remarks

Previously stored symbol information is cleared. As a consequence the symbol information must be obtained from the ADS server again if accessed, which needs an extra ADS round trip.

Reference

[IAdsSymbolicAccess Interface](#) [► 1065]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.47.1.2 IAdsSymbolicAccess.ReadDataType Method

Call this method to obtain information about the specified data type.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDataType ReadDataType(  
    string typeName  
)
```

Parameters

typeName	Type: System.String Name of the data type (without namespace)
----------	--

Return Value

Type: [IDataType](#) [► 2475]

An containing the requested type.

Reference

[IAdsSymbolicAccess Interface](#) [► 1065]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsSymbolicAccess.TryReadDataType\(String, IDataType.\)](#) [► 1079]

[IAdsSymbolicAccess.ReadDataTypeAsync\(String, CancellationToken\)](#) [► 1070]

6.2.47.1.3 IAdsSymbolicAccess.ReadDataTypeAsync Method

Call this method to obtain information about the specified data type.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultValue<IDataType>> ReadDataTypeAsync (
    string typeName,
    CancellationToken cancel
)
```

Parameters

typeName	Type: System.String Name of the data type.
cancel	Type: System.Threading.CancellationToken The cancel token.

Return Value

Type: [Task.ResultValue \[▸ 1181\].IDataType \[▸ 2475\]](#)..

A task that represents the asynchronous 'ReadDataType' operation. The [ResultValue.TValue. \[▸ 1181\]](#) parameter contains the read value ([Value \[▸ 1185\]](#)) and the [ErrorCode \[▸ 1120\]](#) after execution.

Reference

[IAdsSymbolicAccess Interface \[▸ 1065\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

[IAdsSymbolicAccess.ReadDataType\(String\) \[▸ 1070\]](#)

[IAdsSymbolicAccess.TryReadDataType\(String, IDataType.\) \[▸ 1079\]](#)

6.2.47.1.4 IAdsSymbolicAccess.ReadSymbol Method

Call this method to obtain information about the individual symbols (variables) in ADS devices.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IAdsSymbol ReadSymbol (
    string symbolPath
)
```

Parameters

symbolPath	Type: System.String Name of the symbol.
------------	--

Return Value

Type: [IAdsSymbol \[▸ 1755\]](#)

An [IAdsSymbol \[▸ 1755\]](#) containing the requested symbol information or null if symbol could not be found.

Reference

[IAdsSymbolicAccess Interface](#) [► 1065]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsSymbolicAccess.TryReadSymbol\(String, IAdsSymbol.\)](#) [► 1080]

[IAdsSymbolicAccess.ReadSymbolAsync\(String, CancellationToken\)](#) [► 1072]

6.2.47.1.5 IAdsSymbolicAccess.ReadSymbolAsync Method

Call this method to obtain information about the individual symbols (variables) in ADS devices.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultValue<IAdsSymbol>> ReadSymbolAsync (
    string symbolPath,
    CancellationToken cancel
)
```

Parameters

symbolPath	Type: System.String Name of the symbol.
cancel	Type: System.Threading.CancellationToken The cancel token.

Return Value

Type: [Task.ResultValue](#) [► 1181].[IAdsSymbol](#) [► 1755].

A task that represents the asynchronous 'ReadSymbolInfo' operation. The [ResultValue.TValue](#). [► 1181] parameter contains the read value ([Value](#) [► 1185]) and the [ErrorCode](#) [► 1120] after execution.

Reference

[IAdsSymbolicAccess Interface](#) [► 1065]




[TwinCAT.Ads Namespace](#) [► 179]


[IAdsSymbolicAccess.ReadSymbol\(String\)](#) [► 1071]

[IAdsSymbolicAccess.TryReadSymbol\(String, IAdsSymbol.\)](#) [► 1080]

6.2.47.1.6 IAdsSymbolicAccess.ReadValue Method

Overload List

	Name	Description
	ReadValue(IAdsSymbol) [► 1073]	Reads the value of a symbol and returns it as an object.
	ReadValue.T. (IAdsSymbol) [► 1074]	Reads the value of a symbol and returns it as an object.
	ReadValue.T.(String) [► 1073]	Reads the value of a symbol and returns the value. The parameter type must have the same binary layout as the ADS symbol.

	Name	Description
	ReadValue(String, Type) [▸ 1075]	Reads the value of a symbol and returns the value as object. The parameter type must have the same binary layout as the ADS symbol.

Reference

[IAdsSymbolicAccess Interface](#) [[▸ 1065](#)]

[TwinCAT.Ads Namespace](#) [[▸ 179](#)]

6.2.47.1.6.1 IAdsSymbolicAccess.ReadValue.T. Method (String)

Reads the value of a symbol and returns the value. The parameter type must have the same binary layout as the ADS symbol.

Namespace: [TwinCAT.Ads](#) [[▸ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
T ReadValue<T>(
    string symbolPath
)
```

Parameters

symbolPath	Type: System.String Name of the ADS symbol.
------------	--

Type Parameters

T	The value type
---	----------------

Return Value

Type: T
Value of the symbol

Reference

[IAdsSymbolicAccess Interface](#) [[▸ 1065](#)]

[ReadValue Overload](#) [[▸ 1072](#)]

[TwinCAT.Ads Namespace](#) [[▸ 179](#)]

[IAdsSymbolicAccess.TryReadValue.T.\(String, T.\)](#) [[▸ 1081](#)]

[IAdsSymbolicAccess.ReadValueAsync.T.\(String, CancellationToken\)](#) [[▸ 1076](#)]

6.2.47.1.6.2 IAdsSymbolicAccess.ReadValue Method (ISymbol)

Reads the value of a symbol and returns it as an object.

Namespace: [TwinCAT.Ads](#) [[▸ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object ReadValue(
    ISymbol symbol
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol that should be read.
--------	--

Return Value

Type: [Object](#)

The value of the symbol as an object.

Remarks

Supported types are limited to 'AnyTypes', what includes all primitive types (UInt32, Int32, Bool etc.), strings, and Arrays that are compositions of 'AnyTypes'.

Reference

[IAdsSymbolicAccess Interface](#) [[▶ 1065](#)]

[ReadValue Overload](#) [[▶ 1072](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsSymbolicAccess.TryReadValue\(ISymbol, Object.\)](#) [[▶ 1082](#)]

[IAdsSymbolicAccess.ReadValueAsync\(ISymbol, CancellationToken\)](#) [[▶ 1077](#)]

6.2.47.1.6.3 IAdsSymbolicAccess.ReadValue.T. Method (ISymbol)

Reads the value of a symbol and returns it as an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
T ReadValue<T>(
    ISymbol symbol
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol that should be read.
--------	--

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: T

The value of the symbol.

Remarks

Supported types are limited to 'AnyTypes', what includes all primitive types (UInt32, Int32, Bool etc.), strings, and Arrays that are compositions of 'AnyTypes'.

Reference

[IAdsSymbolicAccess Interface](#) [► 1065]

[ReadValue Overload](#) [► 1072]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsSymbolicAccess.TryReadValue.T.\(ISymbol, T.\)](#) [► 1083]

[IAdsSymbolicAccess.ReadValueAsync.T.\(ISymbol, CancellationToken\)](#) [► 1078]

6.2.47.1.6.4 IAdsSymbolicAccess.ReadValue Method (String, Type)

Reads the value of a symbol and returns the value as object. The parameter type must have the same binary layout as the ADS symbol.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
Object ReadValue(
    string symbolPath,
    Type type
)
```

Parameters

symbolPath	Type: System.String Symbol Path of the ADS symbol.
type	Type: System.Type Managed type of the ADS symbol.

Return Value

Type: [Object](#)
Value of the symbol

Reference

[IAdsSymbolicAccess Interface](#) [► 1065]

[ReadValue Overload](#) [► 1072]





[TwinCAT.Ads Namespace](#) [► 179]

[IAdsSymbolicAccess.TryReadValue\(String, Type, Object.\)](#) [► 1083]

[IAdsSymbolicAccess.ReadValueAsync\(String, Type, CancellationToken\)](#) [► 1079]

6.2.47.1.7 IAdsSymbolicAccess.ReadValueAsync Method

Overload List

	Name	Description
	ReadValueAsync(ISymbol, CancellationToken) [▶ 1077]	Reads the value of a symbol asynchronously and returns it as an object.
	ReadValueAsync.T.(ISymbol, CancellationToken) [▶ 1078]	Reads the value of a symbol asynchronously and returns it as an object.
	ReadValueAsync.T.(String, CancellationToken) [▶ 1076]	Reads the value of a symbol asynchronously.
	ReadValueAsync(String, Type, CancellationToken) [▶ 1079]	Reads the value of a symbol asynchronously.

Reference

[IAdsSymbolicAccess Interface](#) [▶ 1065]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.47.1.7.1 IAdsSymbolicAccess.ReadValueAsync.T. Method (String, CancellationToken)

Reads the value of a symbol asynchronously.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultValue<T>> ReadValueAsync<T>(
    string symbolPath,
    CancellationToken cancel
)
```

Parameters

symbolPath	Type: System.String Name of the ADS symbol.
cancel	Type: System.Threading.CancellationToken The cancel token.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [Task.ResultValue](#) [[▶ 1181](#)].T..

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [[▶ 1129](#)] parameter contains the read value ([Value](#) [[▶ 1185](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Remarks

The parameter type must have the same binary layout as the ADS symbol.

Reference

[IAdsSymbolicAccess Interface](#) [[▶ 1065](#)]

[ReadValueAsync Overload](#) [[▶ 1076](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsSymbolicAccess.ReadValue.T.\(String\)](#) [[▶ 1073](#)]

[IAdsSymbolicAccess.TryReadValue.T.\(String, T.\)](#) [[▶ 1081](#)]

6.2.47.1.7.2 IAdsSymbolicAccess.ReadValueAsync Method (ISymbol, Cancellation token)

Reads the value of a symbol asynchronously and returns it as an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultAnyValue> ReadValueAsync (
    ISymbol symbol,
    Cancellation token cancel
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol that should be read.
cancel	Type: System.Threading.Cancellation token The cancel token.

Return Value

Type: [Task.ResultAnyValue](#) [[▶ 1129](#)].

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [[▶ 1129](#)] parameter contains the read value ([Value](#) [[▶ 1185](#)]) and the [ErrorCode](#) [[▶ 1120](#)] after execution.

Remarks

Supported types are limited to 'AnyTypes', what includes all primitive types (UInt32, Int32, Bool etc.), strings, and Arrays that are compositions of 'AnyTypes'..

Reference

[IAdsSymbolicAccess Interface](#) [[▶ 1065](#)]

[ReadValueAsync Overload](#) [[▶ 1076](#)]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsSymbolicAccess.ReadValue\(ISymbol\)](#) [► 1073]

[IAdsSymbolicAccess.ReadValue\(ISymbol\)](#) [► 1073]

[IAdsSymbolicAccess.TryReadValue\(ISymbol, Object.\)](#) [► 1082]

Also see about this

 [ResultAnyValue.Value Property](#) [► 1131]

6.2.47.1.7.3 IAdsSymbolicAccess.ReadValueAsync.T. Method (ISymbol, CancellationToken)

Reads the value of a symbol asynchronously and returns it as an object.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultValue<T>> ReadValueAsync<T>(
    ISymbol symbol,
    CancellationToken cancel
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol that should be read.
cancel	Type: System.Threading.CancellationToken The cancel token.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [Task.ResultValue](#) [► 1181].T..

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [► 1129] parameter contains the read value ([Value](#) [► 1185]) and the [ErrorCode](#) [► 1120] after execution.

Remarks

Supported types are limited to 'AnyTypes', what includes all primitive types (UInt32, Int32, Bool etc.), strings, and Arrays that are compositions of 'AnyTypes'.

Reference

[IAdsSymbolicAccess Interface](#) [► 1065]

[ReadValueAsync Overload](#) [► 1076]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsSymbolicAccess.ReadValue.T.\(ISymbol\)](#) [► 1074]

[IAdsSymbolicAccess.TryReadValue.T.\(ISymbol, T.\)](#) [► 1083]

6.2.47.1.7.4 IAdsSymbolicAccess.ReadValueAsync Method (String, Type, CancellationToken)

Reads the value of a symbol asynchronously.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultAnyValue> ReadValueAsync (
    string symbolPath,
    Type type,
    CancellationToken cancel
)
```

Parameters

symbolPath	Type: System.String Name of the ADS symbol.
type	Type: System.Type Managed type of the ADS symbol
cancel	Type: System.Threading.CancellationToken The cancel token.

Return Value

Type: [Task.ResultAnyValue](#) [► 1129].

A task that represents the asynchronous read operation. The [ResultAnyValue](#) [► 1129] parameter contains the read value ([Value](#) [► 1185]) and the [ErrorCode](#) [► 1120] after execution.

Remarks

The parameter type must have the same binary layout as the ADS symbol.

Reference

[IAdsSymbolicAccess Interface](#) [► 1065]

[ReadValueAsync Overload](#) [► 1076]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsSymbolicAccess.ReadValue\(String, Type\)](#) [► 1075]

[IAdsSymbolicAccess.TryReadValue\(String, Type, Object.\)](#) [► 1083]

Also see about this

 [ResultAnyValue.Value Property](#) [► 1131]

6.2.47.1.8 IAdsSymbolicAccess.TryReadDataType Method

Call this method to obtain information about the specified data type.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```

AdsErrorCode TryReadDataType(
    string typeName,
    out IDataTypeInfo? dataType
)

```

Parameters

typeName	Type: System.String Name of the symbol.
dataType	Type: TwinCAT.TypeSystem.IDataTypeInfo [▶ 2475]. The symbol.

Return Value

Type: [AdsErrorCode](#) [▶ 664]

A [IDataTypeInfo](#) [▶ 2475] containing the requested symbol information or null if symbol could not be found.

Reference

[IAdsSymbolicAccess Interface](#) [▶ 1065]

[TwinCAT.Ads Namespace](#) [▶ 179]

[IAdsSymbolicAccess.ReadDataType\(String\)](#) [▶ 1070]

6.2.47.1.9 IAdsSymbolicAccess.TryReadSymbol Method

Call this method to obtain information about the individual symbols (variables) in ADS devices.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```

AdsErrorCode TryReadSymbol(
    string symbolPath,
    out IAdsSymbolInfo? symbol
)

```

Parameters

symbolPath	Type: System.String Name of the symbol.
symbol	Type: TwinCAT.Ads.TypeSystem.IAdsSymbolInfo [▶ 1755]. The symbol.

Return Value

Type: [AdsErrorCode](#) [▶ 664]

An [IAdsSymbolInfo](#) [▶ 1755] containing the requested symbol information or null if symbol could not be found.

Reference





[IAdsSymbolicAccess Interface](#) [▶ 1065]

[TwinCAT.Ads Namespace](#) [▶ 179]

[IAdsSymbolicAccess.ReadSymbol\(String\) \[► 1071\]](#)

6.2.47.1.10 IAdsSymbolicAccess.TryReadValue Method

Overload List

	Name	Description
	TryReadValue(ISymbol, Object.) [► 1082]	Reads the value of a symbol and returns it as an object.
	TryReadValue.T.(ISymbol, T.) [► 1083]	Reads the value of a symbol and returns it as an object.
	TryReadValue.T.(String, T.) [► 1081]	Reads the value of a symbol and returns the value as object.
	TryReadValue(String, Type, Object.) [► 1083]	Reads the value of a symbol and returns the value as object.

Reference

[IAdsSymbolicAccess Interface \[► 1065\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.47.1.10.1 IAdsSymbolicAccess.TryReadValue.T. Method (String, T.)

Reads the value of a symbol and returns the value as object.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryReadValue<T>(
    string symbolPath,
    out T value
)
```

Parameters

symbolPath	Type: System.String Name of the ADS symbol.
value	Type: T. The read value of the Symbol.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [AdsErrorCode \[► 664\]](#)

The [AdsErrorCode \[► 664\]](#).

Remarks

The parameter type must have the same binary layout as the ADS symbol.

Reference

[IAdsSymbolicAccess Interface](#) [► 1065]

[TryReadValue Overload](#) [► 1081]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsSymbolicAccess.ReadValue.T.\(String\)](#) [► 1073]

[IAdsSymbolicAccess.ReadValueAsync.T.\(String, CancellationToken\)](#) [► 1076]

6.2.47.1.10.2 IAdsSymbolicAccess.TryReadValue Method (ISymbol, Object.)

Reads the value of a symbol and returns it as an object.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
AdsErrorCode TryReadValue(
    ISymbol symbol,
    out Object?? value
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol that should be read.
value	Type: System.Object . The value.

Return Value

Type: [AdsErrorCode](#) [► 664]

The ADS Error Code

Remarks

Supported types are limited to 'AnyTypes', what includes all primitive types (UInt32, Int32, Bool etc.), strings, and Arrays that are compositions of 'AnyTypes'.

Reference

[IAdsSymbolicAccess Interface](#) [► 1065]

[TryReadValue Overload](#) [► 1081]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsSymbolicAccess.ReadValue\(ISymbol\)](#) [► 1073]

[IAdsSymbolicAccess.ReadValueAsync\(ISymbol, CancellationToken\)](#) [► 1077]

6.2.47.1.10.3 IAdsSymbolicAccess.TryReadValue.T. Method (ISymbol, T.)

Reads the value of a symbol and returns it as an object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryReadValue<T>(
    ISymbol symbol,
    out T value
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol that should be read.
value	Type: T. The value.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The ADS Error Code

Remarks

Supported types are limited to 'AnyTypes', what includes all primitive types (UInt32, Int32, Bool etc.), strings, and Arrays that are compositions of 'AnyTypes'.

Reference

[IAdsSymbolicAccess Interface](#) [[▶ 1065](#)]

[TryReadValue Overload](#) [[▶ 1081](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsSymbolicAccess.ReadValue.T.\(ISymbol\)](#) [[▶ 1074](#)]

[IAdsSymbolicAccess.ReadValueAsync\(ISymbol, CancellationToken\)](#) [[▶ 1077](#)]

6.2.47.1.10.4 IAdsSymbolicAccess.TryReadValue Method (String, Type, Object.)

Reads the value of a symbol and returns the value as object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```

AdsErrorCode TryReadValue(
    string symbolPath,
    Type type,
    out Object?? value
)

```

Parameters

symbolPath	Type: System.String Name of the ADS symbol.
type	Type: System.Type Managed type of the ADS symbol.
value	Type: System.Object . The read value of the Symbol.

Return Value

Type: [AdsErrorCode](#) [▶ 664]

The [AdsErrorCode](#) [▶ 664].

Remarks

The parameter type must have the same binary layout as the ADS symbol.

Reference

[IAdsSymbolicAccess Interface](#) [▶ 1065]

[TryReadValue Overload](#) [▶ 1081]

[TwinCAT.Ads Namespace](#) [▶ 179]





[IAdsSymbolicAccess.ReadValue\(String, Type\)](#) [▶ 1075]

[IAdsSymbolicAccess.ReadValueAsync.T.\(String, CancellationToken\)](#) [▶ 1076]

[IAdsSymbolicAccess.ReadValueAsync\(String, Type, CancellationToken\)](#) [▶ 1079]

6.2.47.1.11 IAdsSymbolicAccess.TryWriteValue Method

Overload List

	Name	Description
	TryWriteValue(ISymbol, Object) [▶ 1086]	Writes a value to the symbol. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	TryWriteValue.T.(ISymbol, T) [▶ 1087]	Writes a value to the symbol. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	TryWriteValue(String, Object) [▶ 1085]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.
	TryWriteValue.T.(String, T) [▶ 1085]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol.

Reference[IAdsSymbolicAccess Interface](#) [► 1065][TwinCAT.Ads Namespace](#) [► 179]**6.2.47.1.11.1 IAdsSymbolicAccess.TryWriteValue Method (String, Object)**

Writes the passed object value to the specified ADS symbol. The parameter type must have the same layout as the ADS symbol.

Namespace: [TwinCAT.Ads](#) [► 179]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
AdsErrorCode TryWriteValue(
    string symbolPath,
    Object value
)
```

Parameters

symbolPath	Type: System.String Name of the ADS symbol.
value	Type: System.Object Object holding the value to be written to the ADS symbol

Return ValueType: [AdsErrorCode](#) [► 664]

AdsErrorCode.

Reference[IAdsSymbolicAccess Interface](#) [► 1065][TryWriteValue Overload](#) [► 1084][TwinCAT.Ads Namespace](#) [► 179][IAdsSymbolicAccess.WriteValue\(String, Object\)](#) [► 1088][IAdsSymbolicAccess.WriteValueAsync.T.\(String, T, CancellationToken\)](#) [► 1091]**6.2.47.1.11.2 IAdsSymbolicAccess.TryWriteValue.T. Method (String, T)**

Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol.

Namespace: [TwinCAT.Ads](#) [► 179]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
AdsErrorCode TryWriteValue<T>(
    string symbolPath,
    T value
)
```

Parameters

symbolPath	Type: System.String Name of the ADS symbol.
value	Type: T Object holding the value to be written to the ADS symbol

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
AdsErrorCode.

Reference

[IAdsSymbolicAccess Interface](#) [[▶ 1065](#)]

[TryWriteValue Overload](#) [[▶ 1084](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[IAdsSymbolicAccess.WriteValue.T.\(String, T\)](#) [[▶ 1089](#)]

[IAdsSymbolicAccess.TryWriteValue.T.\(String, T\)](#)

6.2.47.1.11.3 IAdsSymbolicAccess.TryWriteValue Method (ISymbol, Object)

Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
AdsErrorCode TryWriteValue(
    ISymbol symbol,
    Object val
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol the value is written to.
val	Type: System.Object The value to write.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
AdsErrorCode.

Reference

[IAdsSymbolicAccess Interface](#) [[▶ 1065](#)]

[TryWriteValue Overload \[► 1084\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

[IAdsSymbolicAccess.WriteValue\(ISymbol, Object\) \[► 1089\]](#)

[IAdsSymbolicAccess.WriteValueAsync\(ISymbol, Object, CancellationToken\) \[► 1092\]](#)

6.2.47.1.11.4 IAdsSymbolicAccess.TryWriteValue.T. Method (ISymbol, T)

Writes a value to the symbol. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryWriteValue<T>(
    ISymbol symbol,
    T val
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol the value is written to.
val	Type: T The value to write.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [AdsErrorCode \[► 664\]](#)

AdsErrorCode.

Reference

[IAdsSymbolicAccess Interface \[► 1065\]](#)

[TryWriteValue Overload \[► 1084\]](#)





[TwinCAT.Ads Namespace \[► 179\]](#)

[IAdsSymbolicAccess.WriteValue.T.\(ISymbol, T\) \[► 1090\]](#)

[IAdsSymbolicAccess.WriteValueAsync.T.\(ISymbol, T, CancellationToken\) \[► 1093\]](#)

6.2.47.1.12 IAdsSymbolicAccess.WriteValue Method

Overload List

	Name	Description
	WriteValue(ISymbol, Object) [► 1089]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValue.T.(ISymbol, T) [► 1090]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValue(String, Object) [► 1088]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol.
	WriteValue.T.(String, T) [► 1089]	Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol.

Reference

[IAdsSymbolicAccess Interface \[► 1065\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.47.1.12.1 IAdsSymbolicAccess.WriteValue Method (String, Object)

Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void WriteValue(
    string symbolPath,
    Object value
)
```

Parameters

symbolPath	Type: System.String Name of the ADS symbol.
value	Type: System.Object Object holding the value to be written to the ADS symbol

Reference

[IAdsSymbolicAccess Interface \[► 1065\]](#)

[WriteValue Overload \[► 1088\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

[IAdsSymbolicAccess.TryWriteValue\(String, Object\) \[► 1085\]](#)

[IAdsSymbolicAccess.WriteValueAsync.T.\(String, T, CancellationToken\) \[► 1091\]](#)

6.2.47.1.12.2 IAdsSymbolicAccess.WriteValue.T. Method (String, T)

Writes the passed object value to the specified ADS symbol. The parameter type must have the same binary layout as the ADS symbol.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void WriteValue<T>(
    string symbolPath,
    T value
)
```

Parameters

symbolPath	Type: System.String Name of the ADS symbol.
value	Type: T Object holding the value to be written to the ADS symbol

Type Parameters

T	the value type.
---	-----------------

Reference

[IAdsSymbolicAccess Interface](#) [► 1065]

[WriteValue Overload](#) [► 1088]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsSymbolicAccess.TryWriteValue.T.\(String, T\)](#) [► 1085]

[IAdsSymbolicAccess.WriteValueAsync.T.\(String, T, CancellationToken\)](#) [► 1091]

6.2.47.1.12.3 IAdsSymbolicAccess.WriteValue Method (ISymbol, Object)

Writes a value to the symbol. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void WriteValue(
    ISymbol symbol,
    Object val
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol the value is written to.
--------	--

val	Type: System.Object The value to write.
-----	--

Reference

[IAdsSymbolicAccess Interface](#) [► 1065]

[WriteValue Overload](#) [► 1088]

[TwinCAT.Ads Namespace](#) [► 179]

[IAdsSymbolicAccess.TryWriteValue\(ISymbol, Object\)](#) [► 1086]

[IAdsSymbolicAccess.WriteValueAsync\(ISymbol, Object, CancellationToken\)](#) [► 1092]

6.2.47.1.12.4 IAdsSymbolicAccess.WriteValue.T. Method (ISymbol, T)

Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void WriteValue<T>(
    ISymbol symbol,
    T val
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol the value is written to.
val	Type: T The value to write.

Type Parameters

T	The value type.
---	-----------------

Reference

[IAdsSymbolicAccess Interface](#) [► 1065]

[WriteValue Overload](#) [► 1088]




[TwinCAT.Ads Namespace](#) [► 179]

[IAdsSymbolicAccess.TryWriteValue.T.\(ISymbol, T\)](#) [► 1087]

[IAdsSymbolicAccess.WriteValueAsync.T.\(ISymbol, T, CancellationToken\)](#) [► 1093]

6.2.47.1.13 IAdsSymbolicAccess.WriteValueAsync Method

Overload List

	Name	Description
	WriteValueAsync(ISymbol, Object, CancellationToken) [▶ 1092]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValueAsync.T.(ISymbol, T, CancellationToken) [▶ 1093]	Writes a value to the symbol. Strings and all primitive data types(UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.
	WriteValueAsync.T.(String, T, CancellationToken) [▶ 1091]	Writes the passed object value to the specified ADS symbol.The parameter type must have the same binary layout as the ADS symbol.

Reference

[IAdsSymbolicAccess Interface](#) [▶ 1065]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.47.1.13.1 IAdsSymbolicAccess.WriteValueAsync.T. Method (String, T, CancellationToken)

Writes the passed object value to the specified ADS symbol.The parameter type must have the same binary layout as the ADS symbol.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultWrite> WriteValueAsync<T>(
    string symbolPath,
    T value,
    CancellationToken cancel
)
```

Parameters

symbolPath	Type: System.String Name of the ADS symbol.
value	Type: T Object holding the value to be written to the ADS symbol
cancel	Type: System.Threading.CancellationToken The cancel token.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [Task.ResultWrite](#) [▸ 1187].

A task that represents the asynchronous 'WriteSymbol' operation. The [ResultWrite](#) [▸ 1187] parameter contains the [ErrorCode](#) [▸ 1120] after execution.

Reference

[IAdsSymbolicAccess Interface](#) [▸ 1065]

[WriteValueAsync Overload](#) [▸ 1091]

[TwinCAT.Ads Namespace](#) [▸ 179]

[IAdsSymbolicAccess.WriteValue.T.\(String, T\)](#) [▸ 1089]

[IAdsSymbolicAccess.TryWriteValue.T.\(ISymbol, T\)](#) [▸ 1087]

6.2.47.1.13.2 IAdsSymbolicAccess.WriteValueAsync Method (ISymbol, Object, CancellationToken)

Writes a value to the symbol. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

Namespace: [TwinCAT.Ads](#) [▸ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultWrite> WriteValueAsync(
    ISymbol symbol,
    Object val,
    CancellationToken cancel
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▸ 2691] The symbol the value is written to.
val	Type: System.Object The value to write.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite](#) [▸ 1187].

A task that represents the asynchronous 'WriteSymbol' operation. The [ResultWrite](#) [▸ 1187] parameter contains the [ErrorCode](#) [▸ 1120] after execution.

Reference

[IAdsSymbolicAccess Interface](#) [▸ 1065]

[WriteValueAsync Overload](#) [▸ 1091]

[TwinCAT.Ads Namespace](#) [▸ 179]

[IAdsSymbolicAccess.WriteValue\(ISymbol, Object\)](#) [▸ 1089]

[IAdsSymbolicAccess.TryWriteValue\(ISymbol, Object\) \[► 1086\]](#)

6.2.47.1.13.3 IAdsSymbolicAccess.WriteValueAsync.T. Method (ISymbol, T, CancellationToken)

Writes a value to the symbol. Strings and all primitive data types (UInt32, Int32, Bool etc.) are supported. If a string is passed as parameter, the method attempts to parse the string according to the ADS data type of the symbol.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultWrite> WriteValueAsync<T>(
    ISymbol symbol,
    T val,
    CancellationToken cancel
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol the value is written to.
val	Type: T The value to write.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Type Parameters

T	The value type.
---	-----------------

Return Value

Type: [Task.ResultWrite \[► 1187\]](#).

A task that represents the asynchronous 'WriteSymbol' operation. The [ResultWrite \[► 1187\]](#) parameter contains the [ErrorCode \[► 1120\]](#) after execution.

Reference

[IAdsSymbolicAccess Interface \[► 1065\]](#)

[WriteValueAsync Overload \[► 1091\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

[IAdsSymbolicAccess.WriteValue.T.\(ISymbol, T\) \[► 1090\]](#)

[IAdsSymbolicAccess.TryWriteValue.T.\(ISymbol, T\) \[► 1087\]](#)

6.2.48 IAdsSymbolTableProvider Interface

Interface IAdsSymbolTableProvider

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3




Syntax

C#

```
public interface IAdsSymbolTableProvider
```

The IAdsSymbolTableProvider type exposes the following members.

Methods

	Name	Description
	GetSymbolTableAsync nc [▶ 1094]	Gets the symbol table asynchronously
	SetSymbolEncoding [▶ 1095]	Sets the default encoding.
	TryGetSymbolTable [▶ 1095]	Get the symbol table.




Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.48.1 IAdsSymbolTableProvider Methods

The [IAdsSymbolTableProvider](#) [[▶ 1093](#)] type exposes the following members.

Methods

	Name	Description
	GetSymbolTableAsync nc [▶ 1094]	Gets the symbol table asynchronously
	SetSymbolEncoding [▶ 1095]	Sets the default encoding.
	TryGetSymbolTable [▶ 1095]	Get the symbol table.

Reference

[IAdsSymbolTableProvider Interface](#) [[▶ 1093](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.48.1.1 IAdsSymbolTableProvider.GetSymbolTableAsync Method

Gets the symbol table asynchronously

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
Task<ResultValue<ISymbolInfoTable>> GetSymbolTableAsync(
    CancellationToken cancel
)
```

Parameters

cancel Type: [System.Threading.CancellationToken](#)
The cancellation token.

Return Value

Type: [Task.ResultValue](#) [[▶ 1181](#)].[ISymbolInfoTable](#)..

A task that represents the asynchronous 'RegisterSymbolVersionChanged' operation. The [ResultValue.TValue](#). [[▶ 1181](#)] parameter contains the value [Value](#) [[▶ 1185](#)] and the return code [ErrorCode](#) [[▶ 1120](#)] of the ADS communication after execution.

Reference

[IAdsSymbolTableProvider Interface](#) [[▶ 1093](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.48.1.2 **IAdsSymbolTableProvider.SetSymbolEncoding Method**

Sets the default encoding.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
void SetSymbolEncoding(  
    Encoding encoding  
)
```

Parameters

encoding Type: [System.Text.Encoding](#)
The encoding.

Reference

[IAdsSymbolTableProvider Interface](#) [[▶ 1093](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.48.1.3 **IAdsSymbolTableProvider.TryGetSymbolTable Method**

Get the symbol table.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
AdsErrorCode TryGetSymbolTable(  
    out ISymbolInfoTable?? table  
)
```

Parameters

table Type: [ISymbolInfoTable](#).
The symbol table.

Return Value

Type: [AdsErrorCode](#) [▶ 664]
The ADS Error Code.

Reference

[IAdsSymbolTableProvider](#) Interface [▶ 1093]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.49 INotification Interface

Common INotification interface

Namespace: [TwinCAT.Ads](#) [▶ 179]





Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public interface INotification
```

The INotification type exposes the following members.

Properties

	Name	Description
	Data [▶ 1097]	The notification Data.
	Handle [▶ 1097]	The notification handle
	TimeStamp [▶ 1097]	Gets the time stamp of the INotification
	UserData [▶ 1098]	Attached UserData/Tag at the INotification





Reference

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.49.1 INotification Properties

The [INotification](#) [▶ 1096] type exposes the following members.

Properties

	Name	Description
	Data [▶ 1097]	The notification Data.
	Handle [▶ 1097]	The notification handle
	TimeStamp [▶ 1097]	Gets the time stamp of the INotification [▶ 1096]
	UserData [▶ 1098]	Attached UserData/Tag at the INotification [▶ 1096]

Reference

[INotification Interface \[► 1096\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.49.1.1 INotification.Data Property

The notification Data.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ReadOnlyMemory<byte> Data { get; }
```

Property Value

Type: [ReadOnlyMemory.Byte](#).

The data.

Reference

[INotification Interface \[► 1096\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.49.1.2 INotification.Handle Property

The notification handle

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
uint Handle { get; }
```

Property Value

Type: [UInt32](#)

The handle.

Remarks

This can be the Client handle or the Server handle!

Reference

[INotification Interface \[► 1096\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.49.1.3 INotification.TimeStamp Property

Gets the time stamp of the [INotification \[► 1096\]](#)

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
DateTimeOffset TimeStamp { get; }
```

Property Value

Type: [DateTimeOffset](#)

The time stamp.

Reference

[INotification Interface \[▸ 1096\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.49.1.4 INotification.UserData Property

Attached UserData/Tag at the [INotification \[▸ 1096\]](#)

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object? UserData { get; }
```

Property Value

Type: [Object](#)

The user data.

Reference

[INotification Interface \[▸ 1096\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.50 INotificationSettings Interface

Interface for Notification Settings Implements the [IComparable.T](#).

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#

```
public interface INotificationSettings : IComparable<INotificationSettings>
```

The INotificationSettings type exposes the following members.

Methods

	Name	Description
	<u>CompareTo</u>	Compares the current instance with another object of the same type and returns an integer that indicates whether the current instance precedes, follows, or occurs in the same position in the sort order as the other object. (Inherited from <u>IComparable.INotificationSettings..</u>)

Reference


[TwinCAT.Ads Namespace \[▶ 179\]](#)

[System.IComparable.T.](#)

6.2.50.1 INotificationSettings Methods

The [INotificationSettings \[▶ 1098\]](#) type exposes the following members.

Methods

	Name	Description
	<u>CompareTo</u>	Compares the current instance with another object of the same type and returns an integer that indicates whether the current instance precedes, follows, or occurs in the same position in the sort order as the other object. (Inherited from <u>IComparable.INotificationSettings [▶ 1098]..</u>)

Reference

[INotificationSettings Interface \[▶ 1098\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.51 IRouterNotificationProvider Interface

Interface for AMS Router Notifications.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#

```
public interface IRouterNotificationProvider
```

The IRouterNotificationProvider type exposes the following members.

Events

	Name	Description
	<u>RouterStateChanged</u> [▶ 1100]	(Local) Router state changed event.


Reference

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.51.1 IRouterNotificationProvider Events

The [IRouterNotificationProvider](#) [[▶ 1099](#)] type exposes the following members.

Events

	Name	Description
	RouterStateChanged [▶ 1100]	(Local) Router state changed event.

Reference

[IRouterNotificationProvider Interface](#) [[▶ 1099](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.51.1.1 IRouterNotificationProvider.RouterStateChanged Event

(Local) Router state changed event.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
event EventHandler<AmsRouterNotificationEventArgs> RouterStateChanged
```

Value

Type: [System.EventHandler.AmsRouterNotificationEventArgs](#) [[▶ 798](#)].

Remarks

This event indicates, that a changed event is received from the Local AmsRouter independant of the connected target address. A remote system RouterStateChanged event cannot be received at another system - it cannot traverse TwinCAT systems.

Reference

[IRouterNotificationProvider Interface](#) [[▶ 1099](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.52 Notification Class

Class Notification. Implements the [INotification](#) [[▶ 1096](#)]

Inheritance Hierarchy

[System.Object](#)
[TwinCAT.Ads.Notification](#)

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax

C#





```
public class Notification : INotification
```

The Notification type exposes the following members.







Constructors

	Name	Description
	Notification(INotification) [▶ 1102]	Initializes a new instance of the Notification class.
	Notification(UInt32, DateTimeOffset, Object, ReadOnlyMemory.BYTE) [▶ 1103]	Initializes a new instance of the Notification class.

Properties

	Name	Description
	Data [▶ 1104]	The notification Data.
	Handle [▶ 1105]	The notification handle
	TimeStamp [▶ 1105]	Gets the time stamp of the INotification [▶ 1096]
	UserData [▶ 1105]	Attached UserData/Tag at the INotification [▶ 1096]

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)



Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[TwinCAT.Ads.INotification](#) [[▶ 1096](#)]

6.2.52.1 Notification Constructor

Overload List

	Name	Description
	Notification(INotification) [▶ 1102]	Initializes a new instance of the Notification [▶ 1100] class.
	Notification(UInt32, DateTimeOffset, Object, ReadOnlyMemory.BYTE) [▶ 1103]	Initializes a new instance of the Notification [▶ 1100] class.

Reference

[Notification Class](#) [[▶ 1100](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.52.1.1 Notification Constructor (INotification)

Initializes a new instance of the [Notification](#) [[▶ 1100](#)] class.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Notification(
    INotification notification
)
```

Parameters

notification	Type: TwinCAT.Ads.INotification [▶ 1096] The notification.
--------------	---

Reference

[Notification Class](#) [[▶ 1100](#)]

[Notification Overload](#) [[▶ 1102](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.52.1.2 Notification Constructor (UInt32, DateTimeOffset, Object, ReadOnlyMemory`1)

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Notification(
    uint handle,
    DateTimeOffset dateTime,
    Object userData,
    ReadOnlyMemory data
)
```

Parameters

- handle Type: [System.UInt32](#)
- dateTime Type: [System.DateTimeOffset](#)
- userData Type: [System.Object](#)
- data Type: [ReadOnlyMemory](#)

Reference

- [Notification Class \[▶ 1100\]](#)
- [Notification Overload \[▶ 1102\]](#)
- [TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.52.1.3 Notification Constructor (UInt32, DateTimeOffset, Object, ReadOnlyMemory.Byte.)

Initializes a new instance of the [Notification \[▶ 1100\]](#) class.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Notification(
    uint handle,
    DateTimeOffset dateTime,
    Object? userData,
    ReadOnlyMemory<byte> data
)
```

Parameters

handle	Type: System.UInt32 The handle.
dateTime	Type: System.DateTimeOffset The date time.
userData	Type: System.Object The user data.
data	Type: System.ReadOnlyMemory.Byte. The data.

Reference

[Notification Class \[▶ 1100\]](#)





[Notification Overload \[▶ 1102\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.52.2 Notification Properties

The [Notification \[▶ 1100\]](#) type exposes the following members.

Properties

	Name	Description
	Data [▶ 1104]	The notification Data.
	Handle [▶ 1105]	The notification handle
	TimeStamp [▶ 1105]	Gets the time stamp of the INotification [▶ 1096]
	UserData [▶ 1105]	Attached UserData/Tag at the INotification [▶ 1096]

Reference

[Notification Class \[▶ 1100\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.52.2.1 Notification.Data Property

The notification Data.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyMemory<byte> Data { get; }
```

Property Value

Type: [ReadOnlyMemory.Byte](#).

The data.

Implements

[INotification.Data \[▶ 1097\]](#)

Reference

[Notification Class \[▶ 1100\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.52.2 Notification.Handle Property

The notification handle

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint Handle { get; }
```

Property Value

Type: [UInt32](#)

The handle.

Implements

[INotification.Handle](#) [► 1097]

Reference

[Notification Class](#) [► 1100]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.52.3 Notification.TimeStamp Property

Gets the time stamp of the [INotification](#) [► 1096]

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTimeOffset TimeStamp { get; }
```

Property Value

Type: [DateTimeOffset](#)

The time stamp.

Implements

[INotification.TimeStamp](#) [► 1097]

Reference

[Notification Class](#) [► 1100]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.52.4 Notification.UserData Property

Attached UserData/Tag at the [INotification](#) [► 1096]

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object? UserData { get; }
```

Property Value

Type: [Object](#)

The user data.

Implements

[INotification.UserData](#) [► 1098]

Reference







[Notification Class](#) [► 1100]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.52.3 Notification Methods

The [Notification](#) [► 1100] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[Notification Class](#) [► 1100]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.53 NotificationSettings Class

Notification communication settings

Inheritance Hierarchy

System.Object

TwinCAT.Ads.NotificationSettings

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax

C#








```
public class NotificationSettings : INotificationSettings,
    IComparable<INotificationSettings>
```

The NotificationSettings type exposes the following members.





Constructors




	Name	Description
	NotificationSettings (AdsTransMode, Int32, Int32) [▶ 1108]	Initializes a new instance of the NotificationSettings class.
	NotificationSettings (AdsTransMode, TimeSpan, TimeSpan) [▶ 1109]	Initializes a new instance of the NotificationSettings class.

Properties





	Name	Description
	CycleTime [▶ 1110]	Gets the cycle time (in milliseconds) for AdsNotifications.
 	Default [▶ 1111]	Gets the default Settings (AdsTransMode.OnChange, CycleTime 200 ms, MaxDelay: off)
 	ImmediatelyOnChange [▶ 1111]	Gets the settings for a 'Immediate on change' notification.
	MaxDelay [▶ 1111]	Gets the Maximum Delay Time (in milliseconds) for AdsNotifications.
	NotificationMode [▶ 1112]	Gets the ADS Transmission mode.

Methods

	Name	Description
	CompareTo [▶ 1113]	Compares this NotificationSettings in term of priorities to the other NotificationSettings.
	Equals [▶ 1113]	Determines whether the specified <u>Object</u> is equal to this instance. (Overrides <u>Object.Equals(Object)</u> .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from <u>Object</u> .)
	GetHashCode [▶ 1114]	Gets the HashCode of the Address (Overrides <u>Object.GetHashCode</u> .)

	Name	Description
	GetType	Gets the <u>Type</u> of the current instance. (Inherited from <u>Object</u> .)
	MemberwiseClone	Creates a shallow copy of the current <u>Object</u> . (Inherited from <u>Object</u> .)
	ToString	Returns a string that represents the current object. (Inherited from <u>Object</u> .)

Operators

	Name	Description
 	Equality [▶ 1115]	Implements the == operator.
 	Inequality [▶ 1115]	Implements the != operator.

Remarks



AdsTransMode [▶ 740]	Parameter semantic
CyclicInContext [▶ 740]	Value of parameter CycleTime is interpreted as task context number AdsSymbol.ContextMask/>
OnChangeInContext [▶ 740]	Value of parameter CycleTime is interpreted as task context number AdsSymbol.ContextMask/>

Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.53.1 NotificationSettings Constructor

Overload List

	Name	Description
	NotificationSettings (AdsTransMode , Int32 , Int32) [▶ 1108]	Initializes a new instance of the NotificationSettings [▶ 1106] class.
	NotificationSettings (AdsTransMode , TimeSpan , TimeSpan) [▶ 1109]	Initializes a new instance of the NotificationSettings [▶ 1106] class.

Reference

[NotificationSettings Class](#) [[▶ 1106](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.53.1.1 NotificationSettings Constructor (AdsTransMode, Int32, Int32)

Initializes a new instance of the [NotificationSettings](#) [[▶ 1106](#)] class.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public NotificationSettings(
    AdsTransMode mode,
    int cycleTime,
    int maxDelay
)
```

Parameters

mode	Type: TwinCAT.Ads.AdsTransMode [▶ 740] The ADS Transmission mode.
cycleTime	Type: System.Int32 The cycle time in ms.
maxDelay	Type: System.Int32 The maximum delay in ms

Reference

[NotificationSettings Class](#) [▶ 1106]

[NotificationSettings Overload](#) [▶ 1108]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.53.1.2 NotificationSettings Constructor (AdsTransMode, TimeSpan, TimeSpan)

Initializes a new instance of the [NotificationSettings](#) [▶ 1106] class.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public NotificationSettings(
    AdsTransMode mode,
    TimeSpan cycleTime,
    TimeSpan maxDelay
)
```

Parameters

mode	Type: TwinCAT.Ads.AdsTransMode [▶ 740] The ADS Transmission mode.
cycleTime	Type: System.TimeSpan The cycle time in ms.
maxDelay	Type: System.TimeSpan The maximum delay in ms

Reference

[NotificationSettings Class](#) [▶ 1106]








[NotificationSettings Overload](#) [▶ 1108]

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.53.2 NotificationSettings Properties

The [NotificationSettings \[▸ 1106\]](#) type exposes the following members.

Properties

	Name	Description
	CycleTime [▸ 1110]	Gets the cycle time (in milliseconds) for AdsNotifications.
 	Default [▸ 1111]	Gets the default Settings (AdsTransMode.OnChange, CycleTime 200 ms, MaxDelay: off)
 	ImmediatelyOnChange [▸ 1111]	Gets the settings for a 'Immediate on change' notification.
	MaxDelay [▸ 1111]	Gets the Maximum Delay Time (in milliseconds) for AdsNotifications.
	NotificationMode [▸ 1112]	Gets the ADS Transmission mode.

Reference

[NotificationSettings Class \[▸ 1106\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.53.2.1 NotificationSettings.CycleTime Property

Gets the cycle time (in milliseconds) for AdsNotifications.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int CycleTime { get; }
```

Property Value

Type: [Int32](#)

The cycle time.

Remarks

The ADS server checks if the value changes in this time slice. The unit is 1ms

Reference

[NotificationSettings Class \[▸ 1106\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.53.2.2 NotificationSettings.Default Property

Gets the default Settings (AdsTransMode.OnChange, CycleTime 200 ms, MaxDelay: off)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static NotificationSettings Default { get; }
```

Property Value

Type: [NotificationSettings](#) [► 1106]

The default.

Reference

[NotificationSettings Class](#) [► 1106]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.53.2.3 NotificationSettings.ImmediatelyOnChange Property

Gets the settings for a 'Immediate on change' notification.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static NotificationSettings ImmediatelyOnChange { get; }
```

Property Value

Type: [NotificationSettings](#) [► 1106]

The immediately on change.

Remarks

AdsTransMode.OnChange, CycleTime: 0 ms, MaxDelay: off)

Reference

[NotificationSettings Class](#) [► 1106]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.53.2.4 NotificationSettings.MaxDelay Property

Gets the Maximum Delay Time (in milliseconds) for AdsNotifications.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int MaxDelay { get; }
```

Property Value

Type: [Int32](#)

The maximum Delay time for ADS Notifications.

Reference

[NotificationSettings Class](#) [▸ [1106](#)]

[TwinCAT.Ads Namespace](#) [▸ [179](#)]

6.2.53.2.5 NotificationSettings.NotificationMode Property

Gets the ADS Transmission mode.

Namespace: [TwinCAT.Ads](#) [▸ [179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsTransMode NotificationMode { get; }
```

Property Value

Type: [AdsTransMode](#) [▸ [740](#)]

The Transmission mode.

Reference






[NotificationSettings Class](#) [▸ [1106](#)]



[TwinCAT.Ads Namespace](#) [▸ [179](#)]

6.2.53.3 NotificationSettings Methods

The [NotificationSettings](#) [▸ [1106](#)] type exposes the following members.

Methods

	Name	Description
	CompareTo [▸ 1113]	Compares this NotificationSettings [▸ 1106] in term of priorities to the other NotificationSettings [▸ 1106].
	Equals [▸ 1113]	Determines whether the specified Object is equal to this instance. (Overrides Object.Equals(Object) .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode [▸ 1114]	Gets the GetHashCode of the Address (Overrides Object.GetHashCode() .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)

	Name	Description
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[NotificationSettings Class](#) [► 1106]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.53.3.1 NotificationSettings.CompareTo Method

Compares this [NotificationSettings](#) [► 1106] in term of priorities to the other [NotificationSettings](#) [► 1106].

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int CompareTo(
    INotificationSettings? other
)
```

Parameters

other	Type: TwinCAT.Ads.INotificationSettings [► 1098] The other.
-------	--

Return Value

Type: [Int32](#)

1: this has higher priority (shorter times), 0: Equal, -1: Lower priority

Implements

[IComparable.T..CompareTo\(T\)](#)

Reference

[NotificationSettings Class](#) [► 1106]

[TwinCAT.Ads Namespace](#) [► 179]

[NotificationSettingsPriorityComparer](#)

6.2.53.3.2 NotificationSettings.Equals Method

Determines whether the specified [Object](#) is equal to this instance.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool Equals(
    Object? obj
)
```

Parameters

obj	Type: System.Object The object to compare with the current object.
-----	---

Return Value

Type: [Boolean](#)

true if the specified [Object](#) is equal to this instance; otherwise, false.

Reference

[NotificationSettings Class](#) [► 1106]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.53.3 NotificationSettings.GetHashCode Method

Gets the HashCode of the Address

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override int GetHashCode()
```

Return Value

Type: [Int32](#)

A hash code for this instance, suitable for use in hashing algorithms and data structures like a hash table.

Reference




[NotificationSettings Class](#) [► 1106]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.53.4 NotificationSettings Operators

The [NotificationSettings](#) [► 1106] type exposes the following members.

Operators

	Name	Description
	Equality [► 1115]	Implements the == operator.
		
	Inequality [► 1115]	Implements the != operator.

	Name	Description
S		

Reference

[NotificationSettings Class](#) [► 1106]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.53.4.1 NotificationSettings.Equality Operator

Implements the == operator.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool operator ==(
    NotificationSettings? settings1,
    NotificationSettings? settings2
)
```

Parameters

settings1	Type: TwinCAT.Ads.NotificationSettings [► 1106] The settings1.
settings2	Type: TwinCAT.Ads.NotificationSettings [► 1106] The settings2.

Return Value

Type: [Boolean](#)

The result of the operator.

Reference

[NotificationSettings Class](#) [► 1106]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.53.4.2 NotificationSettings.Inequality Operator

Implements the != operator.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool operator !=(
    NotificationSettings? settings1,
    NotificationSettings? settings2
)
```

Parameters

settings1	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The settings1.
settings2	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The settings2.

Return Value

Type: [Boolean](#)
The result of the operator.

Reference

[NotificationSettings Class](#) [▶ 1106]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.54 ResultAds Class

Base class for an (asynchronous) ADS Task Result

Inheritance Hierarchy

[System.Object](#)
 TwinCAT.Ads.ResultAds
 More... [▶ 1118]

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**



```
public class ResultAds
```




The ResultAds type exposes the following members.

Constructors




















	Name	Description
	ResultAds(AdsError Code) [▶ 1118]	Initializes a new instance of the ResultAds class.
	ResultAds(AdsError Code, UInt32) [▶ 1119]	Initializes a new instance of the ResultAds class.

Properties

	Name	Description
	Empty [▶ 1120]	Gets an empty ResultAds initialized to None [▶ 664].
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result object.

	Name	Description
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds state is failed.
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds state is succeeded.

Methods

	Name	Description
 	CreateError(AdsErrorCode) [▶ 1123]	Creates an Error Result.
 	CreateError(AdsErrorCode, UInt32) [▶ 1124]	Creates an Error Result.
 	CreateSuccess. [▶ 1125]	Creates a success result.
 	CreateSuccess(UInt32) [▶ 1126]	Creates a success result.
 	CreateSuccess.TValue(TValue) [▶ 1126]	Creates a success result.
 	CreateSuccess.TValue(TValue, UInt32) [▶ 1127]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Operators

	Name	Description
	.(ResultAds to ResultAccess) [▶ 1128]	Performs an implicit conversion from ResultAds to ResultAccess [▶ 3197].

Remarks

The base class is used whenever an asynchronous task communicates via ADS and should return its [AdsErrorCode](#) [▶ 664] within its tasks result.

Reference

[TwinCAT.Ads Namespace](#) [▶ 179]

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.ResultAds](#)

[TwinCAT.Ads.ResultDeviceInfo](#) [▶ 1132]

[TwinCAT.Ads.ResultHandle](#) [▶ 1136]

[TwinCAT.Ads.ResultRead](#) [▶ 1143]

[TwinCAT.Ads.ResultReadAdsState](#) [▶ 1146]

[TwinCAT.Ads.ResultReadDeviceState](#) [▶ 1157]

[TwinCAT.Ads.ResultRpcMethod](#) [▶ 1176]

[TwinCAT.Ads.ResultValue.TValue.](#) [▶ 1181]

[TwinCAT.Ads.ResultWrite](#) [▶ 1187]

[TwinCAT.Ads.ResultWriteControl](#) [▶ 1236]

[TwinCAT.Ads.SumCommand.ResultSumCommand](#) [▶ 1533]

Also see about this

- ▣ [ResultAds Constructor](#) [▶ 1118]
- ▣ [ResultAds.CreateError Method](#) [▶ 1123]
- ▣ [ResultAnyValue Class](#) [▶ 1129]

6.2.54.1 ResultAds Constructor**Overload List**

	Name	Description
	ResultAds(AdsError Code) [▶ 1118]	Initializes a new instance of the ResultAds [▶ 1116] class.
	ResultAds(AdsError Code, UInt32) [▶ 1119]	Initializes a new instance of the ResultAds [▶ 1116] class.

Reference

[ResultAds Class](#) [▶ 1116]

[TwinCAT.Ads Namespace](#) [▶ 179]

Also see about this

- ▣ [ResultAds Class](#) [▶ 1116]
- ▣ [TwinCAT.Ads Namespace](#) [▶ 179]

6.2.54.1.1 ResultAds Constructor (AdsErrorCode)

Initializes a new instance of the [ResultAds](#) [▶ 1116] class.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ResultAds(
    AdsErrorCode errorCode
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The Ads ErrorCode
-----------	---

Reference

[ResultAds Class](#) [▶ 1116]

[ResultAds Overload](#) [▶ 1118]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.54.1.2 ResultAds Constructor (AdsErrorCode, UInt32)

Initializes a new instance of the [ResultAds](#) [▶ 1116] class.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ResultAds(
    AdsErrorCode errorCode,
    uint invokeId
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The Ads ErrorCode
invokeld	Type: System.UInt32 The ADS request invoke identifier or 0

Reference

[ResultAds Class](#) [▶ 1116]







[ResultAds Overload](#) [▶ 1118]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.54.2 ResultAds Properties

The [ResultAds](#) [▶ 1116] type exposes the following members.

Properties

	Name	Description
	Empty [▶ 1120]	Gets an empty ResultAds [▶ 1116] initialized to None [▶ 664].
		
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object.
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed.
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded.

Reference

[ResultAds Class](#) [[▶ 1116](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.54.2.1 ResultAds.Empty Property

Gets an empty [ResultAds](#) [[▶ 1116](#)] initialized to [None](#) [[▶ 664](#)].

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static ResultAds Empty { get; }
```

Property Value

Type: [ResultAds](#) [[▶ 1116](#)]

The empty.

Reference

[ResultAds Class](#) [[▶ 1116](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.54.2.2 ResultAds.ErrorCode Property

Gets the ADS Error code bound to this [Result](#) [[▶ 1116](#)] object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public AdsErrorCode ErrorCode { get; }
```


Property Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The error code.

Reference

[ResultAds Class](#) [[▶ 1116](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.54.2.3 ResultAds.Failed Property

Gets a value indicating whether the [ResultAds](#) [[▶ 1116](#)] state is failed.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Failed { get; }
```

Property Value

Type: [Boolean](#)

true if failed; otherwise, false.

Reference

[ResultAds Class](#) [[▶ 1116](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.54.2.4 ResultAds.Succeeded Property

Gets a value indicating whether the [ResultAds](#) [[▶ 1116](#)] state is succeeded.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Succeeded { get; }
```

Property Value

Type: [Boolean](#)

true if succeeded; otherwise, false.

Reference

[ResultAds Class](#) [[▶ 1116](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.54.2.5 ResultAds.InvokeId Property

Gets the ADS requests invoke identifier (or 0 if not supported)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint InvokeId { get; }
```

Property Value

Type: [UInt32](#)

The invoke identifier.

Reference

















[ResultAds Class](#) [► 1116]




[TwinCAT.Ads Namespace](#) [► 179]

6.2.54.3 ResultAds Methods

The [ResultAds](#) [► 1116] type exposes the following members.

Methods

	Name	Description
 	CreateError(AdsError rCode) [► 1123]	Creates an Error Result.
 	CreateError(AdsError rCode, UInt32) [► 1124]	Creates an Error Result.
 	CreateSuccess. [► 1125]	Creates a success result.
 	CreateSuccess(UInt32) [► 1126]	Creates a success result.
 	CreateSuccess.TValue(TValue) [► 1126]	Creates a success result.
 	CreateSuccess.TValue(TValue, UInt32) [► 1127]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)

	Name	Description
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116]
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultAds Class](#) [[▶ 1116](#)]





[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

 [ResultAds.CreateError Method](#) [[▶ 1123](#)]

6.2.54.3.1 ResultAds.CreateError Method

Overload List

	Name	Description
 	CreateError(AdsError rCode) [▶ 1123]	Creates an Error Result.
 	CreateError(AdsError rCode, UInt32) [▶ 1124]	Creates an Error Result.

Reference

[ResultAds Class](#) [[▶ 1116](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

 [TwinCAT.Ads Namespace](#) [[▶ 179](#)]

 [ResultAds.CreateError.TValue. Method \(AdsErrorCode, TValue\)](#) [[▶ 1124](#)]

6.2.54.3.1.1 ResultAds.CreateError Method (AdsErrorCode)

Creates an Error Result.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultAds CreateError(
    AdsErrorCode errorCode
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
-----------	---

Return Value

Type: [ResultAds](#) [[▶ 1116](#)]
ResultValue<T>.

Reference

[ResultAds Class](#) [[▶ 1116](#)]

[CreateError Overload](#) [[▶ 1123](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.54.3.1.2 ResultAds.CreateError.TValue. Method (AdsErrorCode, TValue)

Creates an Error Result.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public static ResultValue<TValue> CreateError<TValue>(
    AdsErrorCode errorCode,
    TValue defaultValue
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
defaultValue	Type: TValue The default value.

Type Parameters

TValue	The type of the t value.
--------	--------------------------

Return Value

Type: [ResultValue](#) [[▶ 1181](#)].TValue.
ResultValue<T>.

Reference

[ResultAds Class](#) [[▶ 1116](#)]

[CreateError Overload](#) [[▶ 1123](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.54.3.1.3 ResultAds.CreateError Method (AdsErrorCode, UInt32)

Creates an Error Result.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultAds CreateError(
    AdsErrorCode errorCode,
    uint invokeId
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
invokeld	Type: System.UInt32 The ADS request invoke identifier or 0

Return Value

Type: [ResultAds](#) [[▶ 1116](#)]
ResultValue<T>.

Reference









[ResultAds Class](#) [[▶ 1116](#)]

[CreateError Overload](#) [[▶ 1123](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.54.3.2 ResultAds.CreateSuccess Method

Overload List

	Name	Description
 	CreateSuccess. [▶ 1125]	Creates a success result.
 	CreateSuccess.TValue. TValue(TValue) [▶ 1126]	Creates a success result.
 	CreateSuccess(UInt32) [▶ 1126]	Creates a success result.
 	CreateSuccess.TValue. TValue(TValue, UInt32) [▶ 1127]	Creates a success result.

Reference

[ResultAds Class](#) [[▶ 1116](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.54.3.2.1 ResultAds.CreateSuccess Method

Creates a success result.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultAds CreateSuccess()
```

Return Value

Type: [ResultAds](#) [▶ 1116]

ResultValue<T>.

Reference

[ResultAds Class](#) [▶ 1116]

[CreateSuccess Overload](#) [▶ 1125]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.54.3.2.2 ResultAds.CreateSuccess.TValue. Method (TValue)

Creates a success result.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultValue<TValue> CreateSuccess<TValue>(
    TValue value
)
```

Parameters

value	Type: TValue The value.
-------	----------------------------

Type Parameters

TValue

Return Value

Type: [ResultValue](#) [▶ 1181].TValue.

ResultValue<T>.

Reference

[ResultAds Class](#) [▶ 1116]

[CreateSuccess Overload](#) [▶ 1125]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.54.3.2.3 ResultAds.CreateSuccess Method (UInt32)

Creates a success result.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultAds CreateSuccess(  
    uint invokeId  
)
```

Parameters

invokeld	Type: System.UInt32
----------	-------------------------------------

Return Value

Type: [ResultAds](#) [► 1116]

ResultValue<T>.

Reference

[ResultAds Class](#) [► 1116]

[CreateSuccess Overload](#) [► 1125]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.54.3.2.4 ResultAds.CreateSuccess.TValue. Method (TValue, UInt32)

Creates a success result.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultValue<TValue> CreateSuccess<TValue>(  
    TValue value,  
    uint invokeId  
)
```

Parameters

value	Type: TValue The value.
invokeld	Type: System.UInt32 The ADS request invoke identifier or 0

Type Parameters

TValue

Return Value

Type: [ResultValue](#) [► 1181].TValue.

ResultValue<T>.

Reference[ResultAds Class](#) [[▶ 1116](#)][CreateSuccess Overload](#) [[▶ 1125](#)][TwinCAT.Ads Namespace](#) [[▶ 179](#)]**6.2.54.3 ResultAds.SetError Method**Sets the error state of this [ResultAds](#) [[▶ 1116](#)]**Namespace:** [TwinCAT.Ads](#) [[▶ 179](#)]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public void SetError(
    AdsErrorCode error
)
```

Parameters

error	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error.
-------	--

Reference[ResultAds Class](#) [[▶ 1116](#)][TwinCAT.Ads Namespace](#) [[▶ 179](#)]**6.2.54.4 ResultAds Type Conversions**The [ResultAds](#) [[▶ 1116](#)] type exposes the following members.**Operators**

	Name	Description
	.(ResultAds to ResultAccess) [▶ 1128]	Performs an implicit conversion from ResultAds [▶ 1116] to ResultAccess [▶ 3197].

Reference[ResultAds Class](#) [[▶ 1116](#)][TwinCAT.Ads Namespace](#) [[▶ 179](#)]**6.2.54.4.1 ResultAds . Conversion (ResultAds to ResultAccess)**Performs an implicit conversion from [ResultAds](#) [[▶ 1116](#)] to [ResultAccess](#) [[▶ 3197](#)].**Namespace:** [TwinCAT.Ads](#) [[▶ 179](#)]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static implicit operator ResultAccess (
    ResultAds resultAds
)
```

Parameters

resultAds	Type: TwinCAT.Ads.ResultAds [▸ 1116] The ADS result.
-----------	---

Return Value

Type: [ResultAccess \[▸ 3197\]](#)
The result of the conversion.

Reference

[ResultAds Class \[▸ 1116\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.55 ResultAnyValue Class

Result object for asynchronous reading an 'AnyValue'/Primitive Value via tasks.

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.ResultAds \[▸ 1116\]](#)

[TwinCAT.Ads.ResultValue \[▸ 1181\].Object.](#)

[TwinCAT.Ads.ResultAnyValue](#)

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229






Syntax


C#

```
public class ResultAnyValue : ResultValue<Object>
```








The ResultAnyValue type exposes the following members.

Properties

	Name	Description
	Empty [▸ 1131]	Gets the empty ResultAnyValue object.
	ErrorCode [▸ 1120]	Gets the ADS Error code bound to this Result [▸ 1116] object. (Inherited from ResultAds [▸ 1116] .)
	Failed [▸ 1121]	Gets a value indicating whether the ResultAds [▸ 1116] state is failed. (Inherited from ResultAds [▸ 1116] .)
	Invokeld [▸ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▸ 1116] .)
	Succeeded [▸ 1121]	Gets a value indicating whether the ResultAds [▸ 1116] state is succeeded. (Inherited from ResultAds [▸ 1116] .)

	Name	Description
	Value [▶ 1185]	The value object. (Inherited from ResultValue.TValue . [▶ 1181].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

 [ResultAnyValue.Value Property](#) [[▶ 1131](#)]

6.2.55.1 ResultAnyValue Constructor

Initializes a new instance of the [ResultAnyValue](#) [[▶ 1129](#)] struct.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public ResultAnyValue(
    AdsErrorCode errorCode,
    Object? value
)
```

Parameters

errorCode Type: [TwinCAT.Ads.AdsErrorCode](#) [[▶ 664](#)]
The error code.

value Type: [System.Object](#)
The value.

Reference







[ResultAnyValue Class](#) [[▶ 1129](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.55.2 ResultAnyValue Properties

The [ResultAnyValue](#) [[▶ 1129](#)] type exposes the following members.

Properties

	Name	Description
	Empty [▶ 1131]	Gets the empty ResultAnyValue [▶ 1129] object.
S		
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)
	Value [▶ 1185]	The value object. (Inherited from ResultValue.TValue . [▶ 1181].)

Reference

[ResultAnyValue Class](#) [[▶ 1129](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

 [ResultAnyValue.Value Property](#) [[▶ 1131](#)]

6.2.55.2.1 ResultAnyValue.Empty Property

Gets the empty [ResultAnyValue](#) [[▶ 1129](#)] object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultAnyValue Empty { get; }
```

Property Value

Type: [ResultAnyValue](#) [[▶ 1129](#)]

The empty / unprocessed result.

Reference

[ResultAnyValue Class](#) [[▶ 1129](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.55.2.2 ResultAnyValue.Value Property

Gets the read value.

Namespace: [TwinCAT.Ads](#) [▸ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public Object? Value { get; }
```

Property Value

Type: [Object](#)

Reference








[ResultAnyValue Class](#) [▸ 1129]

[TwinCAT.Ads Namespace](#) [▸ 179]

6.2.55.3 ResultAnyValue Methods

The [ResultAnyValue](#) [▸ 1129] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▸ 1128]	Sets the error state of this ResultAds [▸ 1116] (Inherited from ResultAds [▸ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultAnyValue Class](#) [▸ 1129]

[TwinCAT.Ads Namespace](#) [▸ 179]

6.2.56 ResultDeviceInfo Class

Ads Task Result for [DeviceInfo](#) [▸ 1134] requests (async operations).

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.ResultAds](#) [▸ 1116]

[TwinCAT.Ads.ResultDeviceInfo](#)

Namespace: [TwinCAT.Ads](#) [▸ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229








Syntax

C#








```
public class ResultDeviceInfo : ResultAds
```

The ResultDeviceInfo type exposes the following members.

Properties

	Name	Description
	DeviceInfo [▸ 1134]	Gets the device information (Task result)
	ErrorCode [▸ 1120]	Gets the ADS Error code bound to this Result [▸ 1116] object. (Inherited from ResultAds [▸ 1116] .)
	Failed [▸ 1121]	Gets a value indicating whether the ResultAds [▸ 1116] state is failed. (Inherited from ResultAds [▸ 1116] .)
	Invokeld [▸ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▸ 1116] .)
	Name [▸ 1134]	The name of the Device
	Succeeded [▸ 1121]	Gets a value indicating whether the ResultAds [▸ 1116] state is succeeded. (Inherited from ResultAds [▸ 1116] .)
	Version [▸ 1135]	The ADS Version of the the Device.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▸ 1128]	Sets the error state of this ResultAds [▸ 1116] (Inherited from ResultAds [▸ 1116] .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)


Reference







[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.56.1 ResultDeviceInfo Properties

The [ResultDeviceInfo \[▸ 1132\]](#) type exposes the following members.

Properties

	Name	Description
	DeviceInfo [▸ 1134]	Gets the device information (Task result)

	Name	Description
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	Name [▶ 1134]	The name of the Device
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)
	Version [▶ 1135]	The ADS Version of the the Device.

Reference

[ResultDeviceInfo Class](#) [[▶ 1132](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.56.1.1 ResultDeviceInfo.DeviceInfo Property

Gets the device information (Task result)

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DeviceInfo DeviceInfo { get; }
```

Property Value

Type: [DeviceInfo](#) [[▶ 801](#)]

The device information.

Reference

[ResultDeviceInfo Class](#) [[▶ 1132](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.56.1.2 ResultDeviceInfo.Name Property

The name of the Device

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Name { get; }
```

Property Value

Type: [String](#)

Reference

[ResultDeviceInfo Class \[► 1132\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.56.1.3 ResultDeviceInfo.Version Property

The ADS Version of the the Device.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsVersion Version { get; }
```

Property Value

Type: [AdsVersion \[► 743\]](#)

Reference








[ResultDeviceInfo Class \[► 1132\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.56.2 ResultDeviceInfo Methods

The [ResultDeviceInfo \[► 1132\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [► 1128]	Sets the error state of this ResultAds [► 1116] (Inherited from ResultAds [► 1116] .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultDeviceInfo Class \[► 1132\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.57 ResultHandle Class

Result object for asynchronous registering an ADS Handle via tasks.

Inheritance Hierarchy

System.Object

TwinCAT.Ads.ResultAds [[▶ 1116](#)]

TwinCAT.Ads.ResultHandle

Namespace: TwinCAT.Ads [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229







Syntax

C#






```
public class ResultHandle : ResultAds
```







The ResultHandle type exposes the following members.

Properties

	Name	Description
	Empty [▶ 1138]	Gets the empty ResultHandle object.
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Handle [▶ 1138]	Gets the registered handle
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Methods

	Name	Description
	CreateError(AdsError rCode) [▶ 1140]	Creates an Error Result.
	CreateError(AdsError rCode, UInt32) [▶ 1140]	Creates an Error Result.
	CreateSuccess(UInt32) [▶ 1141]	Creates a success result.
	CreateSuccess(UInt32, UInt32) [▶ 1142]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)

	Name	Description
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)



Operators

	Name	Description
	.(ResultHandle to ResultHandleAccess) [▶ 1142]	Performs an implicit conversion from ResultHandle to ResultHandleAccess [▶ 3224].

Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]








Also see about this

-  [ResultHandle.CreateError Method](#) [[▶ 1139](#)]
-  [ResultHandle.CreateSuccess Method](#) [[▶ 1141](#)]

6.2.57.1 ResultHandle Properties

The [ResultHandle](#) [[▶ 1136](#)] type exposes the following members.

Properties

	Name	Description
	Empty [▶ 1138]	Gets the empty ResultHandle [▶ 1136] object.
		
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Handle [▶ 1138]	Gets the registered handle
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Reference

[ResultHandle Class](#) [[▶ 1136](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.57.1.1 ResultHandle.Empty Property

Gets the empty [ResultHandle](#) [▸ 1136] object.

Namespace: [TwinCAT.Ads](#) [▸ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultHandle Empty { get; }
```

Property Value

Type: [ResultHandle](#) [▸ 1136]

The empty / unprocessed result.

Reference

[ResultHandle Class](#) [▸ 1136]

[TwinCAT.Ads Namespace](#) [▸ 179]

6.2.57.1.2 ResultHandle.Handle Property

Gets the registered handle

Namespace: [TwinCAT.Ads](#) [▸ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint Handle { get; }
```

Property Value

Type: [UInt32](#)

Reference



[ResultHandle Class](#) [▸ 1136]










[TwinCAT.Ads Namespace](#) [▸ 179]

6.2.57.2 ResultHandle Methods

The [ResultHandle](#) [▸ 1136] type exposes the following members.

Methods

	Name	Description
	CreateError(AdsError rCode) [▸ 1140]	Creates an Error Result.
	CreateError(AdsError rCode, UInt32) [▸ 1140]	Creates an Error Result.



	Name	Description
	CreateSuccess(UInt32) [▶ 1141]	Creates a success result.
	CreateSuccess(UInt32, UInt32) [▶ 1142]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultHandle Class](#) [[▶ 1136](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

-  [ResultHandle.CreateError Method](#) [[▶ 1139](#)]
-  [ResultHandle.CreateSuccess Method](#) [[▶ 1141](#)]

6.2.57.2.1 ResultHandle.CreateError Method

Overload List

	Name	Description
	CreateError(AdsErrorCode) [▶ 1140]	Creates an Error Result.
	CreateError(AdsErrorCode, UInt32) [▶ 1140]	Creates an Error Result.

Reference

[ResultHandle Class](#) [[▶ 1136](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

-  [TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.57.2.1.1 ResultHandle.CreateError Method (AdsErrorCode)

Creates an Error Result.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultHandle CreateError(
    AdsErrorCode errorCode
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
-----------	---

Return Value

Type: [ResultHandle](#) [[▶ 1136](#)]

ResultValue<T>.

Reference

[ResultHandle Class](#) [[▶ 1136](#)]

[CreateError Overload](#) [[▶ 1139](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.57.2.1.2 ResultHandle.CreateError Method (AdsErrorCode, UInt32)

Creates an Error Result.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultHandle CreateError(
    AdsErrorCode errorCode,
    uint invokeId
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
invokeld	Type: System.UInt32 The ADS request invoke identifier.

Return Value

Type: [ResultHandle](#) [[▶ 1136](#)]

ResultValue<T>.

Reference

[ResultHandle Class \[▸ 1136\]](#)

[CreateError Overload \[▸ 1139\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.57.2.2 ResultHandle.CreateSuccess Method

Overload List

	Name	Description
	CreateSuccess(UInt32) [▸ 1141]	Creates a success result.
	CreateSuccess(UInt32, UInt32) [▸ 1142]	Creates a success result.

Reference

[ResultHandle Class \[▸ 1136\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

Also see about this

 [TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.57.2.2.1 ResultHandle.CreateSuccess Method (UInt32)

Creates a success result.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultHandle CreateSuccess(
    uint handle
)
```

Parameters

handle	Type: System.UInt32 The handle.
--------	--

Return Value

Type: [ResultHandle \[▸ 1136\]](#)

ResultValue<T>.

Reference

[ResultHandle Class \[▸ 1136\]](#)

[CreateSuccess Overload \[▸ 1141\]](#)

[TwinCAT.Ads Namespace](#) [► 179]

6.2.57.2.2 ResultHandle.CreateSuccess Method (UInt32, UInt32)

Creates a success result.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultHandle CreateSuccess(
    uint handle,
    uint invokeId
)
```

Parameters

handle	Type: System.UInt32 The handle.
invokeld	Type: System.UInt32 The ADS request invoke identifier.

Return Value

Type: [ResultHandle](#) [► 1136]

ResultValue<T>.

Reference

[ResultHandle Class](#) [► 1136]

[CreateSuccess Overload](#) [► 1141]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.57.3 ResultHandle Type Conversions

The [ResultHandle](#) [► 1136] type exposes the following members.

Operators

	Name	Description
	.(ResultHandle to ResultHandleAccess) [► 1142]	Performs an implicit conversion from ResultHandle [► 1136] to ResultHandleAccess [► 3224].

Reference

[ResultHandle Class](#) [► 1136]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.57.3.1 ResultHandle . Conversion (ResultHandle to ResultHandleAccess)

Performs an implicit conversion from [ResultHandle](#) [► 1136] to [ResultHandleAccess](#) [► 3224].

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static implicit operator ResultHandleAccess (
    ResultHandle r
)
```

Parameters

r	Type: TwinCAT.Ads.ResultHandle [▶ 1136] The r.
---	---

Return Value

Type: [ResultHandleAccess \[▶ 3224\]](#)

The result of the conversion.

Reference

[ResultHandle Class \[▶ 1136\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.58 ResultRead Class

Asynchronous ADS Read result.

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.ResultAds \[▶ 1116\]](#)

[TwinCAT.Ads.ResultRead](#)

[TwinCAT.Ads.ResultReadBytes \[▶ 1150\]](#)

[TwinCAT.Ads.ResultReadWrite \[▶ 1163\]](#)

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229




Syntax

C#

```
public class ResultRead : ResultAds
```

The ResultRead type exposes the following members.

Properties

	Name	Description
	Empty [▶ 1145]	Gets the empty ResultRead object.
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116] .)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116] .)

	Name	Description
	Invokeld [▸ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▸ 1116] .)
	ReadBytes [▸ 1145]	Gets the number of Read bytes.
	Succeeded [▸ 1121]	Gets a value indicating whether the ResultAds [▸ 1116] state is succeeded. (Inherited from ResultAds [▸ 1116] .)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▸ 1128]	Sets the error state of this ResultAds [▸ 1116] (Inherited from ResultAds [▸ 1116] .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.58.1 ResultRead Properties

The [ResultRead \[▸ 1143\]](#) type exposes the following members.

Properties

	Name	Description
	Empty [▸ 1145]	Gets the empty ResultRead [▸ 1143] object.
	ErrorCode [▸ 1120]	Gets the ADS Error code bound to this Result [▸ 1116] object. (Inherited from ResultAds [▸ 1116] .)
	Failed [▸ 1121]	Gets a value indicating whether the ResultAds [▸ 1116] state is failed. (Inherited from ResultAds [▸ 1116] .)
	Invokeld [▸ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▸ 1116] .)
	ReadBytes [▸ 1145]	Gets the number of Read bytes.
	Succeeded [▸ 1121]	Gets a value indicating whether the ResultAds [▸ 1116] state is succeeded. (Inherited from ResultAds [▸ 1116] .)

Reference

[ResultRead Class \[▸ 1143\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.58.1.1 ResultRead.Empty Property

Gets the empty [ResultRead \[► 1143\]](#) object.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultRead Empty { get; }
```

Property Value

Type: [ResultRead \[► 1143\]](#)

The empty / unprocessed result.

Reference

[ResultRead Class \[► 1143\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.58.1.2 ResultRead.ReadBytes Property

Gets the number of Read bytes.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ReadBytes { get; }
```

Property Value

Type: [Int32](#)

Reference


[ResultRead Class \[► 1143\]](#)







[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.58.2 ResultRead Methods

The [ResultRead \[► 1143\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)

	Name	Description
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultRead Class](#) [[▶ 1143](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.59 ResultReadAdsState Class

Result object for asynchronous reading AdsStates via tasks.

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.ResultAds](#) [[▶ 1116](#)]

[TwinCAT.Ads.ResultReadAdsState](#)

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229








Syntax

C#










```
public class ResultReadAdsState : ResultAds
```

The ResultReadAdsState type exposes the following members.

Properties

	Name	Description
 	Empty [▶ 1148]	Gets the empty ResultReadAdsState object.
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	State [▶ 1148]	Gets the ADS state.
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Methods

	Name	Description
	CreateError [▸ 1149]	Creates an Error Result.
	CreateSuccess [▸ 1149]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▸ 1128]	Sets the error state of this ResultAds [▸ 1116] (Inherited from ResultAds [▸ 1116] .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)







Reference

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.59.1 ResultReadAdsState Properties

The [ResultReadAdsState \[▸ 1146\]](#) type exposes the following members.

Properties

	Name	Description
	Empty [▸ 1148]	Gets the empty ResultReadAdsState [▸ 1146] object.
	ErrorCode [▸ 1120]	Gets the ADS Error code bound to this Result [▸ 1116] object. (Inherited from ResultAds [▸ 1116] .)
	Failed [▸ 1121]	Gets a value indicating whether the ResultAds [▸ 1116] state is failed. (Inherited from ResultAds [▸ 1116] .)
	Invokeld [▸ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▸ 1116] .)
	State [▸ 1148]	Gets the ADS state.
	Succeeded [▸ 1121]	Gets a value indicating whether the ResultAds [▸ 1116] state is succeeded. (Inherited from ResultAds [▸ 1116] .)

Reference

[ResultReadAdsState Class \[▸ 1146\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.59.1.1 ResultReadAdsState.Empty Property

Gets the empty [ResultReadAdsState](#) [▶ 1146] object.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadAdsState Empty { get; }
```

Property Value

Type: [ResultReadAdsState](#) [▶ 1146]

The empty / unprocessed result.

Reference

[ResultReadAdsState Class](#) [▶ 1146]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.59.1.2 ResultReadAdsState.State Property

Gets the ADS state.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsState State { get; }
```

Property Value

Type: [AdsState](#) [▶ 729]

Reference



[ResultReadAdsState Class](#) [▶ 1146]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.59.2 ResultReadAdsState Methods

The [ResultReadAdsState](#) [▶ 1146] type exposes the following members.

Methods

	Name	Description
	CreateError [▶ 1149]	Creates an Error Result.
	CreateSuccess [▶ 1149]	Creates a success result.

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultReadAdsState Class](#) [[▶ 1146](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.59.2.1 ResultReadAdsState.CreateError Method

Creates an Error Result.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadAdsState CreateError(
    AdsErrorCode errorCode
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
-----------	---

Return Value

Type: [ResultReadAdsState](#) [[▶ 1146](#)]

ResultValue<T>.

Reference

[ResultReadAdsState Class](#) [[▶ 1146](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.59.2.2 ResultReadAdsState.CreateSuccess Method

Creates a success result.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadAdsState CreateSuccess(
    AdsState state
)
```

Parameters

state	Type: TwinCAT.Ads.AdsState [▶ 729] The state.
-------	--

Return Value

Type: [ResultReadAdsState](#) [▶ 1146]
ResultValue<T>.

Reference

[ResultReadAdsState Class](#) [▶ 1146]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.60 ResultReadBytes Class

ADS Task Result returning Read data for async Read operations.

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.ResultAds](#) [▶ 1116]

[TwinCAT.Ads.ResultRead](#) [▶ 1143]

[TwinCAT.Ads.ResultReadBytes](#)

[TwinCAT.Ads.ResultReadWriteBytes](#) [▶ 1170]

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229







Syntax

C#

```
public class ResultReadBytes : ResultRead
```

The ResultReadBytes type exposes the following members.

Properties

	Name	Description
	Data [▶ 1152]	The read bytes as continuous region of memory.
	Empty [▶ 1152]	Gets the empty ResultReadBytes object.
		
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)

	Name	Description
	ReadBytes [▶ 1145]	Gets the number of Read bytes. (Inherited from ResultRead [▶ 1143].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Methods

	Name	Description
	CreateError(AdsErrorCode) [▶ 1154]	Creates an error result.
	CreateError(AdsErrorCode, UInt32) [▶ 1155]	Creates an error result.
	CreateSuccess(ReadOnlyMemory.Byte) [▶ 1156]	Creates a success result.
	CreateSuccess(ReadOnlyMemory.Byte, UInt32) [▶ 1156]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]









Also see about this

- [ResultReadBytes.CreateError Method](#) [[▶ 1154](#)]
- [ResultReadBytes.CreateSuccess Method](#) [[▶ 1155](#)]

6.2.60.1 ResultReadBytes Properties

The [ResultReadBytes](#) [[▶ 1150](#)] type exposes the following members.

Properties

	Name	Description
	Data [▶ 1152]	The read bytes as continuous region of memory.
	Empty [▶ 1152]	Gets the empty ResultReadBytes [▶ 1150] object.
		
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116] .)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116] .)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116] .)
	ReadBytes [▶ 1145]	Gets the number of Read bytes. (Inherited from ResultRead [▶ 1143] .)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116] .)

Reference

[ResultReadBytes Class \[▶ 1150\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.60.1.1 ResultReadBytes.Data Property

The read bytes as continuous region of memory.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public ReadOnlyMemory<byte> Data { get; }
```

Property Value

Type: [ReadOnlyMemory.Byte](#).

Reference

[ResultReadBytes Class \[▶ 1150\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.60.1.2 ResultReadBytes.Empty Property

Gets the empty [ResultReadBytes \[▶ 1150\]](#) object.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadBytes Empty { get; }
```

Property Value

Type: [ResultReadBytes](#) [▶ 1150]
 The empty / unprocessed result.

Reference

[ResultReadBytes Class](#) [▶ 1150]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.60.2 ResultReadBytes Methods

The [ResultReadBytes](#) [▶ 1150] type exposes the following members.

Methods

	Name	Description
	CreateError(AdsError rCode) [▶ 1154]	Creates an error result.
	CreateError(AdsError rCode, UInt32) [▶ 1155]	Creates an error result.
	CreateSuccess(ReadOnlyMemory<Byte>.) [▶ 1156]	Creates a success result.
	CreateSuccess(ReadOnlyMemory<Byte>, UInt32) [▶ 1156]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultReadBytes Class](#) [▶ 1150]

[TwinCAT.Ads Namespace \[▸ 179\]](#)

Also see about this

- [ResultReadBytes.CreateError Method \[▸ 1154\]](#)
- [ResultReadBytes.CreateSuccess Method \[▸ 1155\]](#)

6.2.60.2.1 ResultReadBytes.CreateError Method

Overload List

	Name	Description
	CreateError(AdsError rCode) [▸ 1154]	Creates an error result.
	CreateError(AdsError rCode, UInt32) [▸ 1155]	Creates an error result.

Reference

[ResultReadBytes Class \[▸ 1150\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

Also see about this

- [TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.60.2.1.1 ResultReadBytes.CreateError Method (AdsErrorCode)

Creates an error result.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadBytes CreateError(
    AdsErrorCode error
)
```

Parameters

error	Type: TwinCAT.Ads.AdsErrorCode [▸ 664] The error.
-------	--

Return Value

Type: [ResultReadBytes \[▸ 1150\]](#)
ResultReadBytes.

Reference

[ResultReadBytes Class \[▸ 1150\]](#)

[CreateError Overload \[▸ 1154\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.60.2.1.2 ResultReadBytes.CreateError Method (AdsErrorCode, UInt32)

Creates an error result.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadBytes CreateError(
    AdsErrorCode error,
    uint invokeId
)
```

Parameters

error	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error.
invokeld	Type: System.UInt32 The ADS request invoke identifier.

Return Value

Type: [ResultReadBytes \[▶ 1150\]](#)

ResultReadBytes.

Reference

[ResultReadBytes Class \[▶ 1150\]](#)

[CreateError Overload \[▶ 1154\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.60.2.2 ResultReadBytes.CreateSuccess Method

Overload List

	Name	Description
	CreateSuccess(ReadOnlyMemory<Byte>) [▶ 1156]	Creates a success result.
	CreateSuccess(ReadOnlyMemory<Byte>, UInt32) [▶ 1156]	Creates a success result.

Reference

[ResultReadBytes Class \[▶ 1150\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

Also see about this

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.60.2.2.1 ResultReadBytes.CreateSuccess Method (ReadOnlyMemory.Byte.)

Creates a success result.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadBytes CreateSuccess(
    ReadOnlyMemory<byte> data
)
```

Parameters

data	Type: System.ReadOnlyMemory.Byte. The data.
------	--

Return Value

Type: [ResultReadBytes](#) [▶ 1150]

ResultReadBytes.

Reference

[ResultReadBytes Class](#) [▶ 1150]

[CreateSuccess Overload](#) [▶ 1155]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.60.2.2.2 ResultReadBytes.CreateSuccess Method (ReadOnlyMemory.Byte., UInt32)

Creates a success result.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadBytes CreateSuccess(
    ReadOnlyMemory<byte> data,
    uint invokeId
)
```

Parameters

data	Type: System.ReadOnlyMemory.Byte. The data.
invokeld	Type: System.UInt32 The ADS request invoke identifier.

Return Value

Type: [ResultReadBytes](#) [[▶ 1150](#)]
 ResultReadBytes.

Reference

- [ResultReadBytes Class](#) [[▶ 1150](#)]
- [CreateSuccess Overload](#) [[▶ 1155](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.61 ResultReadDeviceState Class

Result object for asynchronous ADS ReadDeviceState tasks.

Inheritance Hierarchy

[System.Object](#)
 [TwinCAT.Ads.ResultAds](#) [[▶ 1116](#)]
 TwinCAT.Ads.ResultReadDeviceState

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229








Syntax

C#


```
public class ResultReadDeviceState : ResultAds
```











The ResultReadDeviceState type exposes the following members.

Properties

	Name	Description
	Empty [▶ 1159]	Gets the empty ResultReadDeviceState object.
		
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	State [▶ 1159]	The Device state.
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Methods



	Name	Description
	CreateError(AdsError rCode) [▶ 1161]	Creates an Error Result.

	Name	Description
	CreateError(AdsErrorCode, UInt32) [▶ 1161]	Creates an Error Result.
	CreateSuccess(StateInfo) [▶ 1162]	Creates a success result.
	CreateSuccess(StateInfo, UInt32) [▶ 1163]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]







Also see about this

-  [ResultReadDeviceState.CreateError Method](#) [[▶ 1160](#)]
-  [ResultReadDeviceState.CreateSuccess Method](#) [[▶ 1162](#)]

6.2.61.1 ResultReadDeviceState Properties

The [ResultReadDeviceState](#) [[▶ 1157](#)] type exposes the following members.

Properties

	Name	Description
	Empty [▶ 1159]	Gets the empty ResultReadDeviceState [▶ 1157] object.
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	State [▶ 1159]	The Device state.
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Reference

[ResultReadDeviceState Class \[► 1157\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.61.1.1 ResultReadDeviceState.Empty Property

Gets the empty [ResultReadDeviceState \[► 1157\]](#) object.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadDeviceState Empty { get; }
```

Property Value

Type: [ResultReadDeviceState \[► 1157\]](#)

The empty / unprocessed result.

Reference

[ResultReadDeviceState Class \[► 1157\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.61.1.2 ResultReadDeviceState.State Property

The Device state.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public StateInfo State { get; }
```

Property Value

Type: [StateInfo \[► 1200\]](#)

Reference












[ResultReadDeviceState Class \[► 1157\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.61.2 ResultReadDeviceState Methods

The [ResultReadDeviceState \[► 1157\]](#) type exposes the following members.

Methods



	Name	Description
	CreateError(AdsErrorCode) [► 1161]	Creates an Error Result.
	CreateError(AdsErrorCode, UInt32) [► 1161]	Creates an Error Result.
	CreateSuccess(StateInfo) [► 1162]	Creates a success result.
	CreateSuccess(StateInfo, UInt32) [► 1163]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [► 1128]	Sets the error state of this ResultAds [► 1116] (Inherited from ResultAds [► 1116] .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultReadDeviceState Class \[► 1157\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

Also see about this

-  [ResultReadDeviceState.CreateError Method \[► 1160\]](#)
-  [ResultReadDeviceState.CreateSuccess Method \[► 1162\]](#)

6.2.61.2.1 ResultReadDeviceState.CreateError Method**Overload List**


	Name	Description
	CreateError(AdsErrorCode) [► 1161]	Creates an Error Result.
	CreateError(AdsErrorCode, UInt32) [► 1161]	Creates an Error Result.

Reference

[ResultReadDeviceState Class \[► 1157\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

Also see about this

 [TwinCAT.Ads Namespace \[► 179\]](#)

6.2.61.2.1.1 ResultReadDeviceState.CreateError Method (AdsErrorCode)

Creates an Error Result.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadDeviceState CreateError(  
    AdsErrorCode errorCode  
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [► 664] The error code.
-----------	---

Return Value

Type: [ResultReadDeviceState \[► 1157\]](#)

ResultValue<T>.

Reference

[ResultReadDeviceState Class \[► 1157\]](#)

[CreateError Overload \[► 1160\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.61.2.1.2 ResultReadDeviceState.CreateError Method (AdsErrorCode, UInt32)

Creates an Error Result.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadDeviceState CreateError(  
    AdsErrorCode errorCode,  
    uint invokeId  
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [► 664] The error code.
-----------	---

invokeld	Type: System.UInt32 The ADS request invoke identifier.
----------	---

Return Value

Type: [ResultReadDeviceState](#) [[▶ 1157](#)]
ResultValue<T>.

Reference

[ResultReadDeviceState Class](#) [[▶ 1157](#)]

[CreateError Overload](#) [[▶ 1160](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.61.2.2 ResultReadDeviceState.CreateSuccess Method**Overload List**


	Name	Description
	CreateSuccess(StateInfo) [▶ 1162]	Creates a success result.
	CreateSuccess(StateInfo, UInt32) [▶ 1163]	Creates a success result.

Reference

[ResultReadDeviceState Class](#) [[▶ 1157](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

 [TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.61.2.2.1 ResultReadDeviceState.CreateSuccess Method (StateInfo)

Creates a success result.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static ResultReadDeviceState CreateSuccess(
    StateInfo state
)
```

Parameters

state	Type: TwinCAT.Ads.StateInfo [▶ 1200] The state.
-------	--

Return Value

Type: [ResultReadDeviceState](#) [► 1157]
ResultValue<T>.

Reference

[ResultReadDeviceState Class](#) [► 1157]

[CreateSuccess Overload](#) [► 1162]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.61.2.2 ResultReadDeviceState.CreateSuccess Method (StateInfo, UInt32)

Creates a success result.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadDeviceState CreateSuccess (  
    StateInfo state,  
    uint invokeId  
)
```

Parameters

state	Type: TwinCAT.Ads.StateInfo [► 1200] The state.
invokeld	Type: System.UInt32 The ADS request invoke identifier.

Return Value

Type: [ResultReadDeviceState](#) [► 1157]
ResultValue<T>.

Reference

[ResultReadDeviceState Class](#) [► 1157]

[CreateSuccess Overload](#) [► 1162]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.62 ResultReadWrite Class

Result object for asynchronous ADS ReadWrite tasks.

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.ResultAds](#) [► 1116]

[TwinCAT.Ads.ResultRead](#) [► 1143]

 TwinCAT.Ads.ResultReadWrite

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229







Syntax

C#












```
public class ResultReadWrite : ResultRead
```

The ResultReadWrite type exposes the following members.

Properties

	Name	Description
	Empty [▶ 1165]	Gets the empty ResultReadWrite object.
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	ReadBytes [▶ 1145]	Gets the number of Read bytes. (Inherited from ResultRead [▶ 1143].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)



Methods

	Name	Description
	CreateError(AdsError rCode) [▶ 1167]	Creates an error result;
	CreateError(AdsError rCode, UInt32) [▶ 1167]	Creates an error result;
	CreateSuccess(Int32) [▶ 1168]	Creates a success result.
	CreateSuccess(Int32, UInt32) [▶ 1169]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[TwinCAT.Ads Namespace \[► 179\]](#)








Also see about this

-  [ResultReadWrite.CreateError Method \[► 1166\]](#)
-  [ResultReadWrite.CreateSuccess Method \[► 1168\]](#)

6.2.62.1 ResultReadWrite Properties

The [ResultReadWrite \[► 1163\]](#) type exposes the following members.

Properties

	Name	Description
	Empty [► 1165]	Gets the empty ResultReadWrite [► 1163] object.
		
	ErrorCode [► 1120]	Gets the ADS Error code bound to this Result [► 1116] object. (Inherited from ResultAds [► 1116] .)
	Failed [► 1121]	Gets a value indicating whether the ResultAds [► 1116] state is failed. (Inherited from ResultAds [► 1116] .)
	Invokeld [► 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [► 1116] .)
	ReadBytes [► 1145]	Gets the number of Read bytes. (Inherited from ResultRead [► 1143] .)
	Succeeded [► 1121]	Gets a value indicating whether the ResultAds [► 1116] state is succeeded. (Inherited from ResultAds [► 1116] .)

Reference

[ResultReadWrite Class \[► 1163\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.62.1.1 ResultReadWrite.Empty Property

Gets the empty [ResultReadWrite \[► 1163\]](#) object.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadWrite Empty { get; }
```

Property Value

Type: [ResultReadWrite \[► 1163\]](#)

The empty / unprocessed result.

Reference












[ResultReadWrite Class \[► 1163\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.62.2 ResultReadWrite Methods

The [ResultReadWrite](#) [► 1163] type exposes the following members.

Methods



	Name	Description
	CreateError(AdsErrorCode) [► 1167]	Creates an error result;
	CreateError(AdsErrorCode, UInt32) [► 1167]	Creates an error result;
	CreateSuccess(Int32) [► 1168]	Creates a success result.
	CreateSuccess(Int32, UInt32) [► 1169]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [► 1128]	Sets the error state of this ResultAds [► 1116] (Inherited from ResultAds [► 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultReadWrite Class](#) [► 1163]

[TwinCAT.Ads Namespace](#) [► 179]

Also see about this

-  [ResultReadWrite.CreateError Method](#) [► 1166]
-  [ResultReadWrite.CreateSuccess Method](#) [► 1168]

6.2.62.2.1 ResultReadWrite.CreateError Method

Overload List

	Name	Description
	CreateError(AdsErrorCode) [► 1167]	Creates an error result;

	Name	Description
	CreateError(AdsErrorCode, UInt32) [▶ 1167]	Creates an error result;

Reference

[ResultReadWrite Class](#) [▶ 1163]

[TwinCAT.Ads Namespace](#) [▶ 179]

Also see about this

 [TwinCAT.Ads Namespace](#) [▶ 179]

6.2.62.2.1.1 ResultReadWrite.CreateError Method (AdsErrorCode)

Creates an error result;

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadWrite CreateError(
    AdsErrorCode errorCode
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
-----------	---

Return Value

Type: [ResultReadWrite](#) [▶ 1163]

ResultWrite.

Reference

[ResultReadWrite Class](#) [▶ 1163]

[CreateError Overload](#) [▶ 1166]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.62.2.1.2 ResultReadWrite.CreateError Method (AdsErrorCode, UInt32)

Creates an error result;

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadWrite CreateError(
    AdsErrorCode errorCode,
    uint invokeId
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
invokeld	Type: System.UInt32 The ADS request invoke identifier.

Return Value

Type: [ResultReadWrite](#) [[▶ 1163](#)]
ResultWrite.

Reference

[ResultReadWrite Class](#) [[▶ 1163](#)]

[CreateError Overload](#) [[▶ 1166](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.62.2.2 ResultReadWrite.CreateSuccess Method

Overload List

	Name	Description
	CreateSuccess(Int32) [▶ 1168]	Creates a success result.
	CreateSuccess(Int32, UInt32) [▶ 1169]	Creates a success result.

Reference

[ResultReadWrite Class](#) [[▶ 1163](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

 [TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.62.2.2.1 ResultReadWrite.CreateSuccess Method (Int32)

Creates a success result.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadWrite CreateSuccess(  
    int readBytes  
)
```

Parameters

readBytes	Type: System.Int32
-----------	------------------------------------

Return Value

Type: [ResultReadWrite](#) [[▶ 1163](#)]
ResultWrite.

Reference

[ResultReadWrite Class](#) [[▶ 1163](#)]

[CreateSuccess Overload](#) [[▶ 1168](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.62.2.2 ResultReadWrite.CreateSuccess Method (Int32, UInt32)

Creates a success result.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadWrite CreateSuccess(  
    int readBytes,  
    uint invokeId  
)
```

Parameters

readBytes	Type: System.Int32 The number of read bytes.
invokeld	Type: System.UInt32 The ADS request invoke identifier.

Return Value

Type: [ResultReadWrite](#) [[▶ 1163](#)]
ResultWrite.

Reference

[ResultReadWrite Class](#) [[▶ 1163](#)]

[CreateSuccess Overload](#) [[▶ 1168](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.63 ResultReadWriteBytes Class

Result object for asynchronous ADS ReadWrite tasks.

Inheritance Hierarchy

System.Object

TwinCAT.Ads.ResultAds [[▶ 1116](#)]

TwinCAT.Ads.ResultRead [[▶ 1143](#)]

TwinCAT.Ads.ResultReadBytes [[▶ 1150](#)]

TwinCAT.Ads.ResultReadWriteBytes

Namespace: TwinCAT.Ads [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229








Syntax

C#




```
public class ResultReadWriteBytes : ResultReadBytes
```









The ResultReadWriteBytes type exposes the following members.

Properties

	Name	Description
	Data [▶ 1152]	The read bytes as continuous region of memory. (Inherited from ResultReadBytes [▶ 1150].)
 S	Empty [▶ 1172]	Gets the empty ResultReadWriteBytes object.
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	ReadBytes [▶ 1145]	Gets the number of Read bytes. (Inherited from ResultRead [▶ 1143].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Methods



	Name	Description
	CreateError(AdsError) [▶ 1173]	Creates an error result;
	CreateError(AdsError, UInt32) [▶ 1174]	Creates an error result;
	CreateSuccess(ReadOnlyMemory.Byte) [▶ 1175]	Creates a success result.

	Name	Description
	CreateSuccess(ReadOnlyMemory<Byte>, UInt32) [▶ 1176]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]








Also see about this

-  [ResultReadWriteBytes.CreateError](#) Method [[▶ 1173](#)]
-  [ResultReadWriteBytes.CreateSuccess](#) Method [[▶ 1175](#)]

6.2.63.1 ResultReadWriteBytes Properties

The [ResultReadWriteBytes](#) [[▶ 1170](#)] type exposes the following members.

Properties

	Name	Description
	Data [▶ 1152]	The read bytes as continuous region of memory. (Inherited from ResultReadBytes [▶ 1150].)
	Empty [▶ 1172]	Gets the empty ResultReadWriteBytes [▶ 1170] object.
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	ReadBytes [▶ 1145]	Gets the number of Read bytes. (Inherited from ResultRead [▶ 1143].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Reference

[ResultReadWriteBytes Class](#) [[▶ 1170](#)]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.63.1.1 ResultReadWriteBytes.Empty Property

Gets the empty [ResultReadWriteBytes](#) [▶ 1170] object.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadWriteBytes Empty { get; }
```

Property Value

Type: [ResultReadWriteBytes](#) [▶ 1170]

The empty / unprocessed result.

Reference









[ResultReadWriteBytes Class](#) [▶ 1170]




[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.63.2 ResultReadWriteBytes Methods

The [ResultReadWriteBytes](#) [▶ 1170] type exposes the following members.

Methods

	Name	Description
	CreateError(AdsError rCode) [▶ 1173]	Creates an error result;
	CreateError(AdsError rCode, UInt32) [▶ 1174]	Creates an error result;
	CreateSuccess(ReadOnlyMemory<Byte>) [▶ 1175]	Creates a success result.
	CreateSuccess(ReadOnlyMemory<Byte>, UInt32) [▶ 1176]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)



	Name	Description
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultReadWriteBytes Class](#) [[▶ 1170](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

-  [ResultReadWriteBytes.CreateError Method](#) [[▶ 1173](#)]
-  [ResultReadWriteBytes.CreateSuccess Method](#) [[▶ 1175](#)]

6.2.63.2.1 ResultReadWriteBytes.CreateError Method

Overload List

	Name	Description
	CreateError(AdsError rCode) [▶ 1173]	Creates an error result;
	CreateError(AdsError rCode, UInt32) [▶ 1174]	Creates an error result;

Reference

[ResultReadWriteBytes Class](#) [[▶ 1170](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

-  [TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.63.2.1.1 ResultReadWriteBytes.CreateError Method (AdsErrorCode)

Creates an error result;

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadWriteBytes CreateError(
    AdsErrorCode errorCode
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
-----------	---

Return Value

Type: [ResultReadWriteBytes](#) [▶ 1170]
ResultWrite.

Reference

[ResultReadWriteBytes Class](#) [▶ 1170]

[CreateError Overload](#) [▶ 1173]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.63.2.1.2 ResultReadWriteBytes.CreateError Method (AdsErrorCode, UInt32)

Creates an error result;

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static ResultReadWriteBytes CreateError(
    AdsErrorCode errorCode,
    uint invokeId
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
invokeld	Type: System.UInt32 The ADS request invoke identifier.

Return Value

Type: [ResultReadWriteBytes](#) [▶ 1170]
ResultWrite.

Reference

[ResultReadWriteBytes Class](#) [▶ 1170]

[CreateError Overload](#) [▶ 1173]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.63.2.2 ResultReadWriteBytes.CreateSuccess Method

Overload List


	Name	Description
	CreateSuccess(ReadOnlyMemory<Byte>) [▶ 1175]	Creates a success result.
	CreateSuccess(ReadOnlyMemory<Byte>, UInt32) [▶ 1176]	Creates a success result.

Reference

[ResultReadWriteBytes Class](#) [[▶ 1170](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

Also see about this

 [TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.63.2.2.1 ResultReadWriteBytes.CreateSuccess Method (ReadOnlyMemory<Byte>)

Creates a success result.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadWriteBytes CreateSuccess(  
    ReadOnlyMemory<byte> data  
)
```

Parameters

data	Type: System.ReadOnlyMemory<Byte> .
------	---

Return Value

Type: [ResultReadWriteBytes](#) [[▶ 1170](#)]

ResultWrite.

Reference

[ResultReadWriteBytes Class](#) [[▶ 1170](#)]

[CreateSuccess Overload](#) [[▶ 1175](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.63.2.2 ResultReadWriteBytes.CreateSuccess Method (ReadOnlyMemory.Byte., UInt32)

Creates a success result.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadWriteBytes CreateSuccess(
    ReadOnlyMemory<byte> data,
    uint invokeId
)
```

Parameters

data	Type: System.ReadOnlyMemory.Byte. The data.
invokeId	Type: System.UInt32 The ADS request invoke identifier.

Return Value

Type: [ResultReadWriteBytes](#) [► 1170]

ResultWrite.

Reference

[ResultReadWriteBytes Class](#) [► 1170]

[CreateSuccess Overload](#) [► 1175]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.64 ResultRpcMethod Class

Class representing a result of an asynchronous RpcMethod call. Implements the [ResultAds](#) [► 1116]

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.ResultAds](#) [► 1116]

[TwinCAT.Ads.ResultRpcMethod](#)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#








```
public class ResultRpcMethod : ResultAds
```

The ResultRpcMethod type exposes the following members.










Constructors

	Name	Description
	ResultRpcMethod [▶ 1178]	Initializes a new instance of the ResultRpcMethod struct.

Properties

	Name	Description
	Empty [▶ 1179]	Gets the Empty Result (initialized to None [▶ 664] and default ReturnValue [▶ 1179] (NULL).
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116] .)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116] .)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116] .)
	OutValues [▶ 1179]	Gets the output parameter values.
	ReturnValue [▶ 1179]	The (optional) return value of the RPC Method.
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116] .)

Methods

	Name	Description
	CreateError [▶ 1180]	Creates an error result.
	CreateSuccess [▶ 1181]	Creates a success result
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116] .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Remarks

Optionally this Result holds an [ReturnValue](#) [\[▶ 1179\]](#) if the RpcCall is not void.

Reference

[TwinCAT.Ads Namespace](#) [\[▶ 179\]](#)

[TwinCAT.Ads.ResultAds](#) [[▶ 1116](#)]

6.2.64.1 ResultRpcMethod Constructor

Initializes a new instance of the [ResultRpcMethod](#) [[▶ 1176](#)] struct.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ResultRpcMethod(
    AdsErrorCode errorCode,
    Object? returnValue,
    Object[]? outParameters,
    uint invokeId
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
returnValue	Type: System.Object The value.
outParameters	Type: .System.Object . The out parameters.
invokeld	Type: System.UInt32 The ADS request invoke identifier or 0

Reference







[ResultRpcMethod Class](#) [[▶ 1176](#)]


[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.64.2 ResultRpcMethod Properties

The [ResultRpcMethod](#) [[▶ 1176](#)] type exposes the following members.

Properties

	Name	Description
	Empty [▶ 1179]	Gets the Empty Result (initialized to None [▶ 664] and default ReturnValue [▶ 1179] (NULL).
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	OutValues [▶ 1179]	Gets the output parameter values.
	ReturnValue [▶ 1179]	The (optional) return value of the RPC Method.

	Name	Description
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Reference

[ResultRpcMethod Class](#) [[▶ 1176](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.64.2.1 ResultRpcMethod.Empty Property

Gets the Empty Result (initialized to [None](#) [[▶ 664](#)] and default [ReturnValue](#) [[▶ 1179](#)] (NULL)).

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultRpcMethod Empty { get; }
```

Property Value

Type: [ResultRpcMethod](#) [[▶ 1176](#)]

The empty.

Reference

[ResultRpcMethod Class](#) [[▶ 1176](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.64.2.2 ResultRpcMethod.OutValues Property

Gets the output parameter values.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object[]? OutValues { get; }
```

Property Value

Type: [.Object](#).

The output parameters.

Reference

[ResultRpcMethod Class](#) [[▶ 1176](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.64.2.3 ResultRpcMethod.ReturnValue Property

The (optional) return value of the RPC Method.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object? ReturnValue { get; }
```

Property Value

Type: [Object](#)

Reference










[ResultRpcMethod Class](#) [► 1176]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.64.3 ResultRpcMethod Methods

The [ResultRpcMethod](#) [► 1176] type exposes the following members.

Methods

	Name	Description
	CreateError [► 1180]	Creates an error result.
	CreateSuccess [► 1181]	Creates a success result
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [► 1128]	Sets the error state of this ResultAds [► 1116] (Inherited from ResultAds [► 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultRpcMethod Class](#) [► 1176]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.64.3.1 ResultRpcMethod.CreateError Method

Creates an error result.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultRpcMethod CreateError(
    AdsErrorCode error
)
```

Parameters

error	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error.
-------	--

Return Value

Type: [ResultRpcMethod](#) [▶ 1176]
ResultRpcMethod.

Reference

[ResultRpcMethod Class](#) [▶ 1176]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.64.3.2 ResultRpcMethod.CreateSuccess Method

Creates a success result

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultRpcMethod CreateSuccess(
    Object? returnValue,
    Object[]? outParameters
)
```

Parameters

returnValue	Type: System.Object The return value.
outParameters	Type: .System.Object . The out parameters.

Return Value

Type: [ResultRpcMethod](#) [▶ 1176]
ResultRpcMethod.

Reference

[ResultRpcMethod Class](#) [▶ 1176]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.65 ResultValue.TValue. Class

ADS Result object returning a generic value result (TValue) (asynchronous read). Implements the [ResultAds](#) [▶ 1116]

Inheritance Hierarchy

System.Object

- [TwinCAT.Ads.ResultAds](#) [[▶ 1116](#)]
- [TwinCAT.Ads.ResultValue.TValue](#).
- [TwinCAT.Ads.ResultAnyValue](#) [[▶ 1129](#)]
- [TwinCAT.Ads.ResultValue2.I, V.](#) [[▶ 1234](#)]
- [TwinCAT.TypeSystem.ResultDataTypes](#) [[▶ 2872](#)]
- [TwinCAT.TypeSystem.ResultSymbols.T.](#) [[▶ 2884](#)]

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#







```
public class ResultValue<TValue> : ResultAds
```

Type Parameters






TValue	The type of the result value.
--------	-------------------------------





The ResultValue.TValue. type exposes the following members.

Properties

	Name	Description
	Empty [▶ 1185]	Gets the Empty result initialized to NoError [▶ 664] and the default of TValue.
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)
	Value [▶ 1185]	The value object.

Methods

	Name	Description
	CreateError [▶ 1186]	Creates an Error Result.
	CreateSuccess [▶ 1187]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)

	Name	Description
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Remarks

The [ResultValue.TValue](#) type is used in generic ADS read operations, where a value is returned.



Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[TwinCAT.Ads.ResultAds](#) [[▶ 1116](#)]

6.2.65.1 ResultValue.TValue. Constructor

Overload List

	Name	Description
	ResultValue.TValue . (TValue) [▶ 1183]	Initializes a new 'Succeeded' instance of the ResultValue.TValue . [▶ 1181] class.
	ResultValue.TValue . (AdsErrorCode, TValue) [▶ 1184]	Initializes a new instance of the ResultValue class.

Reference

[ResultValue.TValue](#). Class [[▶ 1181](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.65.1.1 ResultValue.TValue. Constructor (TValue)

Initializes a new 'Succeeded' instance of the [ResultValue.TValue](#). [[▶ 1181](#)] class.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.116+a71ced3

Syntax

C#

```
public ResultValue(
    TValue value
)
```

Parameters

value Type: [TValue](#) [[▶ 1181](#)]
The value.

Reference

[ResultValue.TValue](#). Class [[▶ 1181](#)]

[ResultValue.TValue. Overload \[► 1183\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.65.1.2 ResultValue.TValue. Constructor (AdsErrorCode, TValue)

Initializes a new instance of the ResultValue class.

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public ResultValue(
    AdsErrorCode errorCode,
    TValue value
)
```

Parameters

errorCode Type: [TwinCAT.Ads.AdsErrorCode \[► 664\]](#)
The error code.

value Type: [TValue \[► 1181\]](#)
The value.

Reference

[ResultValue.TValue. Class \[► 1181\]](#)







[ResultValue.TValue. Overload \[► 1183\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.65.2 ResultValue.TValue. Properties

The [ResultValue.TValue. \[► 1181\]](#) generic type exposes the following members.

Properties

	Name	Description
	Empty [► 1185]	Gets the Empty result initialized to NoError [► 664] and the default of TValue.
	ErrorCode [► 1120]	Gets the ADS Error code bound to this Result [► 1116] object. (Inherited from ResultAds [► 1116] .)
	Failed [► 1121]	Gets a value indicating whether the ResultAds [► 1116] state is failed. (Inherited from ResultAds [► 1116] .)
	Invokeld [► 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [► 1116] .)
	Succeeded [► 1121]	Gets a value indicating whether the ResultAds [► 1116] state is succeeded. (Inherited from ResultAds [► 1116] .)
	Value [► 1185]	The value object.

Reference[ResultValue.TValue. Class \[▶ 1181\]](#)[TwinCAT.Ads Namespace \[▶ 179\]](#)**6.2.65.2.1 ResultValue.TValue..Empty Property**

Gets the Empty result initialized to [NoError \[▶ 664\]](#) and the default of TValue.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static ResultValue<TValue> Empty { get; }
```

Property Value

Type: [ResultValue \[▶ 1181\].TValue \[▶ 1181\]](#).
The empty.

Reference[ResultValue.TValue. Class \[▶ 1181\]](#)[TwinCAT.Ads Namespace \[▶ 179\]](#)**6.2.65.2.2 ResultValue.TValue..Value Property**

The value object.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public TValue Value { get; }
```


Property Value









Type: [TValue \[▶ 1181\]](#)

Reference[ResultValue.TValue. Class \[▶ 1181\]](#)[TwinCAT.Ads Namespace \[▶ 179\]](#)**6.2.65.3 ResultValue.TValue. Methods**

The [ResultValue.TValue. \[▶ 1181\]](#) generic type exposes the following members.

Methods

	Name	Description
	CreateError [▶ 1186]	Creates an Error Result.

	Name	Description
	CreateSuccess [▶ 1187]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultValue.TValue. Class](#) [▶ 1181]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.65.3.1 ResultValue.TValue..CreateError Method

Creates an Error Result.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultValue<TValue> CreateError(
    AdsErrorCode errorCode
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
-----------	---

Return Value

Type: [ResultValue](#) [▶ 1181].[TValue](#) [▶ 1181].
[ResultValue](#)<T>.

Reference

[ResultValue.TValue. Class](#) [▶ 1181]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.65.3.2 ResultValue.TValue..CreateSuccess Method

Creates a success result.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultValue<TValue> CreateSuccess(
    TValue value
)
```

Parameters

value	Type: TValue [▶ 1181] The value.
-------	---

Return Value

Type: [ResultValue](#) [[▶ 1181](#)].[TValue](#) [[▶ 1181](#)].
ResultValue<T>.

Reference

[ResultValue.TValue. Class](#) [[▶ 1181](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.66 ResultWrite Class

Result for asynchronous ADS write tasks.

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.ResultAds](#) [[▶ 1116](#)]

[TwinCAT.Ads.ResultWrite](#)

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax




C#

```
public class ResultWrite : ResultAds
```












The ResultWrite type exposes the following members.

Properties

	Name	Description
	Empty [▶ 1189]	Gets the empty ResultWrite object.
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)

	Name	Description
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)



Methods

	Name	Description
	CreateError(AdsErrorCode) [▶ 1191]	Creates an error result;
	CreateError(AdsErrorCode, UInt32) [▶ 1191]	Creates an error result;
	CreateSuccess. [▶ 1192]	Creates a success result.
	CreateSuccess(UInt32) [▶ 1192]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]







Also see about this

-  [ResultWrite.CreateError Method](#) [[▶ 1190](#)]
-  [ResultWrite.CreateSuccess Method](#) [[▶ 1192](#)]

6.2.66.1 ResultWrite Properties

The [ResultWrite](#) [[▶ 1187](#)] type exposes the following members.

Properties

	Name	Description
	Empty [▶ 1189]	Gets the empty ResultWrite [▶ 1187] object.
		
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Reference

[ResultWrite Class](#) [[▶ 1187](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.66.1 ResultWrite.Empty Property

Gets the empty [ResultWrite](#) [[▶ 1187](#)] object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultWrite Empty { get; }
```

Property Value

Type: [ResultWrite](#) [[▶ 1187](#)]

The empty / unprocessed result.

Reference


[ResultWrite Class](#) [[▶ 1187](#)]











[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.66.2 ResultWrite Methods

The [ResultWrite](#) [[▶ 1187](#)] type exposes the following members.

Methods

	Name	Description
	CreateError(AdsError) [▶ 1191]	Creates an error result;



	Name	Description
	CreateError(AdsErrorCode, UInt32) [▶ 1191]	Creates an error result;
	CreateSuccess. [▶ 1192]	Creates a success result.
	CreateSuccess(UInt32) [▶ 1192]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultWrite Class](#) [▶ 1187]

[TwinCAT.Ads Namespace](#) [▶ 179]

Also see about this

-  [ResultWrite.CreateError Method](#) [▶ 1190]
-  [ResultWrite.CreateSuccess Method](#) [▶ 1192]

6.2.66.2.1 ResultWrite.CreateError Method

Overload List

	Name	Description
	CreateError(AdsErrorCode) [▶ 1191]	Creates an error result;
	CreateError(AdsErrorCode, UInt32) [▶ 1191]	Creates an error result;

Reference

[ResultWrite Class](#) [▶ 1187]

[TwinCAT.Ads Namespace](#) [▶ 179]

Also see about this

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.66.2.1.1 ResultWrite.CreateError Method (AdsErrorCode)

Creates an error result;

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultWrite CreateError(  
    AdsErrorCode errorCode  
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
-----------	---

Return Value

Type: [ResultWrite](#) [▶ 1187]

ResultWrite.

Reference

[ResultWrite Class](#) [▶ 1187]

[CreateError Overload](#) [▶ 1190]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.66.2.1.2 ResultWrite.CreateError Method (AdsErrorCode, UInt32)

Creates an error result;

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultWrite CreateError(  
    AdsErrorCode errorCode,  
    uint invokeId  
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
invokeId	Type: System.UInt32 The ADS request invoke identifier.

Return Value

Type: [ResultWrite](#) [▶ 1187]

ResultWrite.

Reference[ResultWrite Class \[▸ 1187\]](#)[CreateError Overload \[▸ 1190\]](#)[TwinCAT.Ads Namespace \[▸ 179\]](#)**6.2.66.2.2 ResultWrite.CreateSuccess Method****Overload List**

	Name	Description
	CreateSuccess. [▸ 1192]	Creates a success result.
	CreateSuccess(UInt32) [▸ 1192]	Creates a success result.

Reference[ResultWrite Class \[▸ 1187\]](#)[TwinCAT.Ads Namespace \[▸ 179\]](#)**Also see about this** [TwinCAT.Ads Namespace \[▸ 179\]](#)**6.2.66.2.2.1 ResultWrite.CreateSuccess Method**

Creates a success result.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static ResultWrite CreateSuccess()
```

Return ValueType: [ResultWrite \[▸ 1187\]](#)

ResultWrite.

Reference[ResultWrite Class \[▸ 1187\]](#)[CreateSuccess Overload \[▸ 1192\]](#)[TwinCAT.Ads Namespace \[▸ 179\]](#)**6.2.66.2.2.2 ResultWrite.CreateSuccess Method (UInt32)**

Creates a success result.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultWrite CreateSuccess(
    uint invokeId
)
```

Parameters

invokeld	Type: System.UInt32 The ADS request invoke identifier.
----------	---

Return Value

Type: [ResultWrite](#) [[▶ 1187](#)]
ResultWrite.

Reference

- [ResultWrite Class](#) [[▶ 1187](#)]
- [CreateSuccess Overload](#) [[▶ 1192](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.67 SessionSettings Class

Session settings class

Inheritance Hierarchy

[System.Object](#)
 TwinCAT.Ads.SessionSettings
Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#


```
public class SessionSettings : IAdsSessionSettings
```











The SessionSettings type exposes the following members.

Constructors







	Name	Description
	SessionSettings(Int32) [▶ 1199]	Initializes a new instance of the SessionSettings class.
	SessionSettings(Int32, Boolean) [▶ 1200]	Initializes a new instance of the SessionSettings class.

Properties

	Name	Description
	Default [▶ 1195]	Gets the default Settings (Synchronized).

	Name	Description
		
	DefaultCommunicationTimeout [▶ 1196]	The default communication timeout (5 Seconds)
		
	DefaultResurrectionTime [▶ 1196]	The default resurrection time (21 Seconds)
		
	FastWriteThrough [▶ 1196]	Gets a Settings object that configures the AdsSession for FastWriteThrough
		
	ResurrectionTime [▶ 1197]	Gets or sets the resurrection time (Default: DefaultResurrectionTime [▶ 1196])
	SymbolLoader [▶ 1198]	Gets or sets the symbol loader settings
	Timeout [▶ 1198]	Gets the ADS timeout in milliseconds.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]



Also see about this







 [SessionSettings Constructor](#) [[▶ 1199](#)]

6.2.67.1 SessionSettings Properties

The [SessionSettings](#) [[▶ 1193](#)] type exposes the following members.

Properties

	Name	Description
	Default [▶ 1195]	Gets the default Settings (Synchronized).
		

	Name	Description
 S	DefaultCommunicationTimeout [▶ 1196]	The default communication timeout (5 Seconds)
 S	DefaultResurrectionTime [▶ 1196]	The default resurrection time (21 Seconds)
 S	FastWriteThrough [▶ 1196]	Gets a Settings object that configures the AdsSession for FastWriteThrough
 S	ResurrectionTime [▶ 1197]	Gets or sets the resurrection time (Default: DefaultResurrectionTime [▶ 1196])
 S	SymbolLoader [▶ 1198]	Gets or sets the symbol loader settings
 S	Timeout [▶ 1198]	Gets the ADS timeout in milliseconds.

Reference

[SessionSettings Class](#) [[▶ 1193](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.67.1.1 SessionSettings.Default Property

Gets the default Settings (Synchronized).

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static SessionSettings Default { get; }
```

Property Value

Type: [SessionSettings](#) [[▶ 1193](#)]

The default settings.

Remarks

The following defaults are set here:

Setting	Description
Communication Timeout (Timeout [▶ 1198])	Default communication timeout (DefaultCommunicationTimeout [▶ 1196], default 5s)
Resurrection Timeout (ResurrectionTime [▶ 1197])	Default communication timeout (DefaultResurrectionTime [▶ 1196], default 21s)
Dynamic SymbolLoader settings SymbolLoader [▶ 1198]	Synchronized mode activated (DefaultDynamic [▶ 154])

Reference

[SessionSettings Class](#) [[▶ 1193](#)]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.67.1.2 SessionSettings.DefaultCommunicationTimeout Property

The default communication timeout (5 Seconds)

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static TimeSpan DefaultCommunicationTimeout { get; }
```

Property Value

Type: [TimeSpan](#)

Reference

[SessionSettings Class](#) [▶ 1193]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.67.1.3 SessionSettings.DefaultResurrectionTime Property

The default resurrection time (21 Seconds)

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static TimeSpan DefaultResurrectionTime { get; }
```

Property Value

Type: [TimeSpan](#)

Reference

[SessionSettings Class](#) [▶ 1193]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.67.1.4 SessionSettings.FastWriteThrough Property

Gets a Settings object that configures the AdsSession for FastWriteThrough

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static SessionSettings FastWriteThrough { get; }
```

Property Value

Type: [SessionSettings](#) [[▶ 1193](#)]

Session settings for a fast write through (with 200 ms Timeout).

Remarks

The settings typically can be used for polling clients, where the "FailFast" feature will be bypassed. That means, that communication fails doesn't trigger the FailFast interceptor and every Request will go out via ADS. This has the Drawback that communication Timeouts are longer and subsequent timeouts block the ADS mailbox (with the danger of overflows). So use this setting with care for specific purposes and should not be used for standard communication.

- No Resurrection time and therefore:
- No FailFastHandler active.
- Default communication timeout 200ms.
- Not synchronized Notifications.

Reference

[SessionSettings Class](#) [[▶ 1193](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.67.1.5 **SessionSettings.ResurrectionTime Property**

Gets or sets the resurrection time (Default: [DefaultResurrectionTime](#) [[▶ 1196](#)])

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public TimeSpan ResurrectionTime { get; set; }
```

Property Value

Type: [TimeSpan](#)

The resurrection time.

Implements

[IAdsSessionSettings.ResurrectionTime](#) [[▶ 1038](#)]

Remarks

The resurrection time is the time after a lost connection [Lost](#) [[▶ 72](#)] can be 'resurrected'. This time is set to 21 Seconds by default (a value greater than the standard Ethernet connection timeout of 20s). The reason for this timeout is not to flood the ADS mailbox with requests that cannot be handled by the ethernet infrastructure. As long this Timespan is not expired after a recognized [Lost](#) [[▶ 72](#)], no further data communication is done, and requests are immediately ('FailFast') answered by communication exceptions.

Change this value only for edge cases.

Reference

[SessionSettings Class](#) [[▶ 1193](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.67.1.6 SessionSettings.SymbolLoader Property

Gets or sets the symbol loader settings

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolLoaderSettings SymbolLoader { get; set; }
```

Property Value

Type: [SymbolLoaderSettings](#) [[▶ 148](#)]

The symbol loader.

Implements

[IAdsSessionSettings.SymbolLoader](#) [[▶ 1039](#)]

Reference

[SessionSettings Class](#) [[▶ 1193](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.67.1.7 SessionSettings.Timeout Property

Gets the ADS timeout in milliseconds.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Timeout { get; set; }
```

Property Value

Type: [Int32](#)

The timeout.

Implements

[IAdsSessionSettings.Timeout](#) [[▶ 1039](#)]

Reference







[SessionSettings Class](#) [[▶ 1193](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.67.2 SessionSettings Methods

The [SessionSettings](#) [[▶ 1193](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[SessionSettings Class](#) [► 1193]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.67.3 SessionSettings Constructor

Overload List



	Name	Description
	SessionSettings(Int32) [► 1199]	Initializes a new instance of the SessionSettings [► 1193] class.
	SessionSettings(Int32, Boolean) [► 1200]	Initializes a new instance of the SessionSettings [► 1193] class.

Reference

[SessionSettings Class](#) [► 1193]

[TwinCAT.Ads Namespace](#) [► 179]

Also see about this

-  [SessionSettings Class](#) [► 1193]
-  [TwinCAT.Ads Namespace](#) [► 179]

6.2.67.3.1 SessionSettings Constructor (Int32)

Initializes a new instance of the [SessionSettings](#) [► 1193] class.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SessionSettings(
    int timeout
)
```

Parameters

timeout	Type: System.Int32
---------	------------------------------------

Reference

[SessionSettings Class](#) [► 1193]

[SessionSettings Overload](#) [► 1199]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.67.3.2 SessionSettings Constructor (Int32, Boolean)

Initializes a new instance of the [SessionSettings](#) [► 1193] class.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public SessionSettings(
    int timeout,
    bool fastWriteThrough
)
```

Parameters

timeout	Type: System.Int32 The timeout.
fastWriteThrough	Type: System.Boolean if set to true [fast write through].

Reference

[SessionSettings Class](#) [► 1193]

[SessionSettings Overload](#) [► 1199]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.68 StateInfo Structure

The structure contains the ADS state and device state.

Namespace: [TwinCAT.Ads](#) [► 179]




Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**





```
public struct StateInfo : IEquatable<StateInfo>
```

The StateInfo type exposes the following members.






Constructors

	Name	Description
	StateInfo(ReadOnly Span.Byte.) [▶ 1203]	Initializes a new instance of the StateInfo struct.
	StateInfo(AdsState, Int16) [▶ 1202]	Initializes a new Instance of the StateInfo struct.
	StateInfo(AdsState, UInt16) [▶ 1203]	Initializes a new Instance of the StateInfo struct.





Properties

	Name	Description
	AdsState [▶ 1204]	Gets or sets the ADS state of this StateInfo object.
	DeviceState [▶ 1205]	Gets or sets the device state of this StateInfo object.
	Empty [▶ 1205]	Empty / Invalid / Uninitialized state.
		

Methods

	Name	Description
	Equals(Object) [▶ 1206]	Determines whether the specified <u>Object</u> is equal to this instance. (Overrides <u>ValueType.Equals(Object)</u> .)
	Equals(StateInfo) [▶ 1207]	Determines whether the specified StateInfo is equal to this instance.
	GetHashCode [▶ 1207]	Returns a hash code for this instance. (Overrides <u>ValueType.GetHashCode</u> .)
	GetType	Gets the <u>Type</u> of the current instance. (Inherited from <u>Object</u> .)
	ToString	Returns the fully qualified type name of this instance. (Inherited from <u>ValueType</u> .)

Operators




	Name	Description
 	Equality [▶ 1208]	Implements the ==.
 	Inequality [▶ 1208]	Implements the !=.

Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.68.1 StateInfo Constructor

Overload List

	Name	Description
	StateInfo(ReadOnlySpan<Byte>) [▶ 1203]	Initializes a new instance of the StateInfo [▶ 1200] struct.
	StateInfo(AdsState, Int16) [▶ 1202]	Initializes a new Instance of the StateInfo struct.
	StateInfo(AdsState, UInt16) [▶ 1203]	Initializes a new Instance of the StateInfo struct.

Reference

[StateInfo Structure](#) [[▶ 1200](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.68.1.1 StateInfo Constructor (ReadOnlySpan`1)

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public StateInfo(
    ReadOnlySpan data
)
```

Parameters

data Type: [ReadOnlySpan](#)

Reference

[StateInfo Structure](#) [[▶ 1200](#)]

[StateInfo Overload](#) [[▶ 1202](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.68.1.2 StateInfo Constructor (AdsState, Int16)

Initializes a new Instance of the StateInfo struct.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public StateInfo(
    AdsState adsState,
    short deviceState
)
```

Parameters

adsState	Type: TwinCAT.Ads.AdsState [▶ 729] Ads state.
deviceState	Type: System.Int16 Device state.

Reference

[StateInfo Structure](#) [▶ 1200]

[StateInfo Overload](#) [▶ 1202]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.68.1.3 StateInfo Constructor (ReadOnlySpan.Byte.)

Initializes a new instance of the [StateInfo](#) [▶ 1200] struct.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public StateInfo(
    ReadOnlySpan<byte> data
)
```

Parameters

data	Type: System.ReadOnlySpan.Byte . The data.
------	---

Reference

[StateInfo Structure](#) [▶ 1200]

[StateInfo Overload](#) [▶ 1202]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.68.1.4 StateInfo Constructor (AdsState, UInt16)

Initializes a new Instance of the StateInfo struct.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public StateInfo(
    AdsState adsState,
    ushort deviceState
)
```

Parameters

adsState	Type: TwinCAT.Ads.AdsState [▸ 729] Ads state.
deviceState	Type: System.UInt16 Device state.

Reference

[StateInfo Structure \[▸ 1200\]](#)





[StateInfo Overload \[▸ 1202\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.68.2 StateInfo Properties

The [StateInfo \[▸ 1200\]](#) type exposes the following members.

Properties

	Name	Description
	AdsState [▸ 1204]	Gets or sets the ADS state of this StateInfo [▸ 1200] object.
	DeviceState [▸ 1205]	Gets or sets the device state of this StateInfo [▸ 1200] object.
 	Empty [▸ 1205]	Empty / Invalid / Uninitialized state.

Reference

[StateInfo Structure \[▸ 1200\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.68.2.1 StateInfo.AdsState Property

Gets or sets the ADS state of this [StateInfo \[▸ 1200\]](#) object.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public AdsState AdsState { get; set; }
```

Property Value

Type: [AdsState \[▸ 729\]](#)

Reference

[StateInfo Structure \[▸ 1200\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.68.2 StateInfo.DeviceState Property

Gets or sets the device state of this [StateInfo \[▸ 1200\]](#) object.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public short DeviceState { get; set; }
```

Property Value

Type: [Int16](#)

Reference

[StateInfo Structure \[▸ 1200\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.68.3 StateInfo.Empty Property

Empty / Invalid / Uninitialized state.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static StateInfo Empty { get; }
```

Property Value

Type: [StateInfo \[▸ 1200\]](#)

The empty.

Reference




[StateInfo Structure \[▸ 1200\]](#)



[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.68.3 StateInfo Methods

The [StateInfo \[▸ 1200\]](#) type exposes the following members.

Methods

	Name	Description
	Equals(Object) [▸ 1206]	Determines whether the specified Object is equal to this instance. (Overrides ValueType.Equals(Object) .)
	Equals(StateInfo) [▸ 1207]	Determines whether the specified StateInfo [▸ 1200] is equal to this instance.
	GetHashCode [▸ 1207]	Returns a hash code for this instance. (Overrides ValueType.GetHashCode() .)

	Name	Description
	GetType	Gets the <u>Type</u> of the current instance. (Inherited from <u>Object</u> .)
	ToString	Returns the fully qualified type name of this instance. (Inherited from <u>ValueType</u> .)



Reference

[StateInfo Structure \[▸ 1200\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.68.3.1 StateInfo.Equals Method

Overload List

	Name	Description
	Equals(Object) [▸ 1206]	Determines whether the specified <u>Object</u> is equal to this instance. (Overrides <u>ValueType.Equals(Object)</u> .)
	Equals(StateInfo) [▸ 1207]	Determines whether the specified <u>StateInfo</u> [▸ 1200] is equal to this instance.

Reference

[StateInfo Structure \[▸ 1200\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.68.3.1.1 StateInfo.Equals Method (Object)

Determines whether the specified Object is equal to this instance.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool Equals(
    Object? ob
)
```

Parameters

ob	Type: <u>System.Object</u> The <u>Object</u> to compare with this instance.
----	--

Return Value

Type: Boolean

true if the specified Object is equal to this instance; otherwise, false.

Reference

[StateInfo Structure \[▸ 1200\]](#)

[Equals Overload \[▸ 1206\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.68.3.1.2 StateInfo.Equals Method (StateInfo)

Determines whether the specified [StateInfo \[▸ 1200\]](#) is equal to this instance.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Equals(  
    StateInfo info  
)
```

Parameters

info	Type: TwinCAT.Ads.StateInfo [▸ 1200] The StateInfo [▸ 1200] to compare with this instance.
------	---

Return Value

Type: [Boolean](#)

true if the specified [StateInfo \[▸ 1200\]](#) is equal to this instance; otherwise, false.

Implements

[IEquatable.T..Equals\(T\)](#)

Reference

[StateInfo Structure \[▸ 1200\]](#)

[Equals Overload \[▸ 1206\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.68.3.2 StateInfo.GetHashCode Method

Returns a hash code for this instance.

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override int GetHashCode()
```

Return Value

Type: [Int32](#)

A hash code for this instance, suitable for use in hashing algorithms and data structures like a hash table.

Reference





[StateInfo Structure \[▸ 1200\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.68.4 StateInfo Operators

The [StateInfo](#) [[▶ 1200](#)] type exposes the following members.

Operators

	Name	Description
 	Equality [▶ 1208]	Implements the ==.
 	Inequality [▶ 1208]	Implements the !=.

Reference

[StateInfo Structure](#) [[▶ 1200](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.68.4.1 StateInfo.Equality Operator

Implements the ==.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool operator ==(
    StateInfo a,
    StateInfo b
)
```

Parameters

a	Type: TwinCAT.Ads.StateInfo [▶ 1200] a.
b	Type: TwinCAT.Ads.StateInfo [▶ 1200] The b.

Return Value

Type: [Boolean](#)

The result of the operator.

Reference

[StateInfo Structure](#) [[▶ 1200](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.68.4.2 StateInfo.Inequality Operator

Implements the !=.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool operator !=(
    StateInfo a,
    StateInfo b
)
```

Parameters

a	Type: TwinCAT.Ads.StateInfo [▸ 1200] a.
b	Type: TwinCAT.Ads.StateInfo [▸ 1200] The b.

Return Value

Type: [Boolean](#)
The result of the operator.

Reference

- [StateInfo Structure \[▸ 1200\]](#)
- [TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.69 TaskExtensions Class

TaskExtensions for Task Cancellation and Timeout

Inheritance Hierarchy

[System.Object](#)
TwinCAT.Ads.TaskExtensions

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229







Syntax

C#

```
public static class TaskExtensions
```

The TaskExtensions type exposes the following members.

Methods

	Name	Description
 	WithCancellation.T. [▸ 1210]	Extends a worker Task with cancellation.
 	WithCancellationAn dTimeout [▸ 1211]	Extends a worker task with timeout and Cancellation
 	WithTimeout [▸ 1212]	Extends a worker task with a timeout.







Reference

[TwinCAT.Ads Namespace](#) [► 179]

6.2.69.1 TaskExtensions Methods

The [TaskExtensions](#) [► 1209] type exposes the following members.

Methods

	Name	Description
 	WithCancellation.T. [► 1210]	Extends a worker Task with cancellation.
 	WithCancellationAndTimeout [► 1211]	Extends a worker task with timeout and Cancellation
 	WithTimeout [► 1212]	Extends a worker task with a timeout.

Reference

[TaskExtensions Class](#) [► 1209]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.69.1.1 TaskExtensions.WithCancellation.T. Method

Extends a worker Task with cancellation.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static Task<T> WithCancellation<T>(
    this Task<T> worker,
    CancellationToken cancellationToken
)
```

Parameters

worker	Type: System.Threading.Tasks.Task.T. The worker task.
cancellationToken	Type: System.Threading.CancellationToken The cancellation token.

Type Parameters

T	The Task Result
---	-----------------

Return Value

Type: [Task.T.](#)
[Task<T>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [Task.T](#).. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Exceptions

Exception	Condition
OperationCanceledException	

Reference

[TaskExtensions Class \[▶ 1209\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.69.1.2 TaskExtensions.WithCancellationAndTimeout Method

Extends a worker task with timeout and Cancellation

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static Task<AdsErrorCode> WithCancellationAndTimeout (
    this Task<AdsErrorCode> worker,
    TimeSpan timeout,
    CancellationToken cancellationToken
)
```

Parameters

worker	Type: System.Threading.Tasks.Task.AdsErrorCode [▶ 664] . The worker task.
timeout	Type: System.TimeSpan The timeout.
cancellationToken	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode \[▶ 664\]](#).

Task<AdsErrorCode>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [Task.AdsErrorCode \[▶ 664\]](#).. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Exceptions

Exception	Condition
OperationCanceledException	

Reference

[TaskExtensions Class](#) [► 1209]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.69.1.3 TaskExtensions.WithTimeout Method

Extends a worker task with a timeout.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static Task<AdsErrorCode> WithTimeout(
    this Task<AdsErrorCode> worker,
    TimeSpan timeout
)
```

Parameters

worker	Type: System.Threading.Tasks.Task.AdsErrorCode [► 664]. The worker task
timeout	Type: System.TimeSpan The timeout.

Return Value

Type: [Task.AdsErrorCode](#) [► 664].

Task<AdsErrorCode>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [Task.AdsErrorCode](#) [► 664]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[TaskExtensions Class](#) [► 1209]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.70 TransportProtocols Enumeration

Enum ADS TransportProtocol

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
[FlagsAttribute]
public enum TransportProtocols
```

Members

	Member name	Value	Description
	None	0	None / Uninitialized
	Router	1	ADS via Router
	Tcplp	2	ADS via TCP/IP (without router)
	All	3	Indicates that Router and Tcplp are appropriate (for establishing connections)

Reference

[TwinCAT.Ads Namespace](#) |▸ [179](#)]

6.2.71 ValueNotificationEventArgs.T. Class

Arguments for [AdsNotificationEx](#) |▸ [993](#)] events.

Inheritance Hierarchy

[System.Object](#)
[System.EventArgs](#)
[TwinCAT.Ads.AdsNotificationEventArgs](#) |▸ [695](#)]
[TwinCAT.Ads.ValueNotificationEventArgs.T.](#)

Namespace: [TwinCAT.Ads](#) |▸ [179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#





```
public class ValueNotificationEventArgs<T> : AdsNotificationEventArgs
```


Type Parameters

T







The ValueNotificationEventArgs.T. type exposes the following members.

Properties

	Name	Description
	Data ▸ 697]	Memory object holding the Notification Data/Value. (Inherited from AdsNotificationEventArgs ▸ 695].)
	Handle ▸ 697]	Gets the Notification handle. (Inherited from AdsNotificationEventArgs ▸ 695].)
	TimeStamp ▸ 698]	Gets the time stamp of this Notification as DateTimeOffset . (Inherited from AdsNotificationEventArgs ▸ 695].)
	UserData ▸ 698]	Gets the user object. This object is passed by to AddDeviceNotification and can be used to store data. (Inherited from AdsNotificationEventArgs ▸ 695].)

	Name	Description
	Value [▶ 1214]	Value of the ADS Notification.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)






Reference

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.71.1 ValueNotificationEventArgs.T. Properties

The [ValueNotificationEventArgs.T. \[▶ 1213\]](#) generic type exposes the following members.

Properties

	Name	Description
	Data [▶ 697]	Memory object holding the Notification Data/Value. (Inherited from AdsNotificationEventArgs [▶ 695] .)
	Handle [▶ 697]	Gets the Notification handle. (Inherited from AdsNotificationEventArgs [▶ 695] .)
	TimeStamp [▶ 698]	Gets the time stamp of this Notification as DateTimeOffset . (Inherited from AdsNotificationEventArgs [▶ 695] .)
	UserData [▶ 698]	Gets the user object. This object is passed by to AddDeviceNotification and can be used to store data. (Inherited from AdsNotificationEventArgs [▶ 695] .)
	Value [▶ 1214]	Value of the ADS Notification.

Reference

[ValueNotificationEventArgs.T. Class \[▶ 1213\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.71.1.1 ValueNotificationEventArgs.T..Value Property

Value of the ADS Notification.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T Value { get; }
```

Property Value

Type: [T](#) [[▶](#) [1213](#)]

Reference







[ValueNotificationEventArgs.T. Class](#) [[▶](#) [1213](#)]

[TwinCAT.Ads Namespace](#) [[▶](#) [179](#)]

6.2.71.2 ValueNotificationEventArgs.T. Methods

The [ValueNotificationEventArgs.T.](#) [[▶](#) [1213](#)] generic type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ValueNotificationEventArgs.T. Class](#) [[▶](#) [1213](#)]

[TwinCAT.Ads Namespace](#) [[▶](#) [179](#)]

6.2.72 AdsClientExtensions Class

Class [AdsClientExtensions](#).

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.AdsClientExtensions](#)

Namespace: [TwinCAT.Ads](#) [[▶](#) [179](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public static class AdsClientExtensions
```

Methods

	Name	Description
	<u>ReadWithFallback(IAdsConnection, UInt32, UInt32, Memory.Byte., UInt32, Boolean.)</u> [► 1219]	Ads Read with fallback.
	<u>ReadWithFallback(IAdsConnection, UInt32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., Boolean.)</u> [► 1220]	Ads Read with Fallback.
	<u>ReadWithFallbackAsync(IAdsConnection, UInt32, UInt32, UInt32, Memory.Byte., CancellationToken)</u> [► 1221]	Ads Read with fallback as an asynchronous operation.
	<u>ReadWithFallbackAsync(IAdsConnection, UInt32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., CancellationToken)</u> [► 1222]	Ads Read with fallback as an asynchronous operation.
	<u>RepeatedRead(IAdsConnection, UInt32, UInt32, Memory.Byte., Int32, TimeSpan)</u> [► 1224]	Repeated AdsRead.
	<u>RepeatedRead(IAdsConnection, UInt32, UInt32, Memory.Byte., Int32, TimeSpan, Func.ResultRead, Boolean.)</u> [► 1225]	Repeated AdsRead.
	<u>RepeatedReadAsync(IAdsConnection,</u>	Repeated AdsRead as an asynchronous operation.

	Name	Description
	UInt32 , UInt32 , Memory.Byte. , Int32 , TimeSpan , CancellationToken) [▶ 1226]	
	RepeatedReadAsync (IAdsConnection , UInt32 , UInt32 , Memory.Byte. , Int32 , TimeSpan , Func.ResultRead , Boolean. , CancellationToken) [▶ 1227]	Repeated AdsRead as an asynchronous operation.

Reference

[TwinCAT.Ads Namespace](#) [▶ [179](#)]

6.2.72.1 AdsClientExtensions Methods

Methods

	Name	Description
	ReadWithFallback (IAdsConnection , UInt32 , UInt32 , Memory.Byte. , UInt32 , Boolean.) [▶ 1219]	Ads Read with fallback.
	ReadWithFallback (IAdsConnection , UInt32 , UInt32 , UInt32 , Memory.Byte. , Func.ResultRead , Boolean. , Boolean.) [▶ 1220]	Ads Read with Fallback.
	ReadWithFallbackAsync (IAdsConnection , UInt32 , UInt32 , UInt32 , Memory.Byte. , CancellationToken) [▶ 1221]	Ads Read with fallback as an asynchronous operation.
	ReadWithFallbackAsync (IAdsConnection , UInt32 , UInt32 ,	Ads Read with fallback as an asynchronous operation.

	Name	Description
	UInt32 , Memory.Byte , Func.ResultRead , Boolean , CancellationToken) [► 1222]	
	RepeatedRead (IAdsConnection , UInt32 , UInt32 , Memory.Byte , Int32 , TimeSpan) [► 1224]	Repeated AdsRead.
	RepeatedRead (IAdsConnection , UInt32 , UInt32 , Memory.Byte , Int32 , TimeSpan , Func.ResultRead , Boolean .) [► 1225]	Repeated AdsRead.
	RepeatedReadAsync (IAdsConnection , UInt32 , UInt32 , Memory.Byte , Int32 , TimeSpan , CancellationToken) [► 1226]	Repeated AdsRead as an asynchronous operation.
	RepeatedReadAsync (IAdsConnection , UInt32 , UInt32 , Memory.Byte , Int32 , TimeSpan , Func.ResultRead , Boolean , CancellationToken) [► 1227]	Repeated AdsRead as an asynchronous operation.

Reference

[AdsClientExtensions Class](#) [► [1215](#)]

[TwinCAT.Ads Namespace](#) [► [179](#)]

6.2.72.1.1 AdsClientExtensions.ReadWithFallback Method

Overload List

	Name	Description
	ReadWithFallback (IAdsConnection ,	Ads Read with fallback.

	Name	Description
	UInt32, UInt32, Memory.Byte., UInt32, Boolean. [▶ 1219]	
	ReadWithFallback(IAdsConnection, UInt32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., Boolean.) [▶ 1220]	Ads Read with Fallback.

Reference

[AdsClientExtensions Class](#) [▶ 1215]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.72.1.1 AdsClientExtensions.ReadWithFallback Method (IAdsConnection, UInt32, UInt32, Memory.Byte., UInt32, Boolean.)

Ads Read with fallback.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultRead ReadWithFallback(
    this IAdsConnection connection,
    uint indexGroup,
    uint indexOffset,
    Memory<byte> buffer,
    uint fallbackOffset,
    out bool fallback
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
buffer	Type: System.Memory.Byte. The buffer.
fallbackOffset	Type: System.UInt32 The fallback offset.
fallback	Type: System.Boolean. if set to true [fallback].

Return Value

Type: [ResultRead](#) [[▶ 1143](#)]
ResultRead.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AdsClientExtensions Class](#) [[▶ 1215](#)]

[ReadWithFallback Overload](#) [[▶ 1218](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.72.1.1.2 **AdsClientExtensions.ReadWithFallback Method (IAdsConnection, UInt32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., Boolean.)**

Ads Read with Fallback.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultRead ReadWithFallback(
    this IAdsConnection connection,
    uint indexGroup,
    uint indexOffset,
    uint fallbackOffset,
    Memory<byte> buffer,
    Func<ResultRead, bool> shouldProcessFallback,
    out bool fallback
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
fallbackOffset	Type: System.UInt32 The fallback offset.
buffer	Type: System.Memory.Byte . The buffer.
shouldProcessFallback	Type: System.Func.ResultRead [▶ 1143], Boolean . The should process fallback.
fallback	Type: System.Boolean . if set to true [fallback].

Return Value

Type: [ResultRead](#) [[▶ 1143](#)]
 ResultRead.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

- [AdsClientExtensions Class](#) [[▶ 1215](#)]
- [ReadWithFallback Overload](#) [[▶ 1218](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.72.1.2 AdsClientExtensions.ReadWithFallbackAsync Method

Overload List

Name	Description
ReadWithFallbackAsync(IAdsConnection, UInt32, UInt32, UInt32, Memory.Byte, CancellationToken) ▶ 1221	Ads Read with fallback as an asynchronous operation.
ReadWithFallbackAsync(IAdsConnection, UInt32, UInt32, UInt32, Memory.Byte, Func.ResultRead, Boolean, CancellationToken) ▶ 1222	Ads Read with fallback as an asynchronous operation.

Reference

- [AdsClientExtensions Class](#) [[▶ 1215](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.72.1.2.1 AdsClientExtensions.ReadWithFallbackAsync Method (IAdsConnection, UInt32, UInt32, UInt32, Memory.Byte, CancellationToken)

Ads Read with fallback as an asynchronous operation.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]
Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static Task<ResultRead> ReadWithFallbackAsync (
    this IAdsConnection connection,
    uint indexGroup,
    uint indexOffset,
    uint fallbackOffset,
    Memory<byte> buffer,
    CancellationToken cancel
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
fallbackOffset	Type: System.UInt32 The fallback offset.
buffer	Type: System.Memory.Byte . The buffer.
cancel	Type: System.Threading.CancellationToken The cancel.

Return Value

Type: [Task.ResultRead](#) [▶ 1143].
Task<ResultRead>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [▶ 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AdsClientExtensions Class](#) [▶ 1215]

[ReadWithFallbackAsync Overload](#) [▶ 1221]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.72.1.2.2 **AdsClientExtensions.ReadWithFallbackAsync Method (IAdsConnection, UInt32, UInt32, UInt32, Memory.Byte., Func.ResultRead, Boolean., CancellationToken)**

Ads Read with fallback as an asynchronous operation.

Namespace: [TwinCAT.Ads](#) [▶ 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static Task<ResultRead> ReadWithFallbackAsync (
    this IAdsConnection connection,
    uint indexGroup,
    uint indexOffset,
    uint fallbackOffset,
    Memory<byte> buffer,
    Func<ResultRead, bool> shouldProcessFallback,
    CancellationToken cancel
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
fallbackOffset	Type: System.UInt32 The fallback offset.
buffer	Type: System.Memory.Byte . The buffer.
shouldProcessFallback	Type: System.Func.ResultRead [▶ 1143], Boolean . The should process fallback.
cancel	Type: System.Threading.CancellationToken The cancel.

Return Value

Type: [Task.ResultRead](#) [▶ 1143].
ResultRead.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [▶ 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AdsClientExtensions Class](#) [▶ 1215]

[ReadWithFallbackAsync Overload](#) [▶ 1221]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.72.1.3 AdsClientExtensions.RepeatedRead Method

Overload List

	Name	Description
	RepeatedRead(IAdsConnection, UInt32,	Repeated AdsRead.

	Name	Description
	UInt32 , Memory.Byte , Int32 , TimeSpan) [▶ 1224]	
	RepeatedRead (IAdsConnection , UInt32 , UInt32 , Memory.Byte , Int32 , TimeSpan , Func.ResultRead , Boolean .) [▶ 1225]	Repeated AdsRead.

Reference

[AdsClientExtensions Class](#) [[▶ 1215](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.72.1.3.1 **AdsClientExtensions.RepeatedRead Method (IAdsConnection, UInt32, UInt32, Memory.Byte, Int32, TimeSpan)**

Repeated AdsRead.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultRead RepeatedRead(
    this IAdsConnection connection,
    uint indexGroup,
    uint indexOffset,
    Memory<byte> buffer,
    int repeat,
    TimeSpan waitTime
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
buffer	Type: System.Memory.Byte . The buffer.
repeat	Type: System.Int32 The repeat.
waitTime	Type: System.TimeSpan The wait time.

Return Value

Type: [ResultRead](#) [[▶ 1143](#)]

ResultRead.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AdsClientExtensions Class](#) [► 1215]

[RepeatedRead Overload](#) [► 1223]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.72.1.3.2 AdsClientExtensions.RepeatedRead Method (IAdsConnection, UInt32, UInt32, Memory.Byte., Int32, TimeSpan, Func.ResultRead, Boolean.)

Repeated AdsRead.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultRead RepeatedRead(
    this IAdsConnection connection,
    uint indexGroup,
    uint indexOffset,
    Memory<byte> buffer,
    int repeat,
    TimeSpan waitTime,
    Func<ResultRead, bool> shouldRepeat
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
buffer	Type: System.Memory.Byte. The buffer.
repeat	Type: System.Int32 The repeat.
waitTime	Type: System.TimeSpan The wait time.
shouldRepeat	Type: System.Func.ResultRead [► 1143], Boolean. The should repeat.

Return Value

Type: [ResultRead](#) [► 1143]
ResultRead.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AdsClientExtensions Class](#) [► 1215]

[RepeatedRead Overload](#) [► 1223]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.72.1.4 AdsClientExtensions.RepeatedReadAsync Method**Overload List**

	Name	Description
	RepeatedReadAsync (IAdsConnection, UInt32, UInt32, Memory.Byte., Int32, TimeSpan, CancellationToken) [► 1226]	Repeated AdsRead as an asynchronous operation.
	RepeatedReadAsync (IAdsConnection, UInt32, UInt32, Memory.Byte., Int32, TimeSpan, Func.ResultRead, Boolean., CancellationToken) [► 1227]	Repeated AdsRead as an asynchronous operation.

Reference

[AdsClientExtensions Class](#) [► 1215]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.72.1.4.1 AdsClientExtensions.RepeatedReadAsync Method (IAdsConnection, UInt32, UInt32, Memory.Byte., Int32, TimeSpan, CancellationToken)

Repeated AdsRead as an asynchronous operation.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static Task<ResultRead> RepeatedReadAsync(
    this IAdsConnection connection,
    uint indexGroup,
```

```

uint indexOffset,
Memory<byte> buffer,
int repeat,
TimeSpan waitTime,
CancellationTokel cancel
)

```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
buffer	Type: System.Memory.Byte . The buffer.
repeat	Type: System.Int32 The repeat.
waitTime	Type: System.TimeSpan The wait time.
cancel	Type: System.Threading.CancellationTokel The cancel.

Return Value

Type: [Task.ResultRead](#) [► 1143].
Task<ResultRead>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AdsClientExtensions Class](#) [► 1215]

[RepeatedReadAsync Overload](#) [► 1226]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.72.1.4.2 **AdsClientExtensions.RepeatedReadAsync Method (IAdsConnection, UInt32, UInt32, Memory.Byte., Int32, TimeSpan, Func.ResultRead, Boolean., CancellationTokel)**

Repeated AdsRead as an asynchronous operation.

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public static Task<ResultRead> RepeatedReadAsync (
    this IAdsConnection connection,
    uint indexGroup,
    uint indexOffset,
    Memory<byte> buffer,

```

```

int repeat,
TimeSpan waitTime,
Func<ResultRead, bool> shouldRepeat,
CancellationToken cancel
)

```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
buffer	Type: System.Memory.Byte . The buffer.
repeat	Type: System.Int32 The repeat.
waitTime	Type: System.TimeSpan The wait time.
shouldRepeat	Type: System.Func.ResultRead [► 1143], Boolean . The should repeat.
cancel	Type: System.Threading.CancellationToken The cancel.

Return Value

Type: [Task.ResultRead](#) [► 1143].
ResultRead.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AdsClientExtensions Class](#) [► 1215]

[RepeatedReadAsync Overload](#) [► 1226]

[TwinCAT.Ads Namespace](#) [► 179]

6.2.73 AdsNotificationsInvalidatedEventArgs Class

Class [AdsNotificationsUnregisteredEventArgs](#). This class cannot be inherited. Implements the [EventArgs](#)

Inheritance Hierarchy

[System.Object](#)

[System.EventArgs](#)

[TwinCAT.Ads.AdsNotificationsInvalidatedEventArgs](#)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public sealed class AdsNotificationsInvalidatedEventArgs : EventArgs
```

The AdsNotificationsInvalidatedEventArgs type exposes the following members.

Constructors

	Name	Description
	AdsNotificationsInvalidatedEventArgs [▶ 1229]	Initializes a new instance of the AdsNotificationsInvalidatedEventArgs class.

Properties

	Name	Description
	Notifications [▶ 1230]	Gets the notifications that occurred at a specific TimeStamp [▶ 1230]
	TimeStamp [▶ 1230]	Gets the time stamp of the Notifications [▶ 1230]

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

[System.EventArgs](#)

6.2.73.1 AdsNotificationsInvalidatedEventArgs Constructor

Initializes a new instance of the [AdsNotificationsInvalidatedEventArgs](#) [[▶ 1228](#)] class.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsNotificationsInvalidatedEventArgs(
    DateTimeOffset dateTime,
    IList<Notification> notifications
)
```

Parameters

dateTime	Type: System.DateTimeOffset The date time.
notifications	Type: System.Collections.Generic.IList.Notification [▶ 1100]. The notifications.

Reference

[AdsNotificationsInvalidatedEventArgs Class \[▸ 1228\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.73.2 AdsNotificationsInvalidatedEventArgs Properties

The [AdsNotificationsInvalidatedEventArgs \[▸ 1228\]](#) type exposes the following members.

Properties

	Name	Description
	Notifications [▸ 1230]	Gets the notifications that occurred at a specific TimeStamp [▸ 1230]
	TimeStamp [▸ 1230]	Gets the time stamp of the Notifications [▸ 1230]

Reference

[AdsNotificationsInvalidatedEventArgs Class \[▸ 1228\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.73.2.1 AdsNotificationsInvalidatedEventArgs.Notifications Property

Gets the notifications that occurred at a specific [TimeStamp \[▸ 1230\]](#)

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public IList<Notification> Notifications { get; }
```

Property Value

Type: [IList.Notification \[▸ 1100\]](#).

The notifications.

Reference

[AdsNotificationsInvalidatedEventArgs Class \[▸ 1228\]](#)

[TwinCAT.Ads Namespace \[▸ 179\]](#)

6.2.73.2.2 AdsNotificationsInvalidatedEventArgs.TimeStamp Property

Gets the time stamp of the [Notifications \[▸ 1230\]](#)

Namespace: [TwinCAT.Ads \[▸ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public DateTimeOffset TimeStamp { get; }
```

Property Value

Type: [DateTimeOffset](#)
 The time stamp.

Reference

[AdsNotificationsInvalidatedEventArgs Class \[▶ 1228\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.73.3 AdsNotificationsInvalidatedEventArgs Methods

The [AdsNotificationsInvalidatedEventArgs \[▶ 1228\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[AdsNotificationsInvalidatedEventArgs Class \[▶ 1228\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.74 IndexGroupSymbolAccess Enumeration

ADS IndexGroups defined for ADS Symbol Handling

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum IndexGroupSymbolAccess
```

Members

	Member name	Value	Description
	HandleByName	61443	Handle by SymbolName (ADSIGRP_SYM_HNDBYNAME, 0xF003)
	ValueByName	61444	Value By SymbolName (ADSIGRP_SYM_VALBYNAME, 0xF004, 61444)
	ValueByHandle	61445	Value By handle(ADSIGRP_SYM_VALBYHND, 0xF005, 61445)
	ReleaseHandle	61446	RELEASEHND (ADSIGRP_SYM_RELEASEHND, 0xF006, 61446)

	Member name	Value	Description
	SymbolByName	61447	Symbol Info by Name (ADSIGRP_SYM_INFOBYNAME, 0xF007, 61447)
	SymbolVersion	61448	Get Symbol Version (ADSIGRP_SYM_VERSION, 0xF008, 61448)
	SymbolByNameEx	61449	Get SymbolInfo by Name (extended version) (ADSIGRP_SYM_INFOBYNAMEEX, 0xF009, 61449)
	DownloadSymbols	61450	Symbol Download (ADSIGRP_SYM_DOWNLOAD, 0xF00A, 61450)
	UploadSymbols	61451	UploadSymbols (ADSIGRP_SYM_UPLOAD, 0xF00B, 61451)
	UploadInfo	61452	Symbol Upload Info (ADSIGRP_SYM_UPLOAD, 0xF00C, 61452)
	DownloadSymbols2	61453	SymbolDownload (2nd version) (ADSIGRP_SYM_DOWNLOAD2, 0xF00D, 61453)
	UploadTypes	61454	DT_UPLOAD (ADSIGRP_SYM_DT_UPLOAD, 0xF00E, 61454)
	UploadInfo2	61455	Symbol Upload Info (2nd Version) (ADSIGRP_SYM_UPLOADINFO2, 0xF00F, 61455)
	Notification	61456	Notification of named handle (ADSIGRP_SYM_SYMNOTE, 0xF010, 61456)
	Data TypeByName	61457	Data Type Info by Name (ADSIGRP_SYM_DT_INFOBYNAMEEX, 0xF011, 61457)
	AddressByHandle	61458	Address by handle (ADSIGRP_SYM_ADDRBYHND, 0xF012, 61458)
	PointerSupport	61459	Symbol Pointer Support (ADSIGRP_SYM_POINTER_SUPPORT, 0xF013, 61459)
	PointerAccess	61460	Pointer Access (ADSIGRP_SYM_POINTER_ACCESS, 0xF014, 61460)
	ReferenceSupport	61461	Reference Support (ADSIGRP_SYM_REFERENCE_SUPPORT, 0xF015, 61461)
	ReferenceAccess	61462	Reference Access (ADSIGRP_SYM_REFERENCE_ACCESS, 0xF016, 61462)
	ValueByHandleMasked	61464	Value Handle with Mask (ADSIGRP_SYM_VALBYHND_WITHMASK, 0xF018, 61464)
	NoSubSymbolAccess	61465	No Access to SubSymbol (ADSIGRP_SYM_NOACCESS_TO_SUBSYM, 0xF019, 61465)

	Member name	Value	Description
	PointerBitAccess	61466	POINTER with BitAccess (ADSIGRP_SYM_POINTER_BITACCESS, 0xF01A, 61466)
	ReferenceBitAccess	61467	Reference with BitAccess (ADSIGRP_SYM_REFERENCE_BITACCESS, 0xF01B, 61467)

Reference

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.75 AdsStateCommand Enumeration

Describes the AdsState Commands

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum AdsStateCommand
```

Members

	Member name	Value	Description
	None	0	Ads State is Invalid / Uninitialized
	Reset	2	Resets the AdsServer
	Start	4	Starts the AdsServer
	Stop	6	Stops teh AdsServer
	Reconfig	16	Reconfig (System should restart in config mode)
	Run	5	Sets an AdsServer to Run

Remarks

This enumeration is to refactor out the commands from the states.

Reference

[TwinCAT.Ads Namespace \[▶ 179\]](#)

[TwinCAT.Ads.AdsState \[▶ 729\]](#)

6.2.76 AmsPortRange Enumeration

Defines ranges of [AmsPorts](#). [[▶ 795](#)]

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum AmsPortRange
```

Members

	Member name	Value	Description
	PORT_FIRST	1	Defines the first valid port.
	PORT_LAST	65534	Defines the last valid port (0xFFFE, 65534)
	S7COMM_FIRST	20100	(AMSPORT_R3_S7COMM_FIRST,20100)
	S7COMM_LAST	20199	(AMSPORT_R3_S7COMM_LAST,20199)
	CUSTOMER_FIRST	25000	First usable port for customer R3 ADS Servers (AMSPORT_R3_CUSTOMER_FIRST,25000, 0x61A8)
	CUSTOMER_LAST	25999	Last usable port for customer R3 ADS Servers (AMSPORT_R3_CUSTOMER_LAST,25999,0x658F)
	CUSTOMERPRIVATE_FIRST	26000	First usable port for private networks (0x6590)
	CUSTOMERPRIVATE_LAST	26999	Last usable port for private networks (0x6977)

Reference

[TwinCAT.Ads Namespace](#) [► 179]

6.2.77 ResultValue2.I, V. Class

Result object of an asynchronous read. Implements the [ResultAccess](#) [► 3197]

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.ResultAds](#) [► 1116]

[TwinCAT.Ads.ResultValue](#) [► 1181].V.

[TwinCAT.Ads.ResultValue2.I, V.](#)

Namespace: [TwinCAT.Ads](#) [► 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public class ResultValue2<I, V> : ResultValue<V>
```

Type Parameters

I	The SymbolID / AddressInformation bound to the value.
V	The type of the read Value.

The ResultValue2.I, V. type exposes the following members.

Properties

	Name	Description
	ErrorCode [► 1120]	Gets the ADS Error code bound to this Result [► 1116] object. (Inherited from ResultAds [► 1116].)

	Name	Description
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116] .)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116] .)
	Source [▶ 1236]	Gets the source of the value (symbolic specifier/Address).
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116] .)
	Value [▶ 1185]	The value object. (Inherited from ResultValue.TValue. [▶ 1181] .)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116] .)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[TwinCAT.Ads Namespace \[▶ 179\]](#)

[TwinCAT.ValueAccess.ResultAccess \[▶ 3197\]](#)

6.2.77.1 ResultValue2.I, V. Properties

The [ResultValue2.I, V. \[▶ 1234\]](#) generic type exposes the following members.

Properties

	Name	Description
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116] .)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116] .)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116] .)
	Source [▶ 1236]	Gets the source of the value (symbolic specifier/Address).
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116] .)
	Value [▶ 1185]	The value object. (Inherited from ResultValue.TValue. [▶ 1181] .)

Reference

[ResultValue2.I, V. Class \[▶ 1234\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.77.1.1 ResultValue2.I, V..Source Property

Gets the source of the value (symbolic specifier/Address).

Namespace: [TwinCAT.Ads \[► 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public I Source { get; }
```

Property Value

Type: [I \[► 1234\]](#)

The source.

Reference

[ResultValue2.I, V. Class \[► 1234\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.77.2 ResultValue2.I, V. Methods

The [ResultValue2.I, V. \[► 1234\]](#) generic type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [► 1128]	Sets the error state of this ResultAds [► 1116] (Inherited from ResultAds [► 1116] .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultValue2.I, V. Class \[► 1234\]](#)

[TwinCAT.Ads Namespace \[► 179\]](#)

6.2.78 ResultWriteControl Class

Class ResultWriteControl. Implements the [ResultAds \[► 1116\]](#)

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.ResultAds](#) [[▶ 1116](#)]

[TwinCAT.Ads.ResultWriteControl](#)

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public class ResultWriteControl : ResultAds
```

The ResultWriteControl type exposes the following members.

Constructors

	Name	Description
	ResultWriteControl(ResultWriteControl) [▶ 1239]	Initializes a new instance of the ResultWriteControl class.
	ResultWriteControl(AmsAddress, Boolean, AdsErrorCode, AdsStateCommand, AdsState, AdsState, TimeSpan) [▶ 1239]	Initializes a new instance of the ResultWriteControl class.

Properties

	Name	Description
	Empty [▶ 1241]	Gets the empty ResultWriteControl
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Methods

	Name	Description
	Create [▶ 1242]	Creates an ResultWriteControl object.
	CreateFailed(AmsAddress,	Creates a failed ResultWriteControl object.

	Name	Description
	AdsErrorCode , AdsStateCommand [► 1243]	
	CreateFailed(AmsAddress, AdsErrorCode, AdsStateCommand, AdsState) [► 1244]	Creates a failed ResultWriteControl object.
	CreateFailed(AmsAddress, AdsErrorCode, AdsStateCommand, AdsState, AdsState) [► 1245]	Creates a failed ResultWriteControl object.
	CreateSucceeded [► 1246]	Creates a succeeding ResultWriteControl object.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [► 1128]	Sets the error state of this ResultAds [► 1116] (Inherited from ResultAds [► 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Fields

	Name	Description
	Address [► 1247]	The address
	Latency [► 1247]	The latency/duration until the Result state has been reached.
	Original [► 1247]	The original state before the request is sent.
	Reached [► 1248]	The reached AdsState [► 729] after the AdsStateCommand [► 1233] request has been sent (Resulting state)
	Requested [► 1248]	The AdsStateCommand [► 1233] (Request)
	RequestSucceeded [► 1249]	Gets an indication if the WriteControl request has been succeeded.

Reference

[TwinCAT.Ads Namespace](#) [► 179]

[TwinCAT.Ads.ResultAds](#) [► 1116]

6.2.78.1 ResultWriteControl Constructor

Overload List

	Name	Description
	ResultWriteControl(ResultWriteControl) [▶ 1239]	Initializes a new instance of the ResultWriteControl [▶ 1236] class.
	ResultWriteControl(AmsAddress, Boolean, AdsErrorCode, AdsStateCommand, AdsState, AdsState, TimeSpan) [▶ 1239]	Initializes a new instance of the ResultWriteControl [▶ 1236] class.

Reference

[ResultWriteControl Class](#) [[▶ 1236](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.78.1.1 ResultWriteControl Constructor (ResultWriteControl)

Initializes a new instance of the [ResultWriteControl](#) [[▶ 1236](#)] class.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected ResultWriteControl (
    ResultWriteControl copy
)
```

Parameters

copy	Type: TwinCAT.Ads.ResultWriteControl [▶ 1236] The copy.
------	--

Reference

[ResultWriteControl Class](#) [[▶ 1236](#)]

[ResultWriteControl Overload](#) [[▶ 1239](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.78.1.2 ResultWriteControl Constructor (AmsAddress, Boolean, AdsErrorCode, AdsStateCommand, AdsState, AdsState, TimeSpan)

Initializes a new instance of the [ResultWriteControl](#) [[▶ 1236](#)] class.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected ResultWriteControl(
    AmsAddress address,
    bool succeeded,
    AdsErrorCode errorCode,
    AdsStateCommand requested,
    AdsState original,
    AdsState reached,
    TimeSpan latency
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [▶ 752] The address.
succeeded	Type: System.Boolean if set to true [succeeded].
errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
requested	Type: TwinCAT.Ads.AdsStateCommand [▶ 1233] The requested.
original	Type: TwinCAT.Ads.AdsState [▶ 729] The original.
reached	Type: TwinCAT.Ads.AdsState [▶ 729] The reached.
latency	Type: System.TimeSpan The latency.

Reference

[ResultWriteControl Class](#) [▶ 1236]

[ResultWriteControl Overload](#) [▶ 1239]

[TwinCAT.Ads Namespace](#) [▶ 179]

6.2.78.2 ResultWriteControl Properties

The [ResultWriteControl](#) [▶ 1236] type exposes the following members.

Properties

	Name	Description
	Empty [▶ 1241]	Gets the empty ResultWriteControl [▶ 1236]
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Reference

[ResultWriteControl Class](#) [[▶ 1236](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.78.2.1 ResultWriteControl.Empty Property

Gets the empty [ResultWriteControl](#) [[▶ 1236](#)]

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultWriteControl Empty { get; }
```

Property Value

Type: [ResultWriteControl](#) [[▶ 1236](#)]

The empty.

Reference

[ResultWriteControl Class](#) [[▶ 1236](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.78.3 ResultWriteControl Methods

The [ResultWriteControl](#) [[▶ 1236](#)] type exposes the following members.

Methods

	Name	Description
	Create [▶ 1242]	Creates an ResultWriteControl [▶ 1236] object.
	CreateFailed(AmsAddress, AdsErrorCode, AdsStateCommand) [▶ 1243]	Creates a failed ResultWriteControl [▶ 1236] object.
	CreateFailed(AmsAddress, AdsErrorCode, AdsStateCommand, AdsState) [▶ 1244]	Creates a failed ResultWriteControl [▶ 1236] object.
	CreateFailed(AmsAddress, AdsErrorCode, AdsStateCommand, AdsState, AdsState) [▶ 1245]	Creates a failed ResultWriteControl [▶ 1236] object.

	Name	Description
	CreateSucceeded [▶ 1246]	Creates a succeeding ResultWriteControl [▶ 1236] object.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116] .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultWriteControl Class](#) [\[▶ 1236\]](#)

[TwinCAT.Ads Namespace](#) [\[▶ 179\]](#)

6.2.78.3.1 ResultWriteControl.Create Method

Creates an [ResultWriteControl](#) [\[▶ 1236\]](#) object.

Namespace: [TwinCAT.Ads](#) [\[▶ 179\]](#)

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultWriteControl Create(
    AmsAddress address,
    bool succeeded,
    AdsErrorCode errorCode,
    AdsStateCommand requested,
    AdsState original,
    AdsState reached,
    TimeSpan latency
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [▶ 752] The address.
succeeded	Type: System.Boolean if set to true [succeeded] .
errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
requested	Type: TwinCAT.Ads.AdsStateCommand [▶ 1233] The requested.
original	Type: TwinCAT.Ads.AdsState [▶ 729] The original.
reached	Type: TwinCAT.Ads.AdsState [▶ 729] The reached.

latency	Type: System.TimeSpan The latency.
---------	---

Return Value

Type: [ResultWriteControl](#) [[▶ 1236](#)]
ResultWriteControl.

Reference

- [ResultWriteControl Class](#) [[▶ 1236](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.78.3.2 ResultWriteControl.CreateFailed Method

Overload List

	Name	Description
	CreateFailed(AmsAddress, AdsErrorCode, AdsStateCommand) [▶ 1243]	Creates a failed ResultWriteControl [▶ 1236] object.
	CreateFailed(AmsAddress, AdsErrorCode, AdsStateCommand, AdsState) [▶ 1244]	Creates a failed ResultWriteControl [▶ 1236] object.
	CreateFailed(AmsAddress, AdsErrorCode, AdsStateCommand, AdsState, AdsState) [▶ 1245]	Creates a failed ResultWriteControl [▶ 1236] object.

Reference

- [ResultWriteControl Class](#) [[▶ 1236](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.78.3.2.1 ResultWriteControl.CreateFailed Method (AmsAddress, AdsErrorCode, AdsStateCommand)

Creates a failed [ResultWriteControl](#) [[▶ 1236](#)] object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultWriteControl CreateFailed(
    AmsAddress address,
    AdsErrorCode errorCode,
    AdsStateCommand requested
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [▶ 752] The address.
errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
requested	Type: TwinCAT.Ads.AdsStateCommand [▶ 1233] The requested.

Return Value

Type: [ResultWriteControl \[▶ 1236\]](#)
ResultWriteControl.

Reference

[ResultWriteControl Class \[▶ 1236\]](#)

[CreateFailed Overload \[▶ 1243\]](#)

[TwinCAT.Ads Namespace \[▶ 179\]](#)

6.2.78.3.2.2 ResultWriteControl.CreateFailed Method (AmsAddress, AdsErrorCode, AdsStateCommand, AdsState)

Creates a failed [ResultWriteControl \[▶ 1236\]](#) object.

Namespace: [TwinCAT.Ads \[▶ 179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultWriteControl CreateFailed(
    AmsAddress address,
    AdsErrorCode errorCode,
    AdsStateCommand requested,
    AdsState original
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [▶ 752] The address.
errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
requested	Type: TwinCAT.Ads.AdsStateCommand [▶ 1233] The requested.
original	Type: TwinCAT.Ads.AdsState [▶ 729] The original.

Return Value

Type: [ResultWriteControl](#) [[▶ 1236](#)]
 ResultWriteControl.

Reference

- [ResultWriteControl Class](#) [[▶ 1236](#)]
- [CreateFailed Overload](#) [[▶ 1243](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.78.3.2.3 ResultWriteControl.CreateFailed Method (AmsAddress, AdsErrorCode, AdsStateCommand, AdsState, AdsState)

Creates a failed [ResultWriteControl](#) [[▶ 1236](#)] object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultWriteControl CreateFailed(
    AmsAddress address,
    AdsErrorCode errorCode,
    AdsStateCommand requested,
    AdsState original,
    AdsState reached
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [▶ 752] The address.
errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
requested	Type: TwinCAT.Ads.AdsStateCommand [▶ 1233] The requested.
original	Type: TwinCAT.Ads.AdsState [▶ 729] The original.
reached	Type: TwinCAT.Ads.AdsState [▶ 729] The reached.

Return Value

Type: [ResultWriteControl](#) [[▶ 1236](#)]
 ResultWriteControl.

Reference

- [ResultWriteControl Class](#) [[▶ 1236](#)]
- [CreateFailed Overload](#) [[▶ 1243](#)]
- [TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.78.3 ResultWriteControl.CreateSucceeded Method

Creates a succeeding [ResultWriteControl](#) [[▶ 1236](#)] object.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultWriteControl CreateSucceeded(
    AmsAddress address,
    AdsStateCommand requested,
    AdsState original,
    AdsState reached,
    TimeSpan latency
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [▶ 752] The address.
requested	Type: TwinCAT.Ads.AdsStateCommand [▶ 1233] The requested.
original	Type: TwinCAT.Ads.AdsState [▶ 729] The original.
reached	Type: TwinCAT.Ads.AdsState [▶ 729] The reached.
latency	Type: System.TimeSpan The latency.

Return Value

Type: [ResultWriteControl](#) [[▶ 1236](#)]
ResultWriteControl.

Reference

[ResultWriteControl Class](#) [[▶ 1236](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.2.78.4 ResultWriteControl Fields

The [ResultWriteControl](#) [[▶ 1236](#)] type exposes the following members.

Fields

	Name	Description
	Address [▶ 1247]	The address
	Latency [▶ 1247]	The latency/duration until the Result state has been reached.
	Original [▶ 1247]	The original state before the request is sent.
	Reached [▶ 1248]	The reached AdsState [▶ 729] after the AdsStateCommand [▶ 1233] request has been sent (Resulting state)
	Requested [▶ 1248]	The AdsStateCommand [▶ 1233] (Request)
	RequestSucceeded [▶ 1249]	Gets an indication if the WriteControl request has been succeeded.

Reference

[ResultWriteControl Class](#) [[▶](#) 1236]

[TwinCAT.Ads Namespace](#) [[▶](#) 179]

6.2.78.4.1 ResultWriteControl.Address Field

The address

Namespace: [TwinCAT.Ads](#) [[▶](#) 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public readonly AmsAddress Address
```

Field Value

Type: [AmsAddress](#) [[▶](#) 752]

Reference

[ResultWriteControl Class](#) [[▶](#) 1236]

[TwinCAT.Ads Namespace](#) [[▶](#) 179]

6.2.78.4.2 ResultWriteControl.Latency Field

The latency/duration until the Result state has been reached.

Namespace: [TwinCAT.Ads](#) [[▶](#) 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public readonly TimeSpan Latency
```

Field Value

Type: [TimeSpan](#)

Reference

[ResultWriteControl Class](#) [[▶](#) 1236]

[TwinCAT.Ads Namespace](#) [[▶](#) 179]

6.2.78.4.3 ResultWriteControl.Original Field

The original state before the request is sent.

Namespace: [TwinCAT.Ads](#) [[▶](#) 179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public readonly AdsState Original
```

Field Value

Type: [AdsState](#) [[▶](#) [729](#)]

Reference

[ResultWriteControl Class](#) [[▶](#) [1236](#)]

[TwinCAT.Ads Namespace](#) [[▶](#) [179](#)]

6.2.78.4.4 ResultWriteControl.Reached Field

The reached [AdsState](#) [[▶](#) [729](#)] after the [AdsStateCommand](#) [[▶](#) [1233](#)] request has been sent (Resulting state)

Namespace: [TwinCAT.Ads](#) [[▶](#) [179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public readonly AdsState Reached
```

Field Value

Type: [AdsState](#) [[▶](#) [729](#)]

Reference

[ResultWriteControl Class](#) [[▶](#) [1236](#)]

[TwinCAT.Ads Namespace](#) [[▶](#) [179](#)]

6.2.78.4.5 ResultWriteControl.Requested Field

The [AdsStateCommand](#) [[▶](#) [1233](#)] (Request)

Namespace: [TwinCAT.Ads](#) [[▶](#) [179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public readonly AdsStateCommand Requested
```

Field Value

Type: [AdsStateCommand](#) [[▶](#) [1233](#)]

Reference

[ResultWriteControl Class](#) [[▶](#) [1236](#)]

[TwinCAT.Ads Namespace](#) [[▶](#) [179](#)]

6.2.78.4.6 ResultWriteControl.RequestSucceeded Field

Gets an indication if the WriteControl request has been succeeded.

Namespace: [TwinCAT.Ads](#) [[▶ 179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public readonly bool RequestSucceeded
```

Field Value

Type: [Boolean](#)

Reference





[ResultWriteControl Class](#) [[▶ 1236](#)]

[TwinCAT.Ads Namespace](#) [[▶ 179](#)]

6.3 TwinCAT.Ads.Reactive Namespace

Reactive Extensions for the ADS Client. All types within are contained in the ADS companion package "Beckhoff.TwinCAT.Ads.Reactive" which must be referenced seperately. <https://www.nuget.org/packages/Beckhoff.TwinCAT.Ads.Reactive/>

Classes

	Class	Description
	AdsClientExtensions [▶ 1249]	Extension class for AdsClient respective IAdsConnection [▶ 876] to provide reactive ADS extensions.
	AnyTypeExtensions [▶ 1281]	Extension class for IAdsConnection [▶ 876] to provide reactive ADS extensions (accessing symbol value sequences with the ANY_TYPE concept)
	SymbolValueNotification [▶ 1331]	Symbol Notification class
	ValueSymbolExtensions [▶ 1333]	Extension class for IAdsConnection [▶ 876] to provide reactive ADS extensions for accessing symbols that are loaded by the IAdsSymbolLoaderFactory

6.3.1 AdsClientExtensions Class

Extension class for AdsClient respective [IAdsConnection](#) [[▶ 876](#)] to provide reactive ADS extensions.

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.Reactive.AdsClientExtensions](#)

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]






















Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229


























Syntax**C#**



```
public static class AdsClientExtensions
```

The AdsClientExtensions type exposes the following members.

Methods

	Name	Description
  	<code>PollAdsState(IAdsConnection, IObservable.Unit.)</code> [▶ 1256]	Gets an observable sequence of AdsState [▶ 729]s via Polling.
  	<code>PollAdsState(IAdsConnection, TimeSpan)</code> [▶ 1257]	Gets an observable sequence of AdsState [▶ 729]s via Polling.
 	<code>PollAdsState2(IAdsConnection, IObservable.Unit.)</code> [▶ 1271]	Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling.
  	<code>PollAdsState2(IAdsConnection, TimeSpan)</code> [▶ 1272]	Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling.
 	<code>PollAdsState2Async(IAdsConnection, IObservable.Unit., Cancellation.Token)</code> [▶ 1274]	Gets an observable sequence of AdsState [▶ 729]s via Polling.
 	<code>PollAdsState2Async(IAdsConnection, TimeSpan, Cancellation.Token)</code> [▶ 1275]	Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling.
  	<code>PollAdsStateAsync(IAdsConnection, IObservable.Unit., Cancellation.Token)</code> [▶ 1259]	Gets an observable sequence of AdsState [▶ 729]s via Polling.
  	<code>PollAdsStateAsync(IAdsConnection, TimeSpan, Cancellation.Token)</code> [▶ 1260]	Gets an observable sequence of AdsState [▶ 729]s via Polling.

	Name	Description
 	PollDeviceState(IAdsConnection, IObservable.Unit) [▶ 1276]	Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling.
  	PollDeviceState(IAdsConnection, TimeSpan) [▶ 1276]	Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling.
  	PollDeviceStateAsync(IAdsConnection, IObservable.Unit, Cancellation.Token) [▶ 1278]	Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling.
  	PollDeviceStateAsync(IAdsConnection, TimeSpan, Cancellation.Token) [▶ 1279]	Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling.
  	WhenAdsStateChanges [▶ 1261]	Gets an observable sequence of AdsState [▶ 729]s.
 	WhenNotification(IAdsConnection, ISymbol) [▶ 1263]	Gets an observable sequence of Notification [▶ 1100]s.
  	WhenNotification(IAdsConnection, ISymbolCollection) [▶ 1264]	Gets an observable sequence of Notification [▶ 1100] objects.
  	WhenNotification(IAdsConnection, IList.ISymbol, NotificationSettings) [▶ 1268]	Gets an observable sequence of Notification [▶ 1100] objects.
 	WhenNotification(IAdsConnection, ISymbol, NotificationSettings) [▶ 1265]	Gets an observable sequence of SymbolValueNotification [▶ 1331]s.
	WhenSymbolVersionChanges(IAdsConnection) [▶ 1270]	Gets an observable sequence of SymbolVersion changed counts.

	Name	Description
 	WhenSymbolVersionChanges(IAdsConnection, IScheduler) [▶ 1270]	Gets an observable sequence of SymbolVersion changed counts.

Remarks

Reactive Extensions (Rx) are a library for composing asynchronous and event-based programs using observable sequences and LINQ-style query operators. Using Rx, developers represent asynchronous data streams with Observables, query asynchronous data streams using LINQ operators, and parameterize the concurrency in the asynchronous data streams using Schedulers. Simply put, Rx = Observables + LINQ + Schedulers. The ADS reactive extensions are built on top of this library to enable ADS Symbol and State Observables, seamlessly bound to the reactive extensions. To use the ADS reactive extensions the TwinCAT.Ads.Reactive Nuget package (or the included TwinCAT.Ads.Reactive.dll) must be referenced. ([Beckhoff.TwinCAT.Ads.Reactive package on Nuget](#)).

Examples

The following sample shows how observe Value changed Notifications with the reactive AdsClientExtensions

Observe changing ADS Symbols with reactive extensions.

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);

    int eventCount = 1;

    // Reactive Notification Handler
    var valueObserver = Observer.Create<SymbolValueNotification>(not =>
    {
        Console.WriteLine(string.Format("{0} {1:u} {2} = '{3}' ({4})", eventCount+
+ , not.TimeStamp, not.Symbol.InstancePath, not.Value, not.Symbol.DataType));
    });

    // Collect the symbols that are registered as Notification sources for their changed values.

    SymbolCollection notificationSymbols = new SymbolCollection();
    IArrayInstance taskInfo = (IArrayInstance)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo"];

    foreach(ISymbol element in taskInfo.Elements)
    {
        ISymbol cycleCount = element.SubSymbols["CycleCount"];
        ISymbol lastExecTime = element.SubSymbols["LastExecTime"];

        notificationSymbols.Add(cycleCount);
        notificationSymbols.Add(lastExecTime);
    }

    // Create a subscription for the first 200 Notifications on Symbol Value changes.
    IDisposable subscription = client.WhenNotification(notificationSymbols, NotificationSettings.Default).Take(200).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Examples

The following sample shows how observe [AdsState](#) [▶ 729] changed Notifications with the reactive AdsClientExtensions

Observe changing ADS states with reactive extensions.

```
// To Test the observer, Start/Stop the local PLC

using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.DefaultDynamic);

    // Reactive Notification Handler
    var valueObserver = Observer.Create<IList<AdsState>>(not =>
    {
        // onNext handler (handling the StateChange)
        AdsState oldValue = not[0];
        AdsState newValue = not[1];
        Console.WriteLine(string.Format("Changed ADSState from '{0}' --
> '{1}!", oldValue, newValue));
    },
    ex => Console.WriteLine($"Error: {ex.Message}"), // Error Handling
    () => Console.WriteLine("WhenAdsStateChanges completed!") // Observer has completed
    );

    // Create a subscription for the AdsState change and buffering 2 Values (for oldValue --
> newValue output).
    IDisposable subscription = client.WhenAdsStateChanges().Buffer(2,1).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[TwinCAT.Ads.Reactive Namespace \[▶ 1249\]](#)

[TwinCAT.Ads.Reactive.AnyTypeExtensions \[▶ 1281\]](#)

[TwinCAT.Ads.Reactive.ValueSymbolExtensions \[▶ 1333\]](#)



Also see about this

- 📖 [AdsClientExtensions.WhenNotification Method \(IAdsConnection, ISymbolCollection, NotificationSettings\) \[▶ 1266\]](#)

6.3.1.1 AdsClientExtensions Methods

The [AdsClientExtensions \[▶ 1249\]](#) type exposes the following members.

Methods

	Name	Description
	PollAdsState(IAdsConnection, IObservable.Unit) [▶ 1256]	Gets an observable sequence of AdsState [▶ 729] s via Polling.
	PollAdsState(IAdsConnection, TimeSpan) [▶ 1257]	Gets an observable sequence of AdsState [▶ 729] s via Polling.

	Name	Description
	PollAdsState2(IAdsConnection, IObservable.Unit.) [1271]	Gets an observable sequence of ResultReadAdsState [1146]s via Polling.
 	PollAdsState2(IAdsConnection, TimeSpan) [1272]	Gets an observable sequence of ResultReadAdsState [1146]s via Polling.
	PollAdsState2Async(IAdsConnection, IObservable.Unit, Cancellation.Token) [1274]	Gets an observable sequence of AdsState [729]s via Polling.
	PollAdsState2Async(IAdsConnection, TimeSpan, Cancellation.Token) [1275]	Gets an observable sequence of ResultReadAdsState [1146]s via Polling.
 	PollAdsStateAsync(IAdsConnection, IObservable.Unit, Cancellation.Token) [1259]	Gets an observable sequence of AdsState [729]s via Polling.
 	PollAdsStateAsync(IAdsConnection, TimeSpan, Cancellation.Token) [1260]	Gets an observable sequence of AdsState [729]s via Polling.
	PollDeviceState(IAdsConnection, IObservable.Unit.) [1276]	Gets an observable sequence of ResultReadDeviceState [1157]s via Polling.
 	PollDeviceState(IAdsConnection, TimeSpan) [1276]	Gets an observable sequence of ResultReadDeviceState [1157]s via Polling.
 	PollDeviceStateAsync(IAdsConnection, IObservable.Unit, Cancellation.Token) [1278]	Gets an observable sequence of ResultReadDeviceState [1157]s via Polling.


	Name	Description
  	PollDeviceStateAsync(IAdsConnection, TimeSpan, CancellationToken) [▶ 1279]	Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling.
  	WhenAdsStateChanges [▶ 1261]	Gets an observable sequence of AdsState [▶ 729]s.
 	WhenNotification(IAdsConnection, ISymbol) [▶ 1263]	Gets an observable sequence of Notification [▶ 1100]s.
  	WhenNotification(IAdsConnection, ISymbolCollection) [▶ 1264]	Gets an observable sequence of Notification [▶ 1100] objects.
  	WhenNotification(IAdsConnection, IList.ISymbol, NotificationSettings) [▶ 1268]	Gets an observable sequence of Notification [▶ 1100] objects.
 	WhenNotification(IAdsConnection, ISymbol, NotificationSettings) [▶ 1265]	Gets an observable sequence of SymbolValueNotification [▶ 1331]s.
 	WhenSymbolVersionChanges(IAdsConnection) [▶ 1270]	Gets an observable sequence of SymbolVersion changed counts.
 	WhenSymbolVersionChanges(IAdsConnection, IScheduler) [▶ 1270]	Gets an observable sequence of SymbolVersion changed counts.

Reference

[AdsClientExtensions Class](#) [[▶ 1249](#)]







[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

Also see about this

-  [AdsClientExtensions.WhenNotification Method \(IAdsConnection, ISymbolCollection, NotificationSettings\)](#) [[▶ 1266](#)]

6.3.1.1.1 AdsClientExtensions.PollAdsState Method

Overload List

	Name	Description
  	PollAdsState(IAdsConnection, IObservable.Unit.) [▶ 1256]	Gets an observable sequence of AdsState [▶ 729]s via Polling.
  	PollAdsState(IAdsConnection, TimeSpan) [▶ 1257]	Gets an observable sequence of AdsState [▶ 729]s via Polling.

Reference

[AdsClientExtensions Class](#) [▶ 1249]

[TwinCAT.Ads.Reactive Namespace](#) [▶ 1249]

6.3.1.1.1.1 AdsClientExtensions.PollAdsState Method (IAdsConnection, IObservable.Unit.)

Gets an observable sequence of [AdsState](#) [▶ 729]s via Polling.

Namespace: [TwinCAT.Ads.Reactive](#) [▶ 1249]

Assembly: [TwinCAT.Ads.Reactive](#) (in [TwinCAT.Ads.Reactive.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<AdsState> PollAdsState(
    this IAdsConnection client,
    IObservable<Unit> trigger
)
```

Parameters

client	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The client.
trigger	Type: System.IObservable.Unit. The polling trigger

Return Value

Type: [IObservable.AdsState](#) [▶ 729].
[IObservable<AdsState>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [▶ 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

The following sample shows how observe [AdsState](#) [► 729] via polling with the reactive [AdsClientExtensions](#) [► 1249]

Observe changing ADS States with reactive extensions.

```
// To Test the observer, Start/Stop the local PLC

using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.DefaultDynamic);

    // Reactive Change Handler
    var valueObserver = Observer.Create<IList<AdsState>>(not =>
    {
        AdsState oldValue = not[0];
        AdsState newValue = not[1];

        Console.WriteLine(string.Format("Changed ADSState from '{0}' --> '{1}!", oldValue, newValue));
    });

    // Create a subscription for the AdsState change and buffering 2 Values (for oldValue --
    > newValue output).
    IDisposable subscription = client.PollAdsState(TimeSpan.FromMilliseconds(200)).Buffer(2, 1).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[AdsClientExtensions Class](#) [► 1249]

[PollAdsState Overload](#) [► 1256]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

[AdsClientExtensions.WhenAdsStateChanges\(IAdsConnection\)](#) [► 1261]

6.3.1.1.1.2 AdsClientExtensions.PollAdsState Method (IAdsConnection, TimeSpan)

Gets an observable sequence of [AdsState](#) [► 729]s via Polling.

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<AdsState> PollAdsState(
    this IAdsConnection connection,
    TimeSpan period
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
period	Type: System.TimeSpan The period.

Return Value

Type: [IObservable.AdsState](#) [[▶ 729](#)].
IObservable<AdsState>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

The following sample shows how observe [AdsState](#) [[▶ 729](#)] via polling with the reactive [AdsClientExtensions](#) [[▶ 1249](#)]

Observe changing ADS States with reactive extensions.

```
// To Test the observer, Start/Stop the local PLC

using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.DefaultDynamic);

    // Reactive Change Handler
    var valueObserver = Observer.Create<IList<AdsState>>(not =>
    {
        AdsState oldValue = not[0];
        AdsState newValue = not[1];

        Console.WriteLine(string.Format("Changed ADSState from '{0}' --> '{1}!", oldValue, newValue));
    });

    // Create a subscription for the AdsState change and buffering 2 Values (for oldValue --
    > newValue output).
    IDisposable subscription = client.PollAdsState(TimeSpan.FromMilliseconds(200)).Buffer(2, 1).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[AdsClientExtensions Class](#) [[▶ 1249](#)]




[PollAdsState Overload](#) [[▶ 1256](#)]




[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

[AdsClientExtensions.WhenAdsStateChanges\(IAdsConnection\)](#) [[▶ 1261](#)]

6.3.1.1.2 AdsClientExtensions.PollAdsStateAsync Method

Overload List

	Name	Description
	PollAdsStateAsync(IAdsConnection,	Gets an observable sequence of AdsState [▶ 729]s via Polling.
	AdsConnection,	
		

	Name	Description
	IObservable.Unit, CancellationTokens [▶ 1259]	
  	PollAdsStateAsync(IAdsConnection, TimeSpan, CancellationTokens) [▶ 1260]	Gets an observable sequence of AdsState [▶ 729]s via Polling.

Reference

[AdsClientExtensions Class](#) [▶ 1249]

[TwinCAT.Ads.Reactive Namespace](#) [▶ 1249]

6.3.1.1.2.1 AdsClientExtensions.PollAdsStateAsync Method (IAdsConnection, IObservable.Unit, CancellationTokens)

Gets an observable sequence of [AdsState](#) [▶ 729]s via Polling.

Namespace: [TwinCAT.Ads.Reactive](#) [▶ 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<AdsState> PollAdsStateAsync(
    this IAdsConnection connection,
    IObservable<Unit> trigger,
    CancellationTokens cancel
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
trigger	Type: System.IObservable.Unit . The polling trigger
cancel	Type: System.Threading.CancellationTokens The cancellation token.

Return Value

Type: [IObservable.AdsState](#) [▶ 729].
IObservable<AdsState>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [▶ 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

The following sample shows how observe [AdsState](#) [▶ 729] via polling with the reactive [AdsClientExtensions](#) [▶ 1249]

Observe changing ADS States with reactive extensions.

```
// To Test the observer, Start/Stop the local PLC

using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.DefaultDynamic);

    // Reactive Change Handler
    var valueObserver = Observer.Create<IList<AdsState>>(not =>
    {
        AdsState oldValue = not[0];
        AdsState newValue = not[1];

        Console.WriteLine(string.Format("Changed ADSState from '{0}' --> '{1}!", oldValue, newValue));
    });

    // Create a subscription for the AdsState change and buffering 2 Values (for oldValue --
    > newValue output).
    IDisposable subscription = client.PollAdsState(TimeSpan.FromMilliseconds(200)).Buffer(2, 1).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference[AdsClientExtensions Class \[► 1249\]](#)[PollAdsStateAsync Overload \[► 1258\]](#)[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)[AdsClientExtensions.WhenAdsStateChanges\(IAdsConnection\) \[► 1261\]](#)**6.3.1.1.2.2 AdsClientExtensions.PollAdsStateAsync Method (IAdsConnection, TimeSpan, CancellationToken)**Gets an observable sequence of [AdsState \[► 729\]](#)s via Polling.**Namespace:** [TwinCAT.Ads.Reactive \[► 1249\]](#)**Assembly:** TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static IObservable<AdsState> PollAdsStateAsync(
    this IAdsConnection connection,
    TimeSpan interval,
    CancellationToken cancel
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
interval	Type: System.TimeSpan The interval.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [IObservable.AdsState](#) [[▶ 729](#)].
[IObservable<AdsState>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

The following sample shows how observe [AdsState](#) [[▶ 729](#)] via polling with the reactive [AdsClientExtensions](#) [[▶ 1249](#)]

Observe changing ADS States with reactive extensions.

```
// To Test the observer, Start/Stop the local PLC

using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.DefaultDynamic);

    // Reactive Change Handler
    var valueObserver = Observer.Create<IList<AdsState>>(not =>
    {
        AdsState oldValue = not[0];
        AdsState newValue = not[1];

        Console.WriteLine(string.Format("Changed ADSState from '{0}' --> '{1}!", oldValue, newValue));
    });

    // Create a subscription for the AdsState change and buffering 2 Values (for oldValue --
    > newValue output).
    IDisposable subscription = client.PollAdsState(TimeSpan.FromMilliseconds(200)).Buffer(2, 1).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[AdsClientExtensions Class](#) [[▶ 1249](#)]

[PollAdsStateAsync Overload](#) [[▶ 1258](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

[AdsClientExtensions.WhenAdsStateChanges\(IAdsConnection\)](#) [[▶ 1261](#)]

6.3.1.1.3 AdsClientExtensions.WhenAdsStateChanges Method

Gets an observable sequence of [AdsState](#) [[▶ 729](#)]s.

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<AdsState> WhenAdsStateChanges(
    this IAdsConnection client
)
```

Parameters

client	Type: TwinCAT.Ads.IAdsConnection [► 876] The client.
--------	---

Return Value

Type: [IObservable.AdsState](#) [► 729].
IObservable<AdsState>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

The following sample shows how observe [AdsState](#) [► 729] changed Notifications with the reactive [AdsClientExtensions](#) [► 1249]

Observe changing ADS States with reactive extensions.

```
// To Test the observer, Start/Stop the local PLC

using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.DefaultDynamic);

    // Reactive Notification Handler
    var valueObserver = Observer.Create<IList<AdsState>>(not =>
    {
        // onNext handler (handling the StateChange)
        AdsState oldValue = not[0];
        AdsState newValue = not[1];
        Console.WriteLine(string.Format("Changed ADSState from '{0}' --
> '{1}!", oldValue, newValue));
    },
    ex => Console.WriteLine($"Error: '{ex.Message}'"), // Error Handling
    () => Console.WriteLine("WhenAdsStateChanges completed!") // Observer has completed
    );

    // Create a subscription for the AdsState change and buffering 2 Values (for oldValue --
> newValue output).
    IDisposable subscription = client.WhenAdsStateChanges().Buffer(2,1).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference











[AdsClientExtensions Class](#) [► 1249]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

[AdsClientExtensions.PollAdsState\(IAdsConnection, IObservable.Unit.\)](#) [► 1256]

6.3.1.1.4 AdsClientExtensions.WhenNotification Method

Overload List


	Name	Description
 	WhenNotification(IAdsConnection, ISymbol) [▶ 1263]	Gets an observable sequence of Notification [▶ 1100]s.
  	WhenNotification(IAdsConnection, ISymbolCollection) [▶ 1264]	Gets an observable sequence of Notification [▶ 1100] objects.
  	WhenNotification(IAdsConnection, IList.ISymbol, NotificationSettings) [▶ 1268]	Gets an observable sequence of Notification [▶ 1100] objects.
 	WhenNotification(IAdsConnection, ISymbol, NotificationSettings) [▶ 1265]	Gets an observable sequence of SymbolValueNotification [▶ 1331]s.

Reference

[AdsClientExtensions Class](#) [[▶ 1249](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

Also see about this

-  [AdsClientExtensions.WhenNotification Method \(IAdsConnection, ISymbolCollection, NotificationSettings\)](#) [[▶ 1266](#)]

6.3.1.1.4.1 AdsClientExtensions.WhenNotification Method (IAdsConnection, ISymbol)

Gets an observable sequence of [Notification](#) [[▶ 1100](#)]s.

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: [TwinCAT.Ads.Reactive](#) (in [TwinCAT.Ads.Reactive.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Notification> WhenNotification(
    this IAdsConnection client,
    ISymbol symbol
)
```

Parameters

client	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The client.
--------	---

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol.
--------	--

Return Value

Type: [IObservable.Notification](#) [[▶ 1100](#)].
[IObservable<NotificationValue>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AdsClientExtensions Class](#) [[▶ 1249](#)]

[WhenNotification Overload](#) [[▶ 1263](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

[AdsClientExtensions.WhenNotification\(IAdsConnection, IList.ISymbol, NotificationSettings\)](#) [[▶ 1268](#)]

[AdsClientExtensions.WhenNotification\(IAdsConnection, ISymbol, NotificationSettings\)](#) [[▶ 1265](#)]

Also see about this

- [AdsClientExtensions.WhenNotification Method \(IAdsConnection, ISymbolCollection, NotificationSettings\)](#) [[▶ 1266](#)]

6.3.1.1.4.2 AdsClientExtensions.WhenNotification Method (IAdsConnection, ISymbolCollection)

Gets an observable sequence of [Notification](#) [[▶ 1100](#)] objects.

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<SymbolValueNotification> WhenNotification(
    this IAdsConnection connection,
    ISymbolCollection symbols
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The client.
symbols	Type: TwinCAT.TypeSystem.ISymbolCollection [▶ 2697] The symbols.

Return Value

Type: [IObservable.SymbolValueNotification](#) [[▶ 1331](#)].
[IObservable<NotificationValue>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

The following sample shows how observe Value changed Notifications with the reactive [AdsClientExtensions](#) [► 1249]

Observe changing ADS Symbols with reactive extensions.

```
// To Test the Observer run a project on the local PLC System (Port 851)

using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);

    int eventCount = 1;

    // Reactive Notification Handler
    var valueObserver = Observer.Create<SymbolValueNotification>(not =>
    {
        Console.WriteLine(string.Format("{0} {1:u} {2} = '{3}' ({4})", eventCount+
+, not.TimeStamp, not.Symbol.InstancePath, not.Value, not.Symbol.DataType));
    }
    );

    // Collect the symbols that are registered as Notification sources for their changed values.

    SymbolCollection notificationSymbols = new SymbolCollection();
    IArrayInstance taskInfo = (IArrayInstance)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskI
nfo"];

    foreach(ISymbol element in taskInfo.Elements)
    {
        ISymbol cycleCount = element.SubSymbols["CycleCount"];
        ISymbol lastExecTime = element.SubSymbols["LastExecTime"];

        notificationSymbols.Add(cycleCount);
        notificationSymbols.Add(lastExecTime);
    }

    // Create a subscription for the first 200 Notifications on Symbol Value changes.
    IDisposable subscription = client.WhenNotification(notificationSymbols, NotificationSettings.Defa
ult).Take(200).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[AdsClientExtensions Class](#) [► 1249]

[WhenNotification Overload](#) [► 1263]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

[AdsClientExtensions.WhenNotification\(IAdsConnection, ISymbol, NotificationSettings\)](#) [► 1265]

6.3.1.1.4.3 AdsClientExtensions.WhenNotification Method (IAdsConnection, ISymbol, NotificationSettings)

Gets an observable sequence of [SymbolValueNotification](#) [► 1331]s.

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<SymbolValueNotification> WhenNotification(
    this IAdsConnection client,
    ISymbol symbol,
    NotificationSettings settings
)
```

Parameters

client	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The client.
symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol to observe.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] Notification settings.

Return Value

Type: [IObservable.SymbolValueNotification](#) [[▶ 1331](#)].

IObservable<NotificationValue>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Exceptions

Exception	Condition
ArgumentNullException	symbol
ArgumentOutOfRangeException	Symbol is not an IValueSymbol - symbol

Reference

[AdsClientExtensions Class](#) [[▶ 1249](#)]

[WhenNotification Overload](#) [[▶ 1263](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

[AdsClientExtensions.WhenNotification\(IAdsConnection, IList.ISymbol, NotificationSettings\)](#) [[▶ 1268](#)]

Also see about this

- [AdsClientExtensions.WhenNotification Method \(IAdsConnection, ISymbolCollection, NotificationSettings\)](#) [[▶ 1266](#)]

6.3.1.1.4.4 AdsClientExtensions.WhenNotification Method (IAdsConnection, ISymbolCollection, NotificationSettings)

Gets an observable sequence of [Notification](#) [[▶ 1100](#)] objects.

Namespace: [TwinCAT.Ads.Reactive](#) [▶ 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public static IObservable<SymbolValueNotification> WhenNotification(  
    this IAdsConnection client,  
    ISymbolCollection symbols,  
    NotificationSettings settings  
)
```

Parameters

client	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The client.
symbols	Type: TwinCAT.TypeSystem.ISymbolCollection [▶ 2697] The symbols to observe.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.

Return Value

Type: [IObservable.SymbolValueNotification](#) [▶ 1331].
IObservable<NotificationValue>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [▶ 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

The following sample shows how observe Value changed Notifications with the reactive [AdsClientExtensions](#) [▶ 1249]

Observe changing ADS Symbols with reactive extensions.

```
// To Test the Observer run a project on the local PLC System (Port 851)  
using (AdsClient client = new AdsClient())  
{  
    // Connect to target  
    client.Connect(new AmsAddress(AmsNetId.Local, 851));  
  
    // Create Symbol information  
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);  
  
    int eventCount = 1;  
  
    // Reactive Notification Handler  
    var valueObserver = Observer.Create<SymbolValueNotification>(not =>  
    {  
        Console.WriteLine(string.Format("{0} {1:u} {2} = '{3}' ({4})", eventCount+  
+, not.TimeStamp, not.Symbol.InstancePath, not.Value, not.Symbol.DataType));  
    }  
    );  
  
    // Collect the symbols that are registered as Notification sources for their changed values.  
  
    SymbolCollection notificationSymbols = new SymbolCollection();  
    IArrayInstance taskInfo = (IArrayInstance)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskI  
nfo"];  
  
    foreach(ISymbol element in taskInfo.Elements)  
    {  
        ISymbol cycleCount = element.SubSymbols["CycleCount"];  
    }  
}
```

```

ISymbol lastExecTime = element.SubSymbols["LastExecTime"];

notificationSymbols.Add(cycleCount);
notificationSymbols.Add(lastExecTime);
}

// Create a subscription for the first 200 Notifications on Symbol Value changes.
IDisposable subscription = client.WhenNotification(notificationSymbols, NotificationSettings.Default).Take(200).Subscribe(valueObserver);

Console.ReadKey(); // Wait for Key press
subscription.Dispose(); // Dispose the Subscription
}

```

Reference

[AdsClientExtensions Class \[► 1249\]](#)

[WhenNotification Overload \[► 1263\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.1.1.4.5 AdsClientExtensions.WhenNotification Method (IAdsConnection, IList.ISymbol., NotificationSettings)

Gets an observable sequence of [Notification \[► 1100\]](#) objects.

Namespace: [TwinCAT.Ads.Reactive \[► 1249\]](#)

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public static IObservable<SymbolValueNotification> WhenNotification(
    this IAdsConnection client,
    IList<ISymbol> symbols,
    NotificationSettings settings
)

```

Parameters

client	Type: TwinCAT.Ads.IAdsConnection [► 876] The client.
symbols	Type: System.Collections.Generic.IList.ISymbol [► 2691] . The symbols to observe.
settings	Type: TwinCAT.Ads.NotificationSettings [► 1106] The Notification settings.

Return Value

Type: [IObservable.SymbolValueNotification \[► 1331\]](#).
IObservable<NotificationValue>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection \[► 876\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

The following sample shows how observe Value changed Notifications with the reactive [AdsClientExtensions \[► 1249\]](#)

Observe changing ADS Symbols with reactive extensions.

```
// To Test the Observer run a project on the local PLC System (Port 851)

using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);

    int eventCount = 1;

    // Reactive Notification Handler
    var valueObserver = Observer.Create<SymbolValueNotification>(not =>
    {
        Console.WriteLine(string.Format("{0} {1:u} {2} = '{3}' ({4})", eventCount+
+, not.TimeStamp, not.Symbol.InstancePath, not.Value, not.Symbol.DataType));
    }
    );

    // Collect the symbols that are registered as Notification sources for their changed values.

    SymbolCollection notificationSymbols = new SymbolCollection();
    IArrayInstance taskInfo = (IArrayInstance)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskI
nfo"];

    foreach(ISymbol element in taskInfo.Elements)
    {
        ISymbol cycleCount = element.SubSymbols["CycleCount"];
        ISymbol lastExecTime = element.SubSymbols["LastExecTime"];

        notificationSymbols.Add(cycleCount);
        notificationSymbols.Add(lastExecTime);
    }

    // Create a subscription for the first 200 Notifications on Symbol Value changes.
    IDisposable subscription = client.WhenNotification(notificationSymbols,NotificationSettings.Defa
ult).Take(200).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference



[AdsClientExtensions Class \[▶ 1249\]](#)

[WhenNotification Overload \[▶ 1263\]](#)

[TwinCAT.Ads.Reactive Namespace \[▶ 1249\]](#)

6.3.1.1.5 AdsClientExtensions.WhenSymbolVersionChanges Method

Overload List

	Name	Description
	WhenSymbolVersio nChanges(IAdsConn ection) [▶ 1270]	Gets an observable sequence of SymbolVersion changed counts.
	WhenSymbolVersio nChanges(IAdsConn ection, IScheduler) [▶ 1270]	Gets an observable sequence of SymbolVersion changed counts.

Reference

[AdsClientExtensions Class](#) [► 1249]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.1.1.5.1 AdsClientExtensions.WhenSymbolVersionChanges Method (IAdsConnection)

Gets an observable sequence of SymbolVersion changed counts.

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<int> WhenSymbolVersionChanges(  
    this IAdsConnection connection  
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The client.
------------	---

Return Value

Type: [IObservable.Int32](#).

Counter, unique only within the [WhenSymbolVersionChanges\(IAdsConnection\)](#) observable.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AdsClientExtensions Class](#) [► 1249]

[WhenSymbolVersionChanges Overload](#) [► 1269]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.1.1.5.2 AdsClientExtensions.WhenSymbolVersionChanges Method (IAdsConnection, IScheduler)

Gets an observable sequence of SymbolVersion changed counts.

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<int> WhenSymbolVersionChanges(  
    this IAdsConnection connection,  
    IScheduler? scheduler  
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The client.
scheduler	Type: IScheduler The scheduler.

Return Value

Type: [IObservable.Int32](#).
Counter, unique only within the [WhenSymbolVersionChanges\(IAdsConnection, IScheduler\)](#) observable.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

- [AdsClientExtensions Class](#) [[▶ 1249](#)]
- [WhenSymbolVersionChanges Overload](#) [[▶ 1269](#)]
- [TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

6.3.1.1.6 AdsClientExtensions.PollAdsState2 Method

Overload List

	Name	Description
	PollAdsState2(IAdsConnection, IObservable.Unit) [▶ 1271]	Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling.
	PollAdsState2(IAdsConnection, TimeSpan) [▶ 1272]	Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling.

Reference

- [AdsClientExtensions Class](#) [[▶ 1249](#)]
- [TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

6.3.1.1.6.1 AdsClientExtensions.PollAdsState2 Method (IAdsConnection, IObservable.Unit.)

Gets an observable sequence of [ResultReadAdsState](#) [[▶ 1146](#)]s via Polling.

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: [TwinCAT.Ads.Reactive](#) (in [TwinCAT.Ads.Reactive.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultReadAdsState> PollAdsState2(
    this IAdsConnection connection,
    IObservable<Unit> trigger
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
trigger	Type: System.IObservable .Unit. The polling trigger

Return Value

Type: [IObservable.ResultReadAdsState](#) [▶ 1146].
IObservable<AdsState>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [▶ 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AdsClientExtensions Class](#) [▶ 1249]

[PollAdsState2 Overload](#) [▶ 1271]

[TwinCAT.Ads.Reactive Namespace](#) [▶ 1249]

6.3.1.1.6.2 AdsClientExtensions.PollAdsState2 Method (IAdsConnection, TimeSpan)

Gets an observable sequence of [ResultReadAdsState](#) [▶ 1146]s via Polling.

Namespace: [TwinCAT.Ads.Reactive](#) [▶ 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultReadAdsState> PollAdsState2(
    this IAdsConnection connection,
    TimeSpan period
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
period	Type: System.TimeSpan The period.

Return Value

Type: [IObservable.ResultReadAdsState](#) [▶ 1146].
IObservable<AdsState>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [▶ 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

The following sample shows how observe [AdsState](#) [▶ 729] via polling with the reactive [AdsClientExtensions](#) [▶ 1249]

Observe changing ADS States with reactive extensions.

```
// To Test the observer, Start/Stop the local PLC

using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.DefaultDynamic);

    // Reactive Change Handler
    var valueObserver = Observer.Create<IList<AdsState>>(not =>
    {
        AdsState oldValue = not[0];
        AdsState newValue = not[1];

        Console.WriteLine(string.Format("Changed ADSState from '{0}' --> '{1}!", oldValue, newValue));
    });

    // Create a subscription for the AdsState change and buffering 2 Values (for oldValue --
    > newValue output).
    IDisposable subscription = client.PollAdsState(TimeSpan.FromMilliseconds(200)).Buffer(2, 1).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[AdsClientExtensions Class](#) [▶ 1249]

[PollAdsState2 Overload](#) [▶ 1271]

[TwinCAT.Ads.Reactive Namespace](#) [▶ 1249]

[AdsClientExtensions.WhenAdsStateChanges\(IAdsConnection\)](#) [▶ 1261]

6.3.1.1.7 AdsClientExtensions.PollAdsState2Async Method

Overload List

	Name	Description
	PollAdsState2Async(IAdsConnection,	Gets an observable sequence of AdsState [▶ 729]s via Polling.

	Name	Description
	IObservable.Unit, CancellationTokens [▶ 1274]	
	PollAdsState2Async(IAdsConnection, TimeSpan, CancellationTokens) [▶ 1275]	Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling.

Reference

[AdsClientExtensions Class](#) [▶ 1249]

[TwinCAT.Ads.Reactive Namespace](#) [▶ 1249]

6.3.1.1.7.1 AdsClientExtensions.PollAdsState2Async Method (IAdsConnection, IObservable.Unit, CancellationTokens)

Gets an observable sequence of [AdsState](#) [▶ 729]s via Polling.

Namespace: [TwinCAT.Ads.Reactive](#) [▶ 1249]

Assembly: [TwinCAT.Ads.Reactive](#) (in [TwinCAT.Ads.Reactive.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultReadAdsState> PollAdsState2Async(
    this IAdsConnection connection,
    IObservable<Unit> trigger,
    CancellationTokens cancel
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
trigger	Type: System.IObservable.Unit . The polling trigger
cancel	Type: System.Threading.CancellationTokens The cancellation token.

Return Value

Type: [IObservable.ResultReadAdsState](#) [▶ 1146].
[IObservable<AdsState>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [▶ 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AdsClientExtensions Class](#) [▶ 1249]

[PollAdsState2Async Overload](#) [▶ 1273]

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.1.1.7.2 **AdsClientExtensions.PollAdsState2Async Method (IAdsConnection, TimeSpan, CancellationToken)**

Gets an observable sequence of [ResultReadAdsState \[► 1146\]](#)s via Polling.

Namespace: [TwinCAT.Ads.Reactive \[► 1249\]](#)

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultReadAdsState> PollAdsState2Async(
    this IAdsConnection connection,
    TimeSpan interval,
    CancellationToken cancel
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
interval	Type: System.TimeSpan The interval.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [IObservable.ResultReadAdsState \[► 1146\]](#).
IObservable<ResultReadAdsState>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection \[► 876\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AdsClientExtensions Class \[► 1249\]](#)

[PollAdsState2Async Overload \[► 1273\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.1.1.8 **AdsClientExtensions.PollDeviceState Method**

Overload List

	Name	Description
	PollDeviceState(IAdsConnection, IObservable.Unit.) [► 1276]	Gets an observable sequence of ResultReadDeviceState [► 1157] s via Polling.

	Name	Description
	PollDeviceState(IAdsConnection, TimeSpan) [► 1276]	Gets an observable sequence of ResultReadDeviceState [► 1157] s via Polling.

Reference

[AdsClientExtensions Class \[► 1249\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.1.1.8.1 AdsClientExtensions.PollDeviceState Method (IAdsConnection, IObservable.Unit.)

Gets an observable sequence of [ResultReadDeviceState \[► 1157\]](#)s via Polling.

Namespace: [TwinCAT.Ads.Reactive \[► 1249\]](#)

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultReadDeviceState> PollDeviceState(
    this IAdsConnection connection,
    IObservable<Unit> trigger
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
trigger	Type: System.IObservable.Unit. The polling trigger

Return Value

Type: [IObservable.ResultReadDeviceState \[► 1157\]](#).
[IObservable<AdsState>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection \[► 876\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AdsClientExtensions Class \[► 1249\]](#)

[PollDeviceState Overload \[► 1275\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.1.1.8.2 AdsClientExtensions.PollDeviceState Method (IAdsConnection, TimeSpan)

Gets an observable sequence of [ResultReadDeviceState \[► 1157\]](#)s via Polling.

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultReadDeviceState> PollDeviceState(
    this IAdsConnection connection,
    TimeSpan period
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The client.
period	Type: System.TimeSpan The period.

Return Value

Type: [IObservable.ResultReadDeviceState](#) [[▶ 1157](#)].
IObservable<AdsState>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

The following sample shows how observe [AdsState](#) [[▶ 729](#)] via polling with the reactive [AdsClientExtensions](#) [[▶ 1249](#)]

Observe changing ADS States with reactive extensions.

```
// To Test the observer, Start/Stop the local PLC
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.DefaultDynamic);

    // Reactive Change Handler
    var valueObserver = Observer.Create<IList<AdsState>>(not =>
    {
        AdsState oldValue = not[0];
        AdsState newValue = not[1];

        Console.WriteLine(string.Format("Changed ADSState from '{0}' --> '{1}!", oldValue, newValue));
    });

    // Create a subscription for the AdsState change and buffering 2 Values (for oldValue --
    > newValue output).
    IDisposable subscription = client.PollAdsState(TimeSpan.FromMilliseconds(200)).Buffer(2, 1).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference[AdsClientExtensions Class](#) [[▶ 1249](#)][PollDeviceState Overload](#) [[▶ 1275](#)][TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)][AdsClientExtensions.WhenAdsStateChanges\(IAdsConnection\)](#) [[▶ 1261](#)]**6.3.1.1.9 AdsClientExtensions.PollDeviceStateAsync Method****Overload List**

	Name	Description
	PollDeviceStateAsync(IAdsConnection, IObservable.Unit, CancellationTokens) [▶ 1278]	Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling.
	PollDeviceStateAsync(IAdsConnection, TimeSpan, CancellationTokens) [▶ 1279]	Gets an observable sequence of ResultReadDeviceState [▶ 1157]s via Polling.

Reference[AdsClientExtensions Class](#) [[▶ 1249](#)][TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]**6.3.1.1.9.1 AdsClientExtensions.PollDeviceStateAsync Method (IAdsConnection, IObservable.Unit., CancellationTokens)**Gets an observable sequence of [ResultReadDeviceState](#) [[▶ 1157](#)]s via Polling.**Namespace:** [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]**Assembly:** TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static IObservable<ResultReadDeviceState> PollDeviceStateAsync(
    this IAdsConnection connection,
    IObservable<Unit> trigger,
    CancellationTokens cancel
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
trigger	Type: System.IObservable.Unit . The polling trigger

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [IObservable.ResultReadDeviceState](#) [[▶ 1157](#)].
IObservable<AdsState>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

The following sample shows how observe [AdsState](#) [[▶ 729](#)] via polling with the reactive [AdsClientExtensions](#) [[▶ 1249](#)]

Observe changing ADS States with reactive extensions.

```
// To Test the observer, Start/Stop the local PLC
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.DefaultDynamic);

    // Reactive Change Handler
    var valueObserver = Observer.Create<IList<AdsState>>(not =>
    {
        AdsState oldValue = not[0];
        AdsState newValue = not[1];

        Console.WriteLine(string.Format("Changed ADSState from '{0}' --> '{1}!", oldValue, newValue));
    });

    // Create a subscription for the AdsState change and buffering 2 Values (for oldValue --
    > newValue output).
    IDisposable subscription = client.PollAdsState(TimeSpan.FromMilliseconds(200)).Buffer(2, 1).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[AdsClientExtensions Class](#) [[▶ 1249](#)]

[PollDeviceStateAsync Overload](#) [[▶ 1278](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

[AdsClientExtensions.WhenAdsStateChanges\(IAdsConnection\)](#) [[▶ 1261](#)]

6.3.1.1.9.2 **AdsClientExtensions.PollDeviceStateAsync Method (IAdsConnection, TimeSpan, CancellationToken)**

Gets an observable sequence of [ResultReadDeviceState](#) [[▶ 1157](#)]s via Polling.

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultReadDeviceState> PollDeviceStateAsync(
    this IAdsConnection connection,
    TimeSpan interval,
    CancellationToken cancel
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The client.
interval	Type: System.TimeSpan The interval.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [IObservable.ResultReadDeviceState](#) [▶ [1157](#)].
[IObservable<AdsState>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [▶ [876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

The following sample shows how observe [AdsState](#) [▶ [729](#)] via polling with the reactive [AdsClientExtensions](#) [▶ [1249](#)]

Observe changing ADS States with reactive extensions.

```
// To Test the observer, Start/Stop the local PLC

using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.DefaultDynamic);

    // Reactive Change Handler
    var valueObserver = Observer.Create<IList<AdsState>>(not =>
    {
        AdsState oldValue = not[0];
        AdsState newValue = not[1];

        Console.WriteLine(string.Format("Changed ADSState from '{0}' --> '{1}!", oldValue, newValue));
    });

    // Create a subscription for the AdsState change and buffering 2 Values (for oldValue --
    > newValue output).
    IDisposable subscription = client.PollAdsState(TimeSpan.FromMilliseconds(200)).Buffer(2, 1).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[AdsClientExtensions Class](#) [▶ [1249](#)]

[PollDeviceStateAsync Overload \[▶ 1278\]](#)

[TwinCAT.Ads.Reactive Namespace \[▶ 1249\]](#)

[AdsClientExtensions.WhenAdsStateChanges\(IAdsConnection\) \[▶ 1261\]](#)

6.3.2 AnyTypeExtensions Class

Extension class for [IAdsConnection \[▶ 876\]](#) to provide reactive ADS extensions (accessing symbol value sequences with the ANY_TYPE concept)

Inheritance Hierarchy

[System.Object](#)

TwinCAT.Ads.Reactive.AnyTypeExtensions

Namespace: [TwinCAT.Ads.Reactive \[▶ 1249\]](#)











Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#






















```
public static class AnyTypeExtensions
```













Methods

	Name	Description
 	PollValues(IAdsConnection, String, Type, IObservable.Unit.) [▶ 1300]	Polls the symbol values on timepoints where the polling observable streams data / triggers
 	PollValues(IAdsConnection, String, Type, TimeSpan) [▶ 1301]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues(IAdsConnection, ISymbol, Type, IObservable.Unit.) [▶ 1310]	Polls the symbol values on timepoints where the polling observable streams data / triggers
 	PollValues(IAdsConnection, ISymbol, Type, TimeSpan) [▶ 1311]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues(IAdsConnection, String, Type, .Int32., TimeSpan) [▶ 1304]	Polls the symbol as value sequence of object values with a specified period time.

	Name	Description
 	<u>PollValues(IAdsCon</u> <u>nection, String,</u> <u>Type,</u> <u>IObservable.Unit.,</u> <u>Func.Exception,</u> <u>Object.)</u> [▶ 1305]	Polls the symbol values on timepoints where the polling observable streams data / triggers
 	<u>PollValues(IAdsCon</u> <u>nection, String,</u> <u>Type, TimeSpan,</u> <u>Func.Exception,</u> <u>Object.)</u> [▶ 1306]	Polls the symbol as value sequence of object values with a specified period time.
 	<u>PollValues(IAdsCon</u> <u>nection, ISymbol,</u> <u>Type, .Int32.,</u> <u>TimeSpan)</u> [▶ 1316]	Polls the symbol as value sequence of object values with a specified period time.
 	<u>PollValues(IAdsCon</u> <u>nection, ISymbol,</u> <u>Type,</u> <u>IObservable.Unit.,</u> <u>Func.Exception,</u> <u>Object.)</u> [▶ 1317]	Polls the symbol values on timepoints where the polling observable streams data / triggers
 	<u>PollValues(IAdsCon</u> <u>nection, ISymbol,</u> <u>Type, TimeSpan,</u> <u>Func.Exception,</u> <u>Object.)</u> [▶ 1317]	Polls the symbol as value sequence of object values with a specified period time.
 	<u>PollValues(IAdsCon</u> <u>nection, String,</u> <u>Type, .Int32.,</u> <u>IObservable.Unit.,</u> <u>Func.Exception,</u> <u>Object.)</u> [▶ 1306]	Polls the symbol values on time points where the polling observable streams data / triggers
 	<u>PollValues(IAdsCon</u> <u>nection, String,</u> <u>Type, .Int32.,</u> <u>TimeSpan,</u> <u>Func.Exception,</u> <u>Object.)</u> [▶ 1307]	Polls the symbol as value sequence of object values with a specified period time.
 	<u>PollValues(IAdsCon</u> <u>nection, ISymbol,</u> <u>Type, .Int32.,</u> <u>IObservable.Unit.,</u> <u>Func.Exception,</u> <u>Object.)</u> [▶ 1321]	Polls the symbol values on time points where the polling observable streams data / triggers

	Name	Description
 	PollValues(IAdsCon nection, ISymbol, Type, .Int32., TimeSpan, Func.Exception, Object.) [▶ 1322]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T. (IAdsConnection, String, IObservable.Unit.) [▶ 1294]	Polls the symbol values on timepoints where the polling observable streams data / triggers
 	PollValues.T. (IAdsConnection, String, TimeSpan) [▶ 1295]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T. (IAdsConnection, ISymbol, IObservable.Unit.) [▶ 1308]	Polls the symbol values on timepoints where the polling observable streams data / triggers
 	PollValues.T. (IAdsConnection, ISymbol, TimeSpan) [▶ 1309]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T. (IAdsConnection, String, .Int32., IObservable.Unit.) [▶ 1296]	Polls the symbol values on time points where the polling observable streams data / triggers
  	PollValues.T. (IAdsConnection, String, .Int32., TimeSpan) [▶ 1297]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T. (IAdsConnection, String, IObservable.Unit., Func.Exception, T.) [▶ 1298]	Polls the symbol values on timepoints where the polling observable streams data / triggers
 	PollValues.T. (IAdsConnection, String, TimeSpan, Func.Exception, T.) [▶ 1299]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T. (IAdsConnection, ISymbol, .Int32., IObservable.Unit.) [▶ 1312]	Polls the symbol values on time points where the polling observable streams data / triggers

	Name	Description
  	PollValues.T. (IAdsConnection , ISymbol , .Int32 , TimeSpan) [▶ 1313]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T. (IAdsConnection , ISymbol , IObservable.Unit , Func.Exception , T .) [▶ 1314]	Polls the symbol values on timepoints where the polling observable streams data / triggers
 	PollValues.T. (IAdsConnection , ISymbol , TimeSpan , Func.Exception , T .) [▶ 1315]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T. (IAdsConnection , String , .Int32 , IObservable.Unit , Func.Exception , T .) [▶ 1302]	Polls the symbol values on timepoints where the polling observable streams data / triggers
  	PollValues.T. (IAdsConnection , String , .Int32 , TimeSpan , Func.Exception , T .) [▶ 1303]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T. (IAdsConnection , ISymbol , .Int32 , IObservable.Unit , Func.Exception , T .) [▶ 1318]	Polls the symbol values on timepoints where the polling observable streams data / triggers
  	PollValues.T. (IAdsConnection , ISymbol , .Int32 , TimeSpan , Func.Exception , T .) [▶ 1319]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues2.T. (IAdsConnection , String , .Int32 , IObservable.Unit .) [▶ 1329]	Polls the symbol values on time points where the polling observable streams data / triggers
 	PollValues2.T. (IAdsConnection , ISymbol , .Int32 , IObservable.Unit .) [▶ 1330]	Polls the symbol values on timepoints where the polling observable streams data / triggers

	Name	Description
  	<u>WhenNotification(AdsConnection, String, Type, NotificationSettings)</u> [▶ 1324]	Creates an observable sequence of values that are created by ADS Notifications.
  	<u>WhenNotification.T.(IAdsConnection, String, NotificationSettings)</u> [▶ 1323]	Creates an observable sequence of values that are created by ADS Notifications.
  	<u>WriteValues.T.(IAdsConnection, String, IObservable.T.)</u> [▶ 1326]	Writes the sequence of values to the symbol specified by the instance path.
  	<u>WriteValues.T.(IAdsConnection, String, IObservable.T., Action.Exception.)</u> [▶ 1327]	Writes the sequence of values to the symbol specified by the instance path.

Remarks

Reactive Extensions (Rx) are a library for composing asynchronous and event-based programs using observable sequences and LINQ-style query operators. Using Rx, developers represent asynchronous data streams with Observables, query asynchronous data streams using LINQ operators, and parameterize the concurrency in the asynchronous data streams using Schedulers. Simply put, Rx = Observables + LINQ + Schedulers. The ADS reactive extensions are build on top of this library to enable ADS Symbol and State Observables, seamlessly bound to the reactive extensions. To use the ADS reactive extensions the TwinCAT.Ads.Reactive Nuget package (or the included TwinCAT.Ads.Reactive.dll) must be referenced. ([Beckhoff.TwinCAT.Ads.Reactive package on Nuget](#)).

Examples

Example1: Observe Value changed Notifications with the reactive AnyTypeExtensions

Observe a single changing ADS Symbols (Extended AdsNotifications, ANY_TYPE)

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Reactive Notification Handler
    var valueObserver = Observer.Create<ushort>(val =>
    {
        Console.WriteLine(string.Format("Value: {0}", val.ToString()));
    });

    // Turning ADS Notifications into sequences of Value Objects (Taking 20 Values)
    // and subscribe to them.
    IDisposable subscription = client.WhenNotification<ushort>("TwinCAT_SystemInfoVarList._TaskInfo.CycleCount", NotificationSettings.Default).Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Examples

Example2: Polling ANY_TYPE values.

Observe changing ADS Symbols by polling (Read Polling) (ANY_TYPE)

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo.CycleCount"];

    // Reactive Notification Handler
    var valueObserver = Observer.Create<object>(val =>
    {
        Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToString()));
    });

    // Take 20 Values in an Interval of 500ms
    IDisposable subscription = cycleCount.PollValues(TimeSpan.FromMilliseconds(500)).Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Examples

Write values sequentially.

Write sequences of values to the target (ANY_TYPE)

```
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information (Symbol 'i : INT' in PLC Global Variables list.
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol gvlIntSymbol = (IValueSymbol)symbolLoader.Symbols["GVL.i"];

    // Produces object (short) Values 0,1,2,3 ... in seconds period
    IObservable<object> timerObservable = Observable.Interval(TimeSpan.FromSeconds(1.0)).Select(i => (object)(short)i);

    // Take 10 Values (0..9) and write them to GVL.i
    IDisposable dispose = gvlIntSymbol.WriteValues(timerObservable.Take(10));

    Console.ReadKey(); // Wait for Key press
    dispose.Dispose(); // Dispose the Subscription
}
```

Reference










[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)


[TwinCAT.Ads.Reactive.AdsClientExtensions \[► 1249\]](#)

[TwinCAT.Ads.Reactive.ValueSymbolExtensions \[► 1333\]](#)
















6.3.2.1 AnyTypeExtensions Methods


Methods

	Name	Description
	<code>PollValues(IAdsCon nection, String, Type, IObservable.Unit.)</code> [▶ 1300]	Polls the symbol values on timepoints where the polling observable streams data / triggers
	<code>PollValues(IAdsCon nection, String, Type, TimeSpan)</code> [▶ 1301]	Polls the symbol as value sequence of object values with a specified period time.
	<code>PollValues(IAdsCon nection, ISymbol, Type, IObservable.Unit.)</code> [▶ 1310]	Polls the symbol values on timepoints where the polling observable streams data / triggers
	<code>PollValues(IAdsCon nection, ISymbol, Type, TimeSpan)</code> [▶ 1311]	Polls the symbol as value sequence of object values with a specified period time.
	<code>PollValues(IAdsCon nection, String, Type, .Int32., TimeSpan)</code> [▶ 1304]	Polls the symbol as value sequence of object values with a specified period time.
	<code>PollValues(IAdsCon nection, String, Type, IObservable.Unit., Func.Exception, Object.)</code> [▶ 1305]	Polls the symbol values on timepoints where the polling observable streams data / triggers
	<code>PollValues(IAdsCon nection, String, Type, TimeSpan, Func.Exception, Object.)</code> [▶ 1306]	Polls the symbol as value sequence of object values with a specified period time.
	<code>PollValues(IAdsCon nection, ISymbol, Type, .Int32., TimeSpan)</code> [▶ 1316]	Polls the symbol as value sequence of object values with a specified period time.
	<code>PollValues(IAdsCon nection, ISymbol, Type,</code>	Polls the symbol values on timepoints where the polling observable streams data / triggers

	Name	Description
	IObservable.Unit. , Func.Exception , Object.) [▶ 1317]	
 	PollValues(IAdsCon nection, ISymbol, Type, TimeSpan, Func.Exception, Object.) [▶ 1317]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues(IAdsCon nection, String, Type, .Int32., IObservable.Unit., Func.Exception, Object.) [▶ 1306]	Polls the symbol values on time points where the polling observable streams data / triggers
 	PollValues(IAdsCon nection, String, Type, .Int32., TimeSpan, Func.Exception, Object.) [▶ 1307]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues(IAdsCon nection, ISymbol, Type, .Int32., IObservable.Unit., Func.Exception, Object.) [▶ 1321]	Polls the symbol values on time points where the polling observable streams data / triggers
 	PollValues(IAdsCon nection, ISymbol, Type, .Int32., TimeSpan, Func.Exception, Object.) [▶ 1322]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T. (IAdsConnection, String, IObservable.Unit.) [▶ 1294]	Polls the symbol values on timepoints where the polling observable streams data / triggers
 	PollValues.T. (IAdsConnection, String, TimeSpan) [▶ 1295]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T. (IAdsConnection, ISymbol, IObservable.Unit.) [▶ 1308]	Polls the symbol values on timepoints where the polling observable streams data / triggers

	Name	Description
 	PollValues.T. (IAdsConnection , ISymbol , TimeSpan) [▶ 1309]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T. (IAdsConnection , String , .Int32. , IObservable.Unit.) [▶ 1296]	Polls the symbol values on time points where the polling observable streams data / triggers
  	PollValues.T. (IAdsConnection , String , .Int32. , TimeSpan) [▶ 1297]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T. (IAdsConnection , String , IObservable.Unit. , Func.Exception , T.) [▶ 1298]	Polls the symbol values on timepoints where the polling observable streams data / triggers
 	PollValues.T. (IAdsConnection , String , TimeSpan , Func.Exception , T.) [▶ 1299]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T. (IAdsConnection , ISymbol , .Int32. , IObservable.Unit.) [▶ 1312]	Polls the symbol values on time points where the polling observable streams data / triggers
  	PollValues.T. (IAdsConnection , ISymbol , .Int32. , TimeSpan) [▶ 1313]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T. (IAdsConnection , ISymbol , IObservable.Unit. , Func.Exception , T.) [▶ 1314]	Polls the symbol values on timepoints where the polling observable streams data / triggers
 	PollValues.T. (IAdsConnection , ISymbol , TimeSpan , Func.Exception , T.) [▶ 1315]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T. (IAdsConnection , String , .Int32. , 	Polls the symbol values on timepoints where the polling observable streams data / triggers

	Name	Description
	<u>IObservable.Unit,</u> <u>Func.Exception, T.)</u> [▶ 1302]	
  	<u>PollValues.T.</u> <u>(IAdsConnection,</u> <u>String, .Int32.,</u> <u>TimeSpan,</u> <u>Func.Exception, T.)</u> [▶ 1303]	Polls the symbol as value sequence of object values with a specified period time.
 	<u>PollValues.T.</u> <u>(IAdsConnection,</u> <u>ISymbol, .Int32.,</u> <u>IObservable.Unit,</u> <u>Func.Exception, T.)</u> [▶ 1318]	Polls the symbol values on timepoints where the polling observable streams data / triggers
  	<u>PollValues.T.</u> <u>(IAdsConnection,</u> <u>ISymbol, .Int32.,</u> <u>TimeSpan,</u> <u>Func.Exception, T.)</u> [▶ 1319]	Polls the symbol as value sequence of object values with a specified period time.
 	<u>PollValues2.T.</u> <u>(IAdsConnection,</u> <u>String, .Int32.,</u> <u>IObservable.Unit.)</u> [▶ 1329]	Polls the symbol values on time points where the polling observable streams data / triggers
 	<u>PollValues2.T.</u> <u>(IAdsConnection,</u> <u>ISymbol, .Int32.,</u> <u>IObservable.Unit.)</u> [▶ 1330]	Polls the symbol values on timepoints where the polling observable streams data / triggers
  	<u>WhenNotification(IA</u> <u>dsConnection,</u> <u>String, Type,</u> <u>NotificationSettings</u> <u>)</u> [▶ 1324]	Creates an observable sequence of values that are created by ADS Notifications.
  	<u>WhenNotification.T.</u> <u>(IAdsConnection,</u> <u>String,</u> <u>NotificationSettings</u> <u>)</u> [▶ 1323]	Creates an observable sequence of values that are created by ADS Notifications.
  	<u>WriteValues.T.</u> <u>(IAdsConnection,</u> <u>String,</u> <u>IObservable.T.)</u> [▶ 1326]	Writes the sequence of values to the symbol specified by the instance path.
 	<u>WriteValues.T.</u> <u>(IAdsConnection,</u> <u>String,</u>	Writes the sequence of values to the symbol specified by the instance path.

	Name	Description
	IObservable.T., Action.Exception.) [▶ 1327]	

Reference










[AnyTypeExtensions Class](#) [▶ 1281]

[TwinCAT.Ads.Reactive Namespace](#) [▶ 1249]












6.3.2.1.1 AnyTypeExtensions.PollValues Method

Overload List

	Name	Description
 	PollValues.T. (IAdsConnection, String, IObservable.Unit.) [▶ 1294]	Polls the symbol values on timepoints where the polling observable streams data / triggers
 	PollValues.T. (IAdsConnection, String, TimeSpan) [▶ 1295]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T. (IAdsConnection, ISymbol, IObservable.Unit.) [▶ 1308]	Polls the symbol values on timepoints where the polling observable streams data / triggers
 	PollValues.T. (IAdsConnection, ISymbol, TimeSpan) [▶ 1309]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues(IAdsConnection, String, Type, IObservable.Unit.) [▶ 1300]	Polls the symbol values on timepoints where the polling observable streams data / triggers
 	PollValues(IAdsConnection, String, Type, TimeSpan) [▶ 1301]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues(IAdsConnection, ISymbol, Type, IObservable.Unit.) [▶ 1310]	Polls the symbol values on timepoints where the polling observable streams data / triggers

	Name	Description
 	PollValues(IAdsConnection, ISymbol, Type, TimeSpan) [▶ 1311]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T.(IAdsConnection, String, .Int32., IObservable.Unit.) [▶ 1296]	Polls the symbol values on time points where the polling observable streams data / triggers
  	PollValues.T.(IAdsConnection, String, .Int32., TimeSpan) [▶ 1297]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T.(IAdsConnection, String, IObservable.Unit, Func.Exception, T.) [▶ 1298]	Polls the symbol values on timepoints where the polling observable streams data / triggers
 	PollValues.T.(IAdsConnection, String, TimeSpan, Func.Exception, T.) [▶ 1299]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T.(IAdsConnection, ISymbol, .Int32., IObservable.Unit.) [▶ 1312]	Polls the symbol values on time points where the polling observable streams data / triggers
  	PollValues.T.(IAdsConnection, ISymbol, .Int32., TimeSpan) [▶ 1313]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T.(IAdsConnection, ISymbol, IObservable.Unit, Func.Exception, T.) [▶ 1314]	Polls the symbol values on timepoints where the polling observable streams data / triggers
 	PollValues.T.(IAdsConnection, ISymbol, TimeSpan, Func.Exception, T.) [▶ 1315]	Polls the symbol as value sequence of object values with a specified period time.

	Name	Description
 	PollValues(IAdsCon nection, String, Type, .Int32., TimeSpan) [▶ 1304]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues(IAdsCon nection, String, Type, IObservable.Unit., Func.Exception, Object.) [▶ 1305]	Polls the symbol values on timepoints where the polling observable streams data / triggers
 	PollValues(IAdsCon nection, String, Type, TimeSpan, Func.Exception, Object.) [▶ 1306]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues(IAdsCon nection, ISymbol, Type, .Int32., TimeSpan) [▶ 1316]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues(IAdsCon nection, ISymbol, Type, IObservable.Unit., Func.Exception, Object.) [▶ 1317]	Polls the symbol values on timepoints where the polling observable streams data / triggers
 	PollValues(IAdsCon nection, ISymbol, Type, TimeSpan, Func.Exception, Object.) [▶ 1317]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T. (IAdsConnection, String, .Int32., IObservable.Unit., Func.Exception, T.) [▶ 1302]	Polls the symbol values on timepoints where the polling observable streams data / triggers
  	PollValues.T. (IAdsConnection, String, .Int32., TimeSpan, Func.Exception, T.) [▶ 1303]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T. (IAdsConnection, ISymbol, .Int32.,	Polls the symbol values on timepoints where the polling observable streams data / triggers

	Name	Description
	IObservable.Unit, Func.Exception, T.) [► 1318]	
  	PollValues.T. (IAdsConnection, ISymbol, .Int32., TimeSpan, Func.Exception, T.) [► 1319]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues(IAdsCon nection, String, Type, .Int32., IObservable.Unit, Func.Exception, Object.) [► 1306]	Polls the symbol values on time points where the polling observable streams data / triggers
 	PollValues(IAdsCon nection, String, Type, .Int32., TimeSpan, Func.Exception, Object.) [► 1307]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues(IAdsCon nection, ISymbol, Type, .Int32., IObservable.Unit, Func.Exception, Object.) [► 1321]	Polls the symbol values on time points where the polling observable streams data / triggers
 	PollValues(IAdsCon nection, ISymbol, Type, .Int32., TimeSpan, Func.Exception, Object.) [► 1322]	Polls the symbol as value sequence of object values with a specified period time.

Reference

[AnyTypeExtensions Class \[► 1281\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.2.1.1.1 AnyTypeExtensions.PollValues.T. Method (IAdsConnection, String, IObservable.Unit.)

Polls the symbol values on timepoints where the polling observable streams data / triggers

Namespace: [TwinCAT.Ads.Reactive \[► 1249\]](#)

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> PollValues<T>(
    this IAdsConnection connection,
    string instancePath,
    IObservable<Unit> trigger
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
instancePath	Type: System.String The instance path.
trigger	Type: System.IObservable.Unit . The Polling trigger

Type Parameters

T	The ANY_TYPE compatible .NET Type.
---	------------------------------------

Return Value

Type: [IObservable.T](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class](#) [► 1281]

[PollValues Overload](#) [► 1291]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.2.1.1.2 AnyTypeExtensions.PollValues.T. Method (IAdsConnection, String, TimeSpan)

Polls the symbol as value sequence of object values with a specified period time.

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: [TwinCAT.Ads.Reactive](#) (in [TwinCAT.Ads.Reactive.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> PollValues<T>(
    this IAdsConnection connection,
    string instancePath,
    TimeSpan period
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
instancePath	Type: System.String The instance path.
period	Type: System.TimeSpan The period.

Type Parameters

T	The ANY_TYPE compatible .NET Type.
---	------------------------------------

Return Value

Type: [IObservable.T](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class](#) [► 1281]

[PollValues Overload](#) [► 1291]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.2.1.1.3 **AnyTypeExtensions.PollValues.T. Method (IAdsConnection, String, .Int32., IObservable.Unit.)**

Polls the symbol values on time points where the polling observable streams data / triggers

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: [TwinCAT.Ads.Reactive](#) (in [TwinCAT.Ads.Reactive.dll](#)) Version: 6.0.328+39e3229

Syntax**C#**

```
public static IObservable<T> PollValues<T>(
    this IAdsConnection connection,
    string instancePath,
    int[]? args,
    IObservable<Unit> trigger
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
instancePath	Type: System.String The instance path.
args	Type: .System.Int32 . ANY_TYPE arguments

trigger	Type: System.IObservable.Unit . The Polling trigger
---------	--

Type Parameters

T	The ANY_TYPE compatible .NET Type.
---	------------------------------------

Return Value

Type: [IObservable.T](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

- [AnyTypeExtensions Class](#) [[▶ 1281](#)]
- [PollValues Overload](#) [[▶ 1291](#)]
- [TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

6.3.2.1.1.4 AnyTypeExtensions.PollValues.T. Method (IAdsConnection, String, .Int32., TimeSpan)

Polls the symbol as value sequence of object values with a specified period time.

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]
Assembly: [TwinCAT.Ads.Reactive](#) (in [TwinCAT.Ads.Reactive.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> PollValues<T>(
    this IAdsConnection connection,
    string instancePath,
    int[]? args,
    TimeSpan period
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
instancePath	Type: System.String The instance path.
args	Type: .System.Int32 . ANY_TYPE arguments.
period	Type: System.TimeSpan The period.

Type Parameters

T	The ANY_TYPE compatible .NET Type.
---	------------------------------------

Return Value

Type: [IObservable.T](#).
 IObservable<System.Object>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

Polling ANY_TYPE values.

Observe changing ADS Symbols by polling (Read Polling) (ANY_TYPE)

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo.CycleCount"];

    // Reactive Notification Handler
    var valueObserver = Observer.Create<object>(val =>
    {
        Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToString()));
    });

    // Take 20 Values in an Interval of 500ms
    IDisposable subscription = cycleCount.PollValues(TimeSpan.FromMilliseconds(500)).Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[AnyTypeExtensions Class](#) [[▶ 1281](#)]

[PollValues Overload](#) [[▶ 1291](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

6.3.2.1.1.5 AnyTypeExtensions.PollValues.T. Method (IAdsConnection, String, IObservable.Unit., Func.Exception, T.)

Polls the symbol values on timepoints where the polling observable streams data / triggers

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> PollValues<T>(
    this IAdsConnection connection,
    string instancePath,
```

```
IObservable<Unit> trigger,
Func<Exception, T>? errorHandler
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
instancePath	Type: System.String The instance path.
trigger	Type: System.IObservable.Unit . The Polling trigger
errorHandler	Type: System.Func.Exception, T . The error handler.

Type Parameters

T	The ANY_TYPE compatible .NET Type.
---	------------------------------------

Return Value

Type: [IObservable.T](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [▶ 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

- [AnyTypeExtensions Class](#) [▶ 1281]
- [PollValues Overload](#) [▶ 1291]
- [TwinCAT.Ads.Reactive Namespace](#) [▶ 1249]

6.3.2.1.1.6 AnyTypeExtensions.PollValues.T. Method (IAdsConnection, String, TimeSpan, Func.Exception, T.)

Polls the symbol as value sequence of object values with a specified period time.

Namespace: [TwinCAT.Ads.Reactive](#) [▶ 1249]
Assembly: [TwinCAT.Ads.Reactive](#) (in [TwinCAT.Ads.Reactive.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> PollValues<T>(
    this IAdsConnection connection,
    string instancePath,
    TimeSpan period,
    Func<Exception, T>? errorHandler
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
------------	---

instancePath	Type: System.String The instance path.
period	Type: System.TimeSpan The period.
errorHandler	Type: System.Func.Exception, T . The error handler.

Type Parameters

T	The ANY_TYPE compatible .NET Type.
---	------------------------------------

Return Value

Type: [IObservable.T](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class](#) [► 1281]

[PollValues Overload](#) [► 1291]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.2.1.1.7 AnyTypeExtensions.PollValues Method (IAdsConnection, String, Type, IObservable.Unit.)

Polls the symbol values on timepoints where the polling observable streams data / triggers

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object> PollValues(
    this IAdsConnection connection,
    string instancePath,
    Type type,
    IObservable<Unit> trigger
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
instancePath	Type: System.String The instance path.
type	Type: System.Type The data type of the symbol (ANY_TYPE)
trigger	Type: System.IObservable.Unit . The Polling trigger

Return Value

Type: [IObservable.Object](#).
 IObservable<System.Object>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class](#) [► 1281]

[PollValues Overload](#) [► 1291]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.2.1.1.8 AnyTypeExtensions.PollValues Method (IAdsConnection, String, Type, TimeSpan)

Polls the symbol as value sequence of object values with a specified period time.

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object> PollValues(
    this IAdsConnection connection,
    string instancePath,
    Type type,
    TimeSpan period
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
instancePath	Type: System.String The instance path.
type	Type: System.Type The data type of the symbol (ANY_TYPE)
period	Type: System.TimeSpan The period.

Return Value

Type: [IObservable.Object](#).
 IObservable<System.Object>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class \[► 1281\]](#)

[PollValues Overload \[► 1291\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.2.1.1.9 AnyTypeExtensions.PollValues.T. Method (IAdsConnection, String, .Int32., IObservable.Unit., Func.Exception, T.)

Polls the symbol values on timepoints where the polling observable streams data / triggers

Namespace: [TwinCAT.Ads.Reactive \[► 1249\]](#)

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> PollValues<T>(
    this IAdsConnection connection,
    string instancePath,
    int[]? args,
    IObservable<Unit> trigger,
    Func<Exception, T>? errorHandler
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
instancePath	Type: System.String The instance path.
args	Type: .System.Int32 . ANY_TYPE arguments
trigger	Type: System.IObservable.Unit . The Polling trigger
errorHandler	Type: System.Func.Exception, T . The error handler.

Type Parameters

T	The ANY_TYPE compatible .NET Type.
---	------------------------------------

Return Value

Type: [IObservable.T](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection \[► 876\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class \[► 1281\]](#)

[PollValues Overload \[► 1291\]](#)

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

6.3.2.1.1.10 AnyTypeExtensions.PollValues.T. Method (IAdsConnection, String, .Int32., TimeSpan, Func.Exception, T.)

Polls the symbol as value sequence of object values with a specified period time.

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> PollValues<T>(
    this IAdsConnection connection,
    string instancePath,
    int[]? args,
    TimeSpan period,
    Func<Exception, T>? errorHandler
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
instancePath	Type: System.String The instance path.
args	Type: .System.Int32 . ANY_TYPE arguments.
period	Type: System.TimeSpan The period.
errorHandler	Type: System.Func.Exception, T . The error handler.

Type Parameters

T	The ANY_TYPE compatible .NET Type.
---	------------------------------------

Return Value

Type: [IObservable.T](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

Polling ANY_TYPE values.

Observe changing ADS Symbols by polling (Read Polling) (ANY_TYPE)

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
```

```

var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo.CycleCount"];

// Reactive Notification Handler
var valueObserver = Observer.Create<object>(val =>
{
    Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToString()));
}
);

// Take 20 Values in an Interval of 500ms
IDisposable subscription = cycleCount.PollValues(TimeSpan.FromMilliseconds(500)).Take(20).Subscribe(valueObserver);

Console.ReadKey(); // Wait for Key press
subscription.Dispose(); // Dispose the Subscription
}

```

Reference

[AnyTypeExtensions Class \[► 1281\]](#)

[PollValues Overload \[► 1291\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.2.1.1.11 AnyTypeExtensions.PollValues Method (IAdsConnection, String, Type, .Int32., TimeSpan)

Polls the symbol as value sequence of object values with a specified period time.

Namespace: [TwinCAT.Ads.Reactive \[► 1249\]](#)

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public static IObservable<Object> PollValues(
    this IAdsConnection connection,
    string instancePath,
    Type type,
    int[]? args,
    TimeSpan period
)

```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
instancePath	Type: System.String The instance path.
type	Type: System.Type The data type of the symbol (ANY_TYPE)
args	Type: .System.Int32 . The ANY_TYPE arguments.
period	Type: System.TimeSpan The period.

Return Value

Type: [IObservable.Object](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [▶ 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class](#) [▶ 1281]

[PollValues Overload](#) [▶ 1291]

[TwinCAT.Ads.Reactive Namespace](#) [▶ 1249]

6.3.2.1.1.12 AnyTypeExtensions.PollValues Method (IAdsConnection, String, Type, IObservable.Unit., Func.Exception, Object.)

Polls the symbol values on timepoints where the polling observable streams data / triggers

Namespace: [TwinCAT.Ads.Reactive](#) [▶ 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object> PollValues(
    this IAdsConnection connection,
    string instancePath,
    Type type,
    IObservable<Unit> trigger,
    Func<Exception, Object>? errorHandler
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
instancePath	Type: System.String The instance path.
type	Type: System.Type The data type of the symbol (ANY_TYPE)
trigger	Type: System.IObservable.Unit. The Polling trigger
errorHandler	Type: System.Func.Exception, Object. The error handler.

Return Value

Type: [IObservable.Object.](#)
[IObservable<System.Object>.](#)

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [▶ 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class](#) [▶ 1281]

[PollValues Overload \[► 1291\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.2.1.1.13 AnyTypeExtensions.PollValues Method (IAdsConnection, String, Type, TimeSpan, Func.Exception, Object.)

Polls the symbol as value sequence of object values with a specified period time.

Namespace: [TwinCAT.Ads.Reactive \[► 1249\]](#)

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object> PollValues(
    this IAdsConnection connection,
    string instancePath,
    Type type,
    TimeSpan period,
    Func<Exception, Object>? errorHandler
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
instancePath	Type: System.String The instance path.
type	Type: System.Type The data type of the symbol (ANY_TYPE)
period	Type: System.TimeSpan The period.
errorHandler	Type: System.Func.Exception, Object . The error handler.

Return Value

Type: [IObservable.Object](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection \[► 876\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class \[► 1281\]](#)

[PollValues Overload \[► 1291\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.2.1.1.14 AnyTypeExtensions.PollValues Method (IAdsConnection, String, Type, .Int32., IObservable.Unit., Func.Exception, Object.)

Polls the symbol values on time points where the polling observable streams data / triggers

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object> PollValues(
    this IAdsConnection connection,
    string instancePath,
    Type type,
    int[]? args,
    IObservable<Unit> trigger,
    Func<Exception, Object>? errorHandler
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
instancePath	Type: System.String The instance path.
type	Type: System.Type The data type of the symbol (ANY_TYPE)
args	Type: .System.Int32 . The ANY_TYPE arguments.
trigger	Type: System.IObservable.Unit . The Polling trigger
errorHandler	Type: System.Func.Exception, Object . The error handler.

Return Value

Type: [IObservable.Object](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class](#) [[▶ 1281](#)]

[PollValues Overload](#) [[▶ 1291](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

6.3.2.1.1.15 AnyTypeExtensions.PollValues Method (IAdsConnection, String, Type, .Int32., TimeSpan, Func.Exception, Object.)

Polls the symbol as value sequence of object values with a specified period time.

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object> PollValues(
    this IAdsConnection connection,
    string instancePath,
    Type type,
    int[]? args,
    TimeSpan period,
    Func<Exception, Object>? errorHandler
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
instancePath	Type: System.String The instance path.
type	Type: System.Type The data type of the symbol (ANY_TYPE)
args	Type: .System.Int32 . The ANY_TYPE arguments.
period	Type: System.TimeSpan The period.
errorHandler	Type: System.Func.Exception , Object . The error handler.

Return Value

Type: [IObservable.Object](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class](#) [[▶ 1281](#)]

[PollValues Overload](#) [[▶ 1291](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

6.3.2.1.1.16 AnyTypeExtensions.PollValues.T. Method (IAdsConnection, ISymbol, IObservable.Unit.)

Polls the symbol values on timepoints where the polling observable streams data / triggers

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> PollValues<T>(
    this IAdsConnection connection,
    ISymbol symbol,
    IObservable<Unit> trigger
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol.
trigger	Type: System.IObservable.Unit . The Polling trigger

Type Parameters

T	The ANY_TYPE compatible .NET Type.
---	------------------------------------

Return Value

Type: [IObservable.T](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class](#) [► 1281]

[PollValues Overload](#) [► 1291]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.2.1.1.17 AnyTypeExtensions.PollValues.T. Method (IAdsConnection, ISymbol, TimeSpan)

Polls the symbol as value sequence of object values with a specified period time.

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> PollValues<T>(
    this IAdsConnection connection,
    ISymbol symbol,
    TimeSpan period
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol.
period	Type: System.TimeSpan The period.

Type Parameters

T	The ANY_TYPE compatible .NET Type.
---	------------------------------------

Return Value

Type: [IObservable.T](#).
[IObservable](#)<System.Object>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class](#) [► 1281]

[PollValues Overload](#) [► 1291]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.2.1.18 **AnyTypeExtensions.PollValues Method (IAdsConnection, ISymbol, Type, IObservable.Unit.)**

Polls the symbol values on timepoints where the polling observable streams data / triggers

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: [TwinCAT.Ads.Reactive](#) (in [TwinCAT.Ads.Reactive.dll](#)) Version: 6.0.328+39e3229

Syntax**C#**

```
public static IObservable<Object> PollValues(
    this IAdsConnection connection,
    ISymbol symbol,
    Type type,
    IObservable<Unit> trigger
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol.
type	Type: System.Type The data type of the symbol (ANY_TYPE)

trigger	Type: System.IObservable.Unit . The Polling trigger
---------	--

Return Value

Type: [IObservable.Object](#).
IObservable<System.Object>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class](#) [► 1281]

[PollValues Overload](#) [► 1291]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.2.1.1.19 AnyTypeExtensions.PollValues Method (IAdsConnection, ISymbol, Type, TimeSpan)

Polls the symbol as value sequence of object values with a specified period time.

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object> PollValues(
    this IAdsConnection connection,
    ISymbol symbol,
    Type type,
    TimeSpan period
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol.
type	Type: System.Type The data type of the symbol (ANY_TYPE)
period	Type: System.TimeSpan The period.

Return Value

Type: [IObservable.Object](#).
IObservable<System.Object>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class](#) [► 1281]

[PollValues Overload](#) [► 1291]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.2.1.1.20 AnyTypeExtensions.PollValues.T. Method (IAdsConnection, ISymbol, .Int32., IObservable.Unit.)

Polls the symbol values on time points where the polling observable streams data / triggers

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> PollValues<T>(
    this IAdsConnection connection,
    ISymbol symbol,
    int[]? args,
    IObservable<Unit> trigger
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol.
args	Type: .System.Int32 . ANY_TYPE arguments
trigger	Type: System.IObservable .Unit. The Polling trigger

Type Parameters

T	The ANY_TYPE compatible .NET Type.
---	------------------------------------

Return Value

Type: [IObservable](#).T.
[IObservable](#)<System.Object>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class](#) [▶ 1281]

[PollValues Overload](#) [▶ 1291]

[TwinCAT.Ads.Reactive Namespace](#) [▶ 1249]

6.3.2.1.1.21 AnyTypeExtensions.PollValues.T. Method (IAdsConnection, ISymbol, .Int32., TimeSpan)

Polls the symbol as value sequence of object values with a specified period time.

Namespace: [TwinCAT.Ads.Reactive](#) [▶ 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> PollValues<T>(
    this IAdsConnection connection,
    ISymbol symbol,
    int[]? args,
    TimeSpan period
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol.
args	Type: .System.Int32 . ANY_TYPE arguments.
period	Type: System.TimeSpan The period.

Type Parameters

T	The ANY_TYPE compatible .NET Type.
---	------------------------------------

Return Value

Type: [IObservable.T](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [▶ 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

Polling ANY_TYPE values.

Observe changing ADS Symbols by polling (Read Polling) (ANY_TYPE)

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
```

```

client.Connect(new AmsAddress(AmsNetId.Local, 851));

// Create Symbol information
var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo.CycleCount"];

// Reactive Notification Handler
var valueObserver = Observer.Create<object>(val =>
{
    Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToString()));
}
);

// Take 20 Values in an Interval of 500ms
IDisposable subscription = cycleCount.PollValues(TimeSpan.FromMilliseconds(500)).Take(20).Subscribe(valueObserver);

Console.ReadKey(); // Wait for Key press
subscription.Dispose(); // Dispose the Subscription
}

```

Reference

[AnyTypeExtensions Class \[► 1281\]](#)

[PollValues Overload \[► 1291\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.2.1.1.22 AnyTypeExtensions.PollValues.T. Method (IAdsConnection, ISymbol, IObservable.Unit., Func.Exception, T.)

Polls the symbol values on timepoints where the polling observable streams data / triggers

Namespace: [TwinCAT.Ads.Reactive \[► 1249\]](#)

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public static IObservable<T> PollValues<T>(
    this IAdsConnection connection,
    ISymbol symbol,
    IObservable<Unit> trigger,
    Func<Exception, T>? errorHandler
)

```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol.
trigger	Type: System.IObservable.Unit. The Polling trigger
errorHandler	Type: System.Func.Exception, T. The error handler.

Type Parameters

T	The ANY_TYPE compatible .NET Type.
---	------------------------------------

Return Value

Type: [IObservable.T](#).
IObservable<System.Object>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

- [AnyTypeExtensions Class](#) [[▶ 1281](#)]
- [PollValues Overload](#) [[▶ 1291](#)]
- [TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

6.3.2.1.1.23 AnyTypeExtensions.PollValues.T. Method (IAdsConnection, ISymbol, TimeSpan, Func.Exception, T.)

Polls the symbol as value sequence of object values with a specified period time.

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> PollValues<T>(
    this IAdsConnection connection,
    ISymbol symbol,
    TimeSpan period,
    Func<Exception, T>? errorHandler
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol.
period	Type: System.TimeSpan The period.
errorHandler	Type: System.Func.Exception, T . The error handler.

Type Parameters

T	The ANY_TYPE compatible .NET Type.
---	------------------------------------

Return Value

Type: [IObservable.T](#).
IObservable<System.Object>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class](#) [► 1281]

[PollValues Overload](#) [► 1291]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.2.1.1.24 AnyTypeExtensions.PollValues Method (IAdsConnection, ISymbol, Type, .Int32., TimeSpan)

Polls the symbol as value sequence of object values with a specified period time.

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object> PollValues(
    this IAdsConnection connection,
    ISymbol symbol,
    Type type,
    int[]? args,
    TimeSpan period
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol.
type	Type: System.Type The data type of the symbol (ANY_TYPE)
args	Type: .System.Int32 . The ANY_TYPE arguments.
period	Type: System.TimeSpan The period.

Return Value

Type: [IObservable.Object](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class](#) [► 1281]

[PollValues Overload \[► 1291\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.2.1.1.25 AnyTypeExtensions.PollValues Method (IAdsConnection, ISymbol, Type, IObservable.Unit., Func.Exception, Object.)

Polls the symbol values on timepoints where the polling observable streams data / triggers

Namespace: [TwinCAT.Ads.Reactive \[► 1249\]](#)

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object> PollValues(
    this IAdsConnection connection,
    ISymbol symbol,
    Type type,
    IObservable<Unit> trigger,
    Func<Exception, Object>? errorHandler
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol.
type	Type: System.Type The data type of the symbol (ANY_TYPE)
trigger	Type: System.IObservable.Unit. The Polling trigger
errorHandler	Type: System.Func.Exception, Object. The error handler.

Return Value

Type: [IObservable.Object.](#)
IObservable<System.Object>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection \[► 876\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class \[► 1281\]](#)

[PollValues Overload \[► 1291\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.2.1.1.26 AnyTypeExtensions.PollValues Method (IAdsConnection, ISymbol, Type, TimeSpan, Func.Exception, Object.)

Polls the symbol as value sequence of object values with a specified period time.

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object> PollValues(
    this IAdsConnection connection,
    ISymbol symbol,
    Type type,
    TimeSpan period,
    Func<Exception, Object>? errorHandler
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol.
type	Type: System.Type The data type of the symbol (ANY_TYPE)
period	Type: System.TimeSpan The period.
errorHandler	Type: System.Func.Exception , Object . The error handler.

Return Value

Type: [IObservable.Object](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class](#) [► 1281]

[PollValues Overload](#) [► 1291]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.2.1.1.27 AnyTypeExtensions.PollValues.T. Method (IAdsConnection, ISymbol, .Int32., IObservable.Unit., Func.Exception, T.)

Polls the symbol values on timepoints where the polling observable streams data / triggers

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> PollValues<T>(
    this IAdsConnection connection,
    ISymbol symbol,
```

```
int[]? args,
IObservable<Unit> trigger,
Func<Exception, T> errorHandler
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The Symbol.
args	Type: .System.Int32 . ANY_TYPE arguments
trigger	Type: System.IObservable.Unit . The Polling trigger
errorHandler	Type: System.Func.Exception, T . The error handler.

Type Parameters

T	The ANY_TYPE compatible .NET Type.
---	------------------------------------

Return Value

Type: [IObservable.T](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class](#) [► 1281]

[PollValues Overload](#) [► 1291]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.2.1.1.28 AnyTypeExtensions.PollValues.T. Method (IAdsConnection, ISymbol, .Int32., TimeSpan, Func.Exception, T.)

Polls the symbol as value sequence of object values with a specified period time.

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> PollValues<T>(
    this IAdsConnection connection,
    ISymbol symbol,
    int[]? args,
    TimeSpan period,
    Func<Exception, T> errorHandler
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The instance path.
args	Type: System.Int32 . ANY_TYPE arguments.
period	Type: System.TimeSpan The period.
errorHandler	Type: System.Func.Exception, T . The error handler.

Type Parameters

T	The ANY_TYPE compatible .NET Type.
---	------------------------------------

Return Value

Type: [IObservable.T](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

Polling ANY_TYPE values.

Observe changing ADS Symbols by polling (Read Polling) (ANY_TYPE)

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo.CycleCount"];

    // Reactive Notification Handler
    var valueObserver = Observer.Create<object>(val =>
    {
        Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToString()));
    });

    // Take 20 Values in an Interval of 500ms
    IDisposable subscription = cycleCount.PollValues(TimeSpan.FromMilliseconds(500)).Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[AnyTypeExtensions Class](#) [► 1281]

[PollValues Overload](#) [► 1291]

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.2.1.1.29 AnyTypeExtensions.PollValues Method (IAdsConnection, ISymbol, Type, .Int32., IObservable.Unit., Func.Exception, Object.)

Polls the symbol values on time points where the polling observable streams data / triggers

Namespace: [TwinCAT.Ads.Reactive \[► 1249\]](#)

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object> PollValues(
    this IAdsConnection connection,
    ISymbol symbol,
    Type type,
    int[]? args,
    IObservable<Unit> trigger,
    Func<Exception, Object>? errorHandler
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol.
type	Type: System.Type The data type of the symbol (ANY_TYPE)
args	Type: .System.Int32. The ANY_TYPE arguments.
trigger	Type: System.IObservable.Unit. The Polling trigger
errorHandler	Type: System.Func.Exception, Object. The error handler.

Return Value

Type: [IObservable.Object.](#)
[IObservable<System.Object>.](#)

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection \[► 876\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class \[► 1281\]](#)

[PollValues Overload \[► 1291\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.2.1.1.30 AnyTypeExtensions.PollValues Method (IAdsConnection, ISymbol, Type, .Int32., TimeSpan, Func.Exception, Object.)

Polls the symbol as value sequence of object values with a specified period time.

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object> PollValues(
    this IAdsConnection connection,
    ISymbol symbol,
    Type type,
    int[]? args,
    TimeSpan period,
    Func<Exception, Object>? errorHandler
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol.
type	Type: System.Type The data type of the symbol (ANY_TYPE)
args	Type: .System.Int32 . The ANY_TYPE arguments.
period	Type: System.TimeSpan The period.
errorHandler	Type: System.Func.Exception , Object . The error handler.

Return Value

Type: [IObservable.Object](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference



[AnyTypeExtensions Class](#) [[▶ 1281](#)]

[PollValues Overload](#) [[▶ 1291](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

6.3.2.1.2 AnyTypeExtensions.WhenNotification Method

Overload List

	Name	Description
	WhenNotification.T . (IAdsConnection , String , NotificationSettings) [▶ 1323]	Creates an observable sequence of values that are created by ADS Notifications.
	WhenNotification (IAdsConnection , String , Type , NotificationSettings) [▶ 1324]	Creates an observable sequence of values that are created by ADS Notifications.

Reference

[AnyTypeExtensions Class](#) [[▶ 1281](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

6.3.2.1.2.1 AnyTypeExtensions.WhenNotification.T Method (IAdsConnection, String, NotificationSettings)

Creates an observable sequence of values that are created by ADS Notifications.

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> WhenNotification<T>(
    this IAdsConnection connection,
    string instancePath,
    NotificationSettings settings
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
instancePath	Type: System.String The instance path.
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The settings.

Type Parameters

T	The .NET Type representation of the specified symbols type.
---	---

Return Value

Type: [IObservable.T](#).
[IObservable<T>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

The values will be cast to the specified type. The .NET type must fit the Symbol type like all ANYTYPES.

Examples

The following sample shows how to observe Value changed Notifications with the reactive [AnyTypeExtensions](#) [► 1281]

Observe changing ADS Symbols with reactive extensions (Extended AdsNotification, ANY_TYPE)

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Reactive Notification Handler
    var valueObserver = Observer.Create<ushort>(val =>
    {
        Console.WriteLine(string.Format("Value: {0}", val.ToString()));
    }
    );

    // Turning ADS Notifications into sequences of Value Objects (Taking 20 Values)
    // and subscribe to them.
    IDisposable subscription = client.WhenNotification<ushort>("TwinCAT_SystemInfoVarList._TaskInfo.CycleCount", NotificationSettings.Default).Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[AnyTypeExtensions Class](#) [► 1281]

[WhenNotification Overload](#) [► 1323]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.2.1.2.2 AnyTypeExtensions.WhenNotification Method (IAdsConnection, String, Type, NotificationSettings)

Creates an observable sequence of values that are created by ADS Notifications.

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object> WhenNotification(
    this IAdsConnection connection,
    string instancePath,
    Type type,
    NotificationSettings settings
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
instancePath	Type: System.String The instance path.
type	Type: System.Type The type.
settings	Type: TwinCAT.Ads.NotificationSettings [► 1106] The settings.

Return Value

Type: [IObservable.Object](#).
[IObservable<T>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

The values will be cast to the specified type. The .NET type must fit be one of the compatible 'ANYTYPES'.

Examples

The following sample shows how to observe Value changed Notifications with the reactive [AnyTypeExtensions](#) [► 1281]

Observe changing ADS Symbols with reactive extensions (Extended AdsNotifications, ANY_TYPE)

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Reactive Notification Handler
    var valueObserver = Observer.Create<ushort>(val =>
    {
        Console.WriteLine(string.Format("Value: {0}", val.ToString()));
    }
    );

    // Turning ADS Notifications into sequences of Value Objects (Taking 20 Values)
    // and subscribe to them.
    IDisposable subscription = client.WhenNotification<ushort>("TwinCAT_SystemInfoVarList._TaskInfo.CycleCount", NotificationSettings.Default).Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference







[AnyTypeExtensions Class](#) [► 1281]

[WhenNotification Overload](#) [► 1323]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.2.1.3 AnyTypeExtensions.WriteValues Method

Overload List

	Name	Description
  	WriteValues.T. (IAdsConnection , String , IObservable.T.) [▶ 1326]	Writes the sequence of values to the symbol specified by the instance path.
  	WriteValues.T. (IAdsConnection , String , IObservable.T. , Action.Exception.) [▶ 1327]	Writes the sequence of values to the symbol specified by the instance path.

Reference

[AnyTypeExtensions Class](#) [▶ 1281]

[TwinCAT.Ads.Reactive Namespace](#) [▶ 1249]

6.3.2.1.3.1 AnyTypeExtensions.WriteValues.T. Method (IAdsConnection, String, IObservable.T.)

Writes the sequence of values to the symbol specified by the instance path.

Namespace: [TwinCAT.Ads.Reactive](#) [▶ 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IDisposable WriteValues<T>(
    this IAdsConnection connection,
    string instancePath,
    IObservable<T> valueSequence
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
instancePath	Type: System.String The instance path.
valueSequence	Type: System.IObservable.T. Value sequence (Any type).

Type Parameters

T

Return Value

Type: [IDisposable](#)
IDisposable.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

Write values sequentially.

Write sequences of values to the target (ANY_TYPE)

```
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information (Symbol 'i : INT' in PLC Global Variables list.
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol gvlIntSymbol = (IValueSymbol)symbolLoader.Symbols["GVL.i"];

    // Produces object (short) Values 0,1,2,3 ... in seconds period
    IObservable<object> timerObservable = Observable.Interval(TimeSpan.FromSeconds(1.0)).Select(i =>
(object) (short)i);

    // Take 10 Values (0..9) and write them to GVL.i
    IDisposable dispose = gvlIntSymbol.WriteValues(timerObservable.Take(10));

    Console.ReadKey(); // Wait for Key press
    dispose.Dispose(); // Dispose the Subscription
}
```

Reference

[AnyTypeExtensions Class](#) [► 1281]

[WriteValues Overload](#) [► 1326]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.2.1.3.2 AnyTypeExtensions.WriteValues.T. Method (IAdsConnection, String, IObservable.T., Action.Exception.)

Writes the sequence of values to the symbol specified by the instance path.

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IDisposable WriteValues<T>(
    this IAdsConnection connection,
    string instancePath,
    IObservable<T> valueSequence,
    Action<Exception>? errorHandler
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
instancePath	Type: System.String The instance path.
valueSequence	Type: System.IObservable.T. Value sequence (Any type).

errorHandler	Type: System.Action.Exception . The error handler.
--------------	---

Type Parameters

T

Return Value

Type: [IDisposable](#)
IDisposable.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

Write values sequentially.

Write sequences of values to the target (ANY_TYPE)

```
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information (Symbol 'i : INT' in PLC Global Variables list.
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol gvlIntSymbol = (IValueSymbol)symbolLoader.Symbols["GVL.i"];

    // Produces object (short) Values 0,1,2,3 ... in seconds period
    IObservable<object> timerObservable = Observable.Interval(TimeSpan.FromSeconds(1.0)).Select(i =>
(object)(short)i);

    // Take 10 Values (0..9) and write them to GVL.i
    IDisposable dispose = gvlIntSymbol.WriteValues(timerObservable.Take(10));

    Console.ReadKey(); // Wait for Key press
    dispose.Dispose(); // Dispose the Subscription
}
```

Reference

[AnyTypeExtensions Class](#) [[▶ 1281](#)]

[WriteValues Overload](#) [[▶ 1326](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

6.3.2.1.4 AnyTypeExtensions.PollValues2 Method

Overload List

	Name	Description
	PollValues2.T. (IAdsConnection , String , .Int32 , IObservable.Unit) ▶ 1329	Polls the symbol values on time points where the polling observable streams data / triggers

	Name	Description
	PollValues2.T. (IAdsConnection. , ISymbol. .Int32. , IObservable.Unit.) [▶ 1330]	Polls the symbol values on timepoints where the polling observable streams data / triggers

Reference

[AnyTypeExtensions Class](#) [▶ [1281](#)]

[TwinCAT.Ads.Reactive Namespace](#) [▶ [1249](#)]

6.3.2.1.4.1 AnyTypeExtensions.PollValues2.T. Method (IAdsConnection, String, .Int32., IObservable.Unit.)

Polls the symbol values on time points where the polling observable streams data / triggers

Namespace: [TwinCAT.Ads.Reactive](#) [▶ [1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultReadValueAccess<T>> PollValues2<T>(
    this IAdsConnection connection,
    string instancePath,
    int[]? args,
    IObservable<Unit> trigger
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
instancePath	Type: System.String The instance path.
args	Type: .System.Int32. The ANY_TYPE arguments.
trigger	Type: System.IObservable.Unit. The Polling trigger

Type Parameters

T

Return Value

Type: [IObservable.ResultReadValueAccess](#) [▶ [3211](#)].T..
IObservable<System.Object>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [▶ [876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Exceptions

Exception	Condition
ArgumentNullException	connection

Reference

[AnyTypeExtensions Class](#) [► 1281]

[PollValues2 Overload](#) [► 1328]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.2.1.4.2 AnyTypeExtensions.PollValues2.T. Method (IAdsConnection, ISymbol, .Int32., IObservable.Unit.)

Polls the symbol values on timepoints where the polling observable streams data / triggers

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultReadValueAccess2<ISymbol, T>> PollValues2<T>(
    this IAdsConnection connection,
    ISymbol symbol,
    int[]? args,
    IObservable<Unit> trigger
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol.
args	Type: .System.Int32 . The arguments.
trigger	Type: System.IObservable.Unit . The Polling trigger

Type Parameters

T	The ANY_TYPE compatible .NET Type.
---	------------------------------------

Return Value

Type: [IObservable.ResultReadValueAccess2](#) [► 3226].[ISymbol](#) [► 2691], T..
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[AnyTypeExtensions Class](#) [► 1281]

[PollValues2 Overload \[► 1328\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.3 SymbolValueNotification Class

Symbol Notification class

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.Notification \[► 1100\]](#)

[ValueNotification](#)

[TwinCAT.Ads.Reactive.SymbolValueNotification](#)

Namespace: [TwinCAT.Ads.Reactive \[► 1249\]](#)

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229






Syntax

C#





```
public sealed class SymbolValueNotification : ValueNotification
```

The SymbolValueNotification type exposes the following members.

Properties

	Name	Description
	Data [► 1104]	The notification Data. (Inherited from Notification [► 1100] .)
	Handle [► 1105]	The notification handle (Inherited from Notification [► 1100] .)
	Symbol [► 1332]	Gets the symbol of the SymbolValueNotification.
	TimeStamp [► 1105]	Gets the time stamp of the INotification [► 1096] (Inherited from Notification [► 1100] .)
	UserData [► 1105]	Attached UserData/Tag at the INotification [► 1096] (Inherited from Notification [► 1100] .)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Remarks

Extends the ValueNotification class by symbol specific information.






Reference

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.3.1 SymbolValueNotification Properties

The [SymbolValueNotification](#) [▶ 1331] type exposes the following members.

Properties

	Name	Description
	Data [▶ 1104]	The notification Data. (Inherited from Notification [▶ 1100].)
	Handle [▶ 1105]	The notification handle (Inherited from Notification [▶ 1100].)
	Symbol [▶ 1332]	Gets the symbol of the SymbolValueNotification [▶ 1331].
	TimeStamp [▶ 1105]	Gets the time stamp of the INotification [▶ 1096] (Inherited from Notification [▶ 1100].)
	UserData [▶ 1105]	Attached UserData/Tag at the INotification [▶ 1096] (Inherited from Notification [▶ 1100].)

Reference

[SymbolValueNotification Class](#) [▶ 1331]

[TwinCAT.Ads.Reactive Namespace](#) [▶ 1249]

6.3.3.1.1 SymbolValueNotification.Symbol Property

Gets the symbol of the [SymbolValueNotification](#) [▶ 1331].

Namespace: [TwinCAT.Ads.Reactive](#) [▶ 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISymbol Symbol { get; }
```

Property Value

Type: [ISymbol](#) [▶ 2691]

The value symbol.

Reference



[SymbolValueNotification Class](#) [▶ 1331]



[TwinCAT.Ads.Reactive Namespace](#) [▶ 1249]

6.3.3.2 SymbolValueNotification Methods

The [SymbolValueNotification](#) [▶ 1331] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)

	Name	Description
	GetType	Gets the <u>Type</u> of the current instance. (Inherited from <u>Object</u> .)
	ToString	Returns a string that represents the current object. (Inherited from <u>Object</u> .)

Reference

[SymbolValueNotification Class](#) [▶ 1331]

[TwinCAT.Ads.Reactive Namespace](#) [▶ 1249]

6.3.4 ValueSymbolExtensions Class

Extension class for [IAdsConnection](#) [▶ 876] to provide reactive ADS extensions for accessing symbols that are loaded by the [IAdsSymbolLoaderFactory](#)

Inheritance Hierarchy

[System.Object](#)

 TwinCAT.Ads.Reactive.ValueSymbolExtensions

Namespace: [TwinCAT.Ads.Reactive](#) [▶ 1249]














Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax














C#

```
public static class ValueSymbolExtensions
```

Methods

	Name	Description
  	PollValues(IAdsRead, IObservable.Unit.) [▶ 1359]	Polls a series of symbols via a IAdsRead [▶ 1596] command. The SumCommand will read all contained values with every trigger.
  	PollValues(IAdsRead, TimeSpan) [▶ 1361]	Polls a series of symbols via a IAdsRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger in one roundtrip.
 	PollValues(IValueSymbol, IObservable.Unit.) [▶ 1355]	Poll symbol values on trigger signals.
  	PollValues(IValueSymbol, TimeSpan) [▶ 1356]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues(IValueSymbol, IObservable.Unit., Boolean) [▶ 1357]	Polls symbol values on trigger signals.

	Name	Description
  	PollValues(IValueSymbol, TimeSpan, Boolean) [▶ 1358]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T.(IValueSymbol, IObservable.Unit.) [▶ 1362]	Poll symbol values as a value sequence on trigger signals (typed)
  	PollValues.T.(IValueSymbol, TimeSpan) [▶ 1363]	Polls the symbol as value sequence of object values with a specified period time (typed)
 	PollValues.T.(IValueSymbol, IObservable.Unit., Boolean) [▶ 1364]	Poll symbol values on trigger signals (typed)
  	PollValues.T.(IValueSymbol, IObservable.Unit., Func.ResultReadValueAccess2.IValueSymbol, Object., T.) [▶ 1365]	Poll symbol values on trigger signals (typed)
 	PollValues.T.(IValueSymbol, TimeSpan, Boolean) [▶ 1367]	Polls the symbol as value sequence of object values with a specified period time (typed)
 	PollValues2(IValueSymbol, IObservable.Unit.) [▶ 1368]	Poll symbol values as a sequence of annotated results (Value + ErrorCode)
 	PollValues2(IValueSymbol, TimeSpan) [▶ 1369]	Poll symbol values with communication return codes.
  	PollValues2.S.(ISumRead2.S., IObservable.Unit.) [▶ 1370]	Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger in one roundtrip.
  	PollValues2.S.(ISumRead2.S., TimeSpan) [▶ 1372]	Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger.
 	WhenValueChanged(IValueSymbol) [▶ 1344]	Gets an observable sequence when the value of the IValueSymbol [▶ 2775] has changed.

	Name	Description
		
  	WhenValueChanged (IAdsConnection , IEnumerable.ISymbol .) [▶ 1345]	Observable sequence of Values driven by ADS Notifications on the specified symbol.
  	WriteValues(IValueSymbol, IObservable.Object.) [▶ 1347]	Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775].
 	WriteValues(IValueSymbol, IObservable.Object., Action.Exception.) [▶ 1348]	Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775].
 	WriteValues(IValueSymbol, IObservable.Object., Cancellation.Token) [▶ 1349]	Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775].
 	WriteValues(IValueSymbol, IObservable.Object., Action.Exception., Cancellation.Token) [▶ 1350]	Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775].

Remarks

Reactive Extensions (Rx) are a library for composing asynchronous and event-based programs using observable sequences and LINQ-style query operators. Using Rx, developers represent asynchronous data streams with Observables, query asynchronous data streams using LINQ operators, and parameterize the concurrency in the asynchronous data streams using Schedulers. Simply put, Rx = Observables + LINQ + Schedulers. The ADS reactive extensions are build on top of this library to enable ADS Symbol and State Observables, seamlessly bound to the reactive extensions. To use the ADS reactive extensions the TwinCAT.Ads.Reactive Nuget package (or the included TwinCAT.Ads.Reactive.dll) must be referenced from All types within are contained in the ADS companion package "Beckhoff.TwinCAT.Ads.Reactive" which must be referenced separately. ([Beckhoff.TwinCAT.Ads.Reactive package on Nuget](#)).

Examples

The following sample shows how to observe Value changed Notifications with the reactive ValueSymbolExtensions from an [IValueSymbol](#) [[▶ 2775](#)].

Observe a single changing ADS Symbol (ADS Notifications)

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
```

```

var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];

// Reactive Notification Handler
var valueObserver = Observer.Create<object>(val =>
{
    Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToString()));
}
);

cycleCount.NotificationSettings = new NotificationSettings(AdsTransMode.OnChange, 500, 5000); // optional: Change NotificationSettings on Symbol

// Turning ADS Notifications into sequences of Value Objects (Taking 20 Values)
// and subscribe to them.
IDisposable subscription = cycleCount.WhenValueChanged().Take(20).Subscribe(valueObserver);

Console.ReadKey(); // Wait for Key press
subscription.Dispose(); // Dispose the Subscription
}

```

Examples

The following sample shows how to observe Value changed Notifications with the reactive ValueSymbolExtensions from an **DynamicSymbol**.

Observe a single changing ADS Symbol (ADS Notifications) with the dynamic language runtime (.NET DLR)

```

// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];

    // Reactive Notification Handler
    var valueObserver = Observer.Create<object>(val =>
    {
        Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToString()));
    }
    );

    cycleCount.NotificationSettings = new NotificationSettings(AdsTransMode.OnChange, 500, 5000); // optional: Change NotificationSettings on Symbol

    // Turning ADS Notifications into sequences of Value Objects (Taking 20 Values)
    // and subscribe to them.
    IDisposable subscription = cycleCount.WhenValueChanged().Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}

```

Examples

The same for more than one [IValueSymbol](#) [[▶ 2775](#)].

Observe changing ADS Symbols (ADS Notifications)

```

// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo

```



```

o.CycleCount"]; // UShort Type
    IValueSymbol lastExecTime = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskI
nfo.LastExecTime"]; // UInt Type

    SymbolCollection symbols = new SymbolCollection();
    symbols.Add(cycleCount);
    symbols.Add(lastExecTime);

    // Reactive Notification Handler
    var valueObserver = Observer.Create<object>(val =>
    {
        Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToStri
ng()));
    }
    );

    cycleCount.NotificationSettings = new NotificationSettings(AdsTransMode.OnChange,500,5000); // o
ptional: Change NotificationSettings on Symbol

    // Turning ADS Notifications into sequences of Value Objects (Taking 20 Values)
    // and subscribe to them.
    IDisposable subscription = client.WhenValueChanged(symbols).Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}

```

Examples

Here, the values are polled in a specific time period and sequential Reads are triggered (in opposite to ADS Notification in the latter example)

Observe changing ADS Symbols by polling (Read Polling)

```

// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInf
o[1].CycleCount"];

    // Reactive Notification Handler
    var valueObserver = Observer.Create<object>(val =>
    {
        Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToStri
ng()));
    }
    );

    // Take 20 Values in an Interval of 500ms
    IDisposable subscription = cycleCount.PollValues(TimeSpan.FromMilliseconds(500)).Take(20).Subscr
ibe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}

```

Examples

In the following example it is demonstrated how to write Values sequentially to a [IValueSymbol](#) [▶ 2775](#) with the help of the reactive extensions.

Write sequences of values to the target

```

using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information (Symbol 'i : INT' in PLC Global Variables list.
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["GVL.i"];
}

```

```

// Produces object Values 0,1,2,3 ... in seconds period
IObservable<object> timerObservable = Observable.Interval(TimeSpan.FromSeconds(1.0)).Select(i =>
(object) (short)i);

// Take 10 Values (0..9) and write them to GVL.i
IDisposable dispose = cycleCount.WriteValues(timerObservable.Take(10));

Console.ReadKey(); // Wait for Key press
dispose.Dispose(); // Dispose the Subscription
}

```

Examples

Polling of values as SumCommands (reading of multiple symbols/values in one ADS roundtrip)

SumCommand polling (Symbols)

```

// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];
    IValueSymbol lastExecTime = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo.LastExecTime"];
    List<ISymbol> symbols = new List<ISymbol>() { cycleCount, lastExecTime };

    //Create the SumCommand
    SumSymbolRead sumRead = new SumSymbolRead(client, symbols);

    // Reactive Notification Handler
    var sumCommandObserver = Observer.Create<ResultSumValues2<ISymbol>>(result =>
    {
        Console.WriteLine($"SumCommand ErrorCode: {result.ErrorCode}");

        if (result.Succeeded)
        {
            Console.WriteLine($"SumCommand OverallSucceeded: {result.OverallSucceeded}");
            foreach (var subResult in result.ValueResults)
            {
                Console.WriteLine($"Source: {subResult.Source}, ErrorCode: {subResult.ErrorCode}, Value: {subResult.Value}");
            }
        }
    });

    // Take 20 Values/Result in an Interval of 500ms
    IDisposable subscription = sumRead.PollValues2(TimeSpan.FromMilliseconds(500)).Take(20).Subscribe(sumCommandObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}

```

SumCommand polling (via InstancePath)

```

// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];
    IValueSymbol lastExecTime = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo.LastExecTime"];
    List<ISymbol> symbols = new List<ISymbol>() { cycleCount, lastExecTime };

    //Create the SumCommand
    SumSymbolRead sumRead = new SumSymbolRead(client, symbols);
}

```

```
// Reactive Notification Handler
var sumCommandObserver = Observer.Create<ResultSumValues>(result =>
{
    Console.WriteLine($"SumCommand Succeeded: {result.OverallSucceeded}, CycleCount: {result.Values[0]}, LastExecTime: {result.Values[1]}");
})
);

// Take 20 Values/Result in an Interval of 500ms
IDisposable subscription = sumRead.PollValues(TimeSpan.FromMilliseconds(500)).Take(20).Subscribe(sumCommandObserver);

Console.ReadKey(); // Wait for Key press
subscription.Dispose(); // Dispose the Subscription
}
```




Reference

[TwinCAT.Ads.Reactive Namespace \[▶ 1249\]](#)

[TwinCAT.Ads.Reactive.AdsClientExtensions \[▶ 1249\]](#)














[TwinCAT.Ads.Reactive.AnyTypeExtensions \[▶ 1281\]](#)

Also see about this

-  [ValueSymbolExtensions.PollValues Method \(ISumSymbolRead, IObservable.Unit.\) \[▶ 1352\]](#)
-  [ISumSymbolRead Interface \[▶ 1587\]](#)
-  [ValueSymbolExtensions.PollValues Method \(ISumSymbolRead, TimeSpan\) \[▶ 1353\]](#)

6.3.4.1 ValueSymbolExtensions Methods

Methods

	Name	Description
  	PollValues(ISumRead, IObservable.Unit.) [▶ 1359]	Polls a series of symbols via a ISumRead [▶ 1596] command. The SumCommand will read all contained values with every trigger.
  	PollValues(ISumRead, TimeSpan) [▶ 1361]	Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger in one roundtrip.
 	PollValues(IValueSymbol, IObservable.Unit.) [▶ 1355]	Poll symbol values on trigger signals.
  	PollValues(IValueSymbol, TimeSpan) [▶ 1356]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues(IValueSymbol, IObservable.Unit, Boolean) [▶ 1357]	Polls symbol values on trigger signals.

	Name	Description
  	PollValues(IValueSymbol, TimeSpan, Boolean) [▶ 1358]	Polls the symbol as value sequence of object values with a specified period time.
 	PollValues.T.(IValueSymbol, IObservable.Unit.) [▶ 1362]	Poll symbol values as a value sequence on trigger signals (typed)
  	PollValues.T.(IValueSymbol, TimeSpan) [▶ 1363]	Polls the symbol as value sequence of object values with a specified period time (typed)
 	PollValues.T.(IValueSymbol, IObservable.Unit., Boolean) [▶ 1364]	Poll symbol values on trigger signals (typed)
  	PollValues.T.(IValueSymbol, IObservable.Unit., Func.ResultReadValueAccess2.IValueSymbol, Object., T.) [▶ 1365]	Poll symbol values on trigger signals (typed)
 	PollValues.T.(IValueSymbol, TimeSpan, Boolean) [▶ 1367]	Polls the symbol as value sequence of object values with a specified period time (typed)
 	PollValues2(IValueSymbol, IObservable.Unit.) [▶ 1368]	Poll symbol values as a sequence of annotated results (Value + ErrorCode)
 	PollValues2(IValueSymbol, TimeSpan) [▶ 1369]	Poll symbol values with communication return codes.
  	PollValues2.S.(ISumRead2.S., IObservable.Unit.) [▶ 1370]	Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger in one roundtrip.
  	PollValues2.S.(ISumRead2.S., TimeSpan) [▶ 1372]	Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger.
 	WhenValueChanged(IValueSymbol) [▶ 1344]	Gets an observable sequence when the value of the IValueSymbol [▶ 2775] has changed.

	Name	Description
 	WhenValueChanged (IAdsConnection , IEnumerable.ISymbol .) [▶ 1345]	Observable sequence of Values driven by ADS Notifications on the specified symbol.
 	WriteValues(IValueSymbol, IObservable.Object.) [▶ 1347]	Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775].
 	WriteValues(IValueSymbol, IObservable.Object., Action.Exception.) [▶ 1348]	Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775].
 	WriteValues(IValueSymbol, IObservable.Object., CancellationToken) [▶ 1349]	Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775].
 	WriteValues(IValueSymbol, IObservable.Object., Action.Exception., CancellationToken) [▶ 1350]	Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775].

Reference

[ValueSymbolExtensions Class](#) [[▶ 1333](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]


Also see about this

- [ValueSymbolExtensions.PollValues Method \(ISumSymbolRead, IObservable.Unit.\)](#) [[▶ 1352](#)]
- [ISumSymbolRead Interface](#) [[▶ 1587](#)]
- [ValueSymbolExtensions.PollValues Method \(ISumSymbolRead, TimeSpan\)](#) [[▶ 1353](#)]

6.3.4.1.1 ValueSymbolExtensions.PollValuesAnnotated Method

Overload List

	Name	Description
 	PollValuesAnnotated(IValueSymbol, IObservable.Unit.) [▶ 1342]	Polls the values as ValueChangedEventArgs [▶ 2971] sequence annotated value on trigger sequence

	Name	Description
	PollValuesAnnotated(IValueSymbol, TimeSpan) [► 1342]	Polls the values as ValueChangedEventArgs [► 2971] sequence with a specified period time.

Reference

[ValueSymbolExtensions Class](#) [► 1333]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.4.1.1.1 ValueSymbolExtensions.PollValuesAnnotated Method (IValueSymbol, IObservable.Unit.)

Polls the values as [ValueChangedEventArgs](#) [► 2971] sequence annotated value on trigger sequence

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934 addedca3e72bc0ea15da1c14

Syntax

C#

```
public static IObservable<ValueChangedEventArgs> PollValuesAnnotated(
    this IValueSymbol symbol,
    IObservable<Unit> trigger
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IValueSymbol [► 2775] The symbol.
trigger	Type: System.IObservable.Unit . The polling Trigger.

Return Value

Type: [IObservable.ValueChangedEventArgs](#) [► 2971].
[IObservable<ValueChangedArgs>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IValueSymbol](#) [► 2775]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[ValueSymbolExtensions Class](#) [► 1333]

[PollValuesAnnotated Overload](#) [► 1341]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.4.1.1.2 ValueSymbolExtensions.PollValuesAnnotated Method (IValueSymbol, TimeSpan)

Polls the values as [ValueChangedEventArgs](#) [► 2971] sequence with a specified period time.

Namespace: [TwinCAT.Ads.Reactive](#) [▶ 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public static IObservable<ValueChangedEventArgs> PollValuesAnnotated(
    this IValueSymbol symbol,
    TimeSpan period
)
```

Parameters

- symbol Type: [TwinCAT.TypeSystem.IValueSymbol](#) [▶ 2775]
The symbol.
- period Type: [System.TimeSpan](#)
The polling period/interval.

Return Value

Type: [IObservable.ValueChangedEventArgs](#) [▶ 2971].
IObservable<ValueChangedArgs>.

Usage Note



In Visual Basic and C#, you can call this method as an instance method on any object of type [IValueSymbol](#) [▶ 2775]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

- [ValueSymbolExtensions Class](#) [▶ 1333]
- [PollValuesAnnotated Overload](#) [▶ 1341]
- [TwinCAT.Ads.Reactive Namespace](#) [▶ 1249]

6.3.4.1.2 ValueSymbolExtensions.WhenValueChanged Method

Overload List

	Name	Description
	WhenValueChanged (IValueSymbol) [▶ 1344]	Gets an observable sequence when the value of the IValueSymbol [▶ 2775] has changed.
	WhenValueChanged (IAdsConnection, IEnumerable.ISymbol.) [▶ 1345]	Observable sequence of Values driven by ADS Notifications on the specified symbol.

Reference

- [ValueSymbolExtensions Class](#) [▶ 1333]
- [TwinCAT.Ads.Reactive Namespace](#) [▶ 1249]

6.3.4.1.2.1 ValueSymbolExtensions.WhenValueChanged Method (IValueSymbol)

Gets an observable sequence when the value of the [IValueSymbol](#) [▸ 2775] has changed.

Namespace: [TwinCAT.Ads.Reactive](#) [▸ 1249]

Assembly: [TwinCAT.Ads.Reactive](#) (in [TwinCAT.Ads.Reactive.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object> WhenValueChanged(
    this IValueSymbol symbol
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IValueSymbol [▸ 2775] The symbol.
--------	---

Return Value

Type: [IObservable.Object](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IValueSymbol](#) [▸ 2775]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

The following sample shows how to observe Value changed Notifications with the reactive [ValueSymbolExtensions](#) [▸ 1333] from an [IValueSymbol](#) [▸ 2775].

Observe a single changing ADS Symbols (ADS Notifications)

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];

    // Reactive Notification Handler
    var valueObserver = Observer.Create<object>(val =>
    {
        Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToString()));
    });

    cycleCount.NotificationSettings = new NotificationSettings(AdsTransMode.OnChange, 500, 5000); // optional: Change NotificationSettings on Symbol

    // Turning ADS Notifications into sequences of Value Objects (Taking 20 Values)
    // and subscribe to them.
    IDisposable subscription = cycleCount.WhenValueChanged().Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```


Reference

[ValueSymbolExtensions Class](#) [► 1333]

[WhenValueChanged Overload](#) [► 1343]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

6.3.4.1.2.2 ValueSymbolExtensions.WhenValueChanged Method (IAdsConnection, IEnumerable.ISymbol.)

Observable sequence of Values driven by ADS Notifications on the specified symbol.

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object> WhenValueChanged(
    this IAdsConnection connection,
    IEnumerable<ISymbol> symbols
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The ADS connection / ADS Client
symbols	Type: System.Collections.Generic.IEnumerable.ISymbol [► 2691]. The symbols to observe.

Return Value

Type: [IObservable.Object](#).
[IObservable<ValueChangedArgs>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

The same for more than one [IValueSymbol](#) [► 2775].

Observe changing ADS Symbols (ADS Notifications)

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo.CycleCount"]; // UShort Type
    IValueSymbol lastExecTime = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo.LastExecTime"]; // UInt Type

    SymbolCollection symbols = new SymbolCollection();
    symbols.Add(cycleCount);
    symbols.Add(lastExecTime);
}
```

```
// Reactive Notification Handler
var valueObserver = Observer.Create<object>(val =>
{
    Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToString()));
}
);

cycleCount.NotificationSettings = new NotificationSettings(AdsTransMode.OnChange,500,5000); // optional: Change NotificationSettings on Symbol

// Turning ADS Notifications into sequences of Value Objects (Taking 20 Values)
// and subscribe to them.
IDisposable subscription = client.WhenValueChanged(symbols).Take(20).Subscribe(valueObserver);

Console.ReadKey(); // Wait for Key press
subscription.Dispose(); // Dispose the Subscription
}
```

Examples

The following sample shows how to observe Value changed Notifications with the reactive [ValueSymbolExtensions \[► 1333\]](#) from an **DynamicSymbol**.

Observe a single changing ADS Symbol (ADS Notifications) with the dynamic language runtime (.NET DLR)

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];

    // Reactive Notification Handler
    var valueObserver = Observer.Create<object>(val =>
    {
        Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToString()));
    }
    );

    cycleCount.NotificationSettings = new NotificationSettings(AdsTransMode.OnChange, 500, 5000); // optional: Change NotificationSettings on Symbol

    // Turning ADS Notifications into sequences of Value Objects (Taking 20 Values)
    // and subscribe to them.
    IDisposable subscription = cycleCount.WhenValueChanged().Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[ValueSymbolExtensions Class \[► 1333\]](#)









[WhenValueChanged Overload \[► 1343\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

Observable

6.3.4.1.3 ValueSymbolExtensions.WriteValues Method

Overload List

	Name	Description
  	WriteValues(IValueSymbol, IObservable.Object.) [1347]	Subscribes the IValueSymbol [2775] to an observable sequence of values and writes them to the IValueSymbol [2775].
 	WriteValues(IValueSymbol, IObservable.Object., Action.Exception.) [1348]	Subscribes the IValueSymbol [2775] to an observable sequence of values and writes them to the IValueSymbol [2775].
 	WriteValues(IValueSymbol, IObservable.Object., CancellationToken) [1349]	Subscribes the IValueSymbol [2775] to an observable sequence of values and writes them to the IValueSymbol [2775].
 	WriteValues(IValueSymbol, IObservable.Object., Action.Exception., CancellationToken) [1350]	Subscribes the IValueSymbol [2775] to an observable sequence of values and writes them to the IValueSymbol [2775].

Reference

[ValueSymbolExtensions Class](#) [[1333](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[1249](#)]

6.3.4.1.3.1 ValueSymbolExtensions.WriteValues Method (IValueSymbol, IObservable.Object.)

Subscribes the [IValueSymbol](#) [[2775](#)] to an observable sequence of values and writes them to the [IValueSymbol](#) [[2775](#)].

Namespace: [TwinCAT.Ads.Reactive](#) [[1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IDisposable WriteValues(
    this IValueSymbol symbol,
    IObservable<Object> valueObservable
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IValueSymbol [2775] The symbol.
--------	---

valueObservable	Type: System.IObservable.Object . Observable of Values.
-----------------	--

Return Value

Type: [IDisposable](#)
IDisposable.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IValueSymbol](#) [[▶ 2775](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

In the following example it is demonstrated how to write Values sequentially to a [IValueSymbol](#) [[▶ 2775](#)] with the help of the reactive extensions.

Write sequences of values to the target

```
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information (Symbol 'i : INT' in PLC Global Variables list.
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["GVL.i"];

    // Produces object Values 0,1,2,3 ... in seconds period
    IObservable<object> timerObservable = Observable.Interval(TimeSpan.FromSeconds(1.0)).Select(i =>
(object) (short)i);

    // Take 10 Values (0..9) and write them to GVL.i
    IDisposable dispose = cycleCount.WriteValues(timerObservable.Take(10));

    Console.ReadKey(); // Wait for Key press
    dispose.Dispose(); // Dispose the Subscription
}
```

Reference

[ValueSymbolExtensions Class](#) [[▶ 1333](#)]

[WriteValues Overload](#) [[▶ 1347](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

6.3.4.1.3.2 ValueSymbolExtensions.WriteValues Method (IValueSymbol, IObservable.Object., Action.Exception.)

Subscribes the [IValueSymbol](#) [[▶ 2775](#)] to an observable sequence of values and writes them to the [IValueSymbol](#) [[▶ 2775](#)].

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IDisposable WriteValues(
    this IValueSymbol symbol,
    IObservable<Object> valueObservable,
    Action<Exception> errorHandler
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IValueSymbol [▸ 2775] The symbol.
valueObservable	Type: System.IObservable.Object . Observable of Values.
errorHandler	Type: System.Action.Exception . The error handler or NULL.

Return Value

Type: [IDisposable](#)
IDisposable.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IValueSymbol \[▸ 2775\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[ValueSymbolExtensions Class \[▸ 1333\]](#)

[WriteValues Overload \[▸ 1347\]](#)

[TwinCAT.Ads.Reactive Namespace \[▸ 1249\]](#)

6.3.4.1.3.3 ValueSymbolExtensions.WriteValues Method (IValueSymbol, IObservable.Object., CancellationToken)

Subscribes the [IValueSymbol \[▸ 2775\]](#) to an observable sequence of values and writes them to the [IValueSymbol \[▸ 2775\]](#).

Namespace: [TwinCAT.Ads.Reactive \[▸ 1249\]](#)

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static void WriteValues(
    this IValueSymbol symbol,
    IObservable<Object> valueObservable,
    CancellationToken cancel
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IValueSymbol [▸ 2775] The symbol.
valueObservable	Type: System.IObservable.Object . Observable of Values.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type:
IDisposable.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IValueSymbol \[► 2775\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[ValueSymbolExtensions Class \[► 1333\]](#)

[WriteValues Overload \[► 1347\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.4.1.3.4 ValueSymbolExtensions.WriteValues Method (IValueSymbol, IObservable.Object., Action.Exception., CancellationToken)

Subscribes the [IValueSymbol \[► 2775\]](#) to an observable sequence of values and writes them to the [IValueSymbol \[► 2775\]](#).

Namespace: [TwinCAT.Ads.Reactive \[► 1249\]](#)

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static void WriteValues(
    this IValueSymbol symbol,
    IObservable<Object> valueObservable,
    Action<Exception> errorHandler,
    CancellationToken cancel
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IValueSymbol [► 2775] The symbol.
valueObservable	Type: System.IObservable.Object. Observable of Values.
errorHandler	Type: System.Action.Exception. The error handler.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type:
IDisposable.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IValueSymbol \[► 2775\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[ValueSymbolExtensions Class \[► 1333\]](#)

[WriteValues Overload \[► 1347\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

6.3.4.1.4 ValueSymbolExtensions.PollValues Method

Overload List

	Name	Description
	PollValues(I SumRead, IObservable.Unit.) [► 1359]	Polls a series of symbols via a ISumRead [► 1596] command. The SumCommand will read all contained values with every trigger.
	PollValues(I SumRead, TimeSpan) [► 1361]	Polls a series of symbols via a ISumRead2.S. [► 1601] command. The SumCommand will read all contained values with every trigger in one roundtrip.
	PollValues(I ValueSymbol, IObservable.Unit.) [► 1355]	Poll symbol values on trigger signals.
	PollValues(I ValueSymbol, TimeSpan) [► 1356]	Polls the symbol as value sequence of object values with a specified period time.
	PollValues.T. (I ValueSymbol, IObservable.Unit.) [► 1362]	Poll symbol values as a value sequence on trigger signals (typed)
	PollValues.T. (I ValueSymbol, TimeSpan) [► 1363]	Polls the symbol as value sequence of object values with a specified period time (typed)
	PollValues(I ValueSymbol, IObservable.Unit., Boolean) [► 1357]	Polls symbol values on trigger signals.
	PollValues(I ValueSymbol, TimeSpan, Boolean) [► 1358]	Polls the symbol as value sequence of object values with a specified period time.
	PollValues.T. (I ValueSymbol, IObservable.Unit., Boolean) [► 1364]	Poll symbol values on trigger signals (typed)
	PollValues.T. (I ValueSymbol, IObservable.Unit., Func.ResultReadValueAccess2.I ValueSymbol, Object., T.) [► 1365]	Poll symbol values on trigger signals (typed)

	Name	Description
	PollValues.T. (IValueSymbol , TimeSpan , Boolean) [▶ 1367]	Polls the symbol as value sequence of object values with a specified period time (typed)

Reference

[ValueSymbolExtensions Class](#) [▶ [1333](#)]

[TwinCAT.Ads.Reactive Namespace](#) [▶ [1249](#)]

Also see about this

- 📖 [ValueSymbolExtensions.PollValues Method \(ISumSymbolRead, IObservable.Unit.\)](#) [▶ [1352](#)]
- 📖 [ISumSymbolRead Interface](#) [▶ [1587](#)]
- 📖 [ValueSymbolExtensions.PollValues Method \(ISumSymbolRead, TimeSpan\)](#) [▶ [1353](#)]

6.3.4.1.4.1 ValueSymbolExtensions.PollValues Method (ISumSymbolRead, IObservable.Unit.)

Polls a series of symbols via a [ISumSymbolRead](#) [▶ [1587](#)] command. The SumCommand will read all contained values with every trigger.

Namespace: [TwinCAT.Ads.Reactive](#) [▶ [1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public static IObservable<ResultSumValues> PollValues (
    this ISumSymbolRead sumRead,
    IObservable<Unit> trigger
)
```

Parameters

- | | |
|---------|---|
| sumRead | Type: TwinCAT.Ads.SumCommand.ISumSymbolRead [▶ 1587]
The SumSymbolRead command. |
| trigger | Type: System.IObservable.Unit .
The Polling trigger. |

Return Value

Type: [IObservable.ResultSumValues](#) [▶ [1547](#)].
IObservable<ResultSumValues>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [ISumSymbolRead](#) [▶ [1587](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

The characteristic of this overload is that more than one values can be read with one trigger signal. Each trigger produces only one Read call to the SumSymbolRead. Dependant on the configuration of the SumSymbolRead this could mean only one ADS Roundtrip (Request/Response). So the advantages are:

- Usage of one trigger (could be a Background thread resource) for all values
- Getting the values of all values in one ADS roundtrip
- Values inside the SumSymbolRead are more consistant dependant of target realtime task configuration.

Examples

Demonstration of polling values efficiently via SumSymbolRead command.

Observe multiple ADS Symbols via Polling of SumCommand

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];
    IValueSymbol lastExecTime = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo.LastExecTime"];
    List<ISymbol> symbols = new List<ISymbol>() { cycleCount, lastExecTime };

    //Create the SumCommand
    SumSymbolRead sumRead = new SumSymbolRead(client, symbols);

    // Reactive Notification Handler
    var sumCommandObserver = Observer.Create<ResultSumValues>(result =>
    {
        Console.WriteLine($"SumCommand Succeeded: {result.OverallSucceeded}, CycleCount: {result.Values[0]}, LastExecTime: {result.Values[1]}");
    });

    // Take 20 Values/Result in an Interval of 500ms
    IDisposable subscription = sumRead.PollValues(TimeSpan.FromMilliseconds(500)).Take(20).Subscribe(sumCommandObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[ValueSymbolExtensions Class](#) [► 1333]

[PollValues Overload](#) [► 1351]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

[TwinCAT.Ads.SumCommand.ISumSymbolRead](#) [► 1587]

[ValueSymbolExtensions.PollValues2\(IValueSymbol, IObservable.Unit.\)](#) [► 1368]

6.3.4.1.4.2 ValueSymbolExtensions.PollValues Method (ISumSymbolRead, TimeSpan)

Polls a series of symbols via a [ISumSymbolRead](#) [► 1587] command. The SumCommand will read all contained values with every trigger.

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public static IObservable<ResultSumValues> PollValues(
    this ISumSymbolRead sumRead,
    TimeSpan period
)
```

Parameters

sumRead	Type: TwinCAT.Ads.SumCommand.ISumSymbolRead [► 1587] The SumSymbolRead command.
period	Type: System.TimeSpan The time period for polling values.

Return Value

Type: [IObservable.ResultSumValues \[► 1547\]](#).
IObservable<ResultSumValues>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [ISumSymbolRead \[► 1587\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

The characteristic of this overload is that more than one values can be read with one trigger signal. Each trigger produces only one Read call to the SumSymbolRead. Dependant on the configuration of the SumSymbolRead this could mean only one ADS Roundtrip (Request/Response). So the advantages are:

- Usage of one trigger (could be a Background thread resource) for all values
- Getting the values of all values in one ADS roundtrip
- Values inside the SumSymbolRead are more consistant dependant of target realtime task configuration.

Examples

Demonstration of polling values efficiently via SumSymbolRead command.

Observe multiple ADS Symbols via Polling of SumCommand

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];
    IValueSymbol lastExecTime = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo.LastExecTime"];
    List<ISymbol> symbols = new List<ISymbol>() { cycleCount, lastExecTime };

    //Create the SumCommand
    SumSymbolRead sumRead = new SumSymbolRead(client, symbols);

    // Reactive Notification Handler
    var sumCommandObserver = Observer.Create<ResultSumValues>(result =>
    {
        Console.WriteLine($"SumCommand Succeeded: {result.OverallSucceeded}, CycleCount: {result.Values[0]}, LastExecTime: {result.Values[1]}");
    })
}
```

```
);

// Take 20 Values/Result in an Interval of 500ms
IDisposable subscription = sumRead.PollValues(TimeSpan.FromMilliseconds(500)).Take(20).Subscribe
(sumCommandObserver);

Console.ReadKey(); // Wait for Key press
subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[ValueSymbolExtensions Class \[▶ 1333\]](#)

[PollValues Overload \[▶ 1351\]](#)

[TwinCAT.Ads.Reactive Namespace \[▶ 1249\]](#)

[TwinCAT.Ads.SumCommand.ISumSymbolRead \[▶ 1587\]](#)

[ValueSymbolExtensions.PollValues2\(IValueSymbol, IObservable.Unit.\) \[▶ 1368\]](#)

6.3.4.1.4.3 ValueSymbolExtensions.PollValues Method (IValueSymbol, IObservable.Unit.)

Poll symbol values on trigger signals.

Namespace: [TwinCAT.Ads.Reactive \[▶ 1249\]](#)

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object> PollValues(
    this IValueSymbol symbol,
    IObservable<Unit> trigger
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IValueSymbol [▶ 2775] The symbol.
trigger	Type: System.IObservable.Unit. The polling trigger.

Return Value

Type: [IObservable.Object.](#)
IObservable<System.Object>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IValueSymbol \[▶ 2775\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

If ReadRequest errors occur, the observable will be closed with and [AdsException \[▶ 61\]](#)./>

Reference

[ValueSymbolExtensions Class \[▶ 1333\]](#)

[PollValues Overload \[▶ 1351\]](#)

[TwinCAT.Ads.Reactive Namespace \[▶ 1249\]](#)

[ValueSymbolExtensions.PollValues2\(IValueSymbol, IObservable.Unit.\) \[▶ 1368\]](#)

[ValueSymbolExtensions.PollValues\(IValueSymbol, TimeSpan\) \[▶ 1356\]](#)

Also see about this

- [ValueSymbolExtensions.PollValues2 Method \(IValueSymbol, IObservable.Unit.\) \[▶ 1368\]](#)
- [ValueSymbolExtensions.PollValues2 Method \(IValueSymbol, TimeSpan\) \[▶ 1369\]](#)

6.3.4.1.4.4 ValueSymbolExtensions.PollValues Method (IValueSymbol, TimeSpan)

Polls the symbol as value sequence of object values with a specified period time.

Namespace: [TwinCAT.Ads.Reactive \[▶ 1249\]](#)

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object> PollValues(
    this IValueSymbol symbol,
    TimeSpan period
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IValueSymbol [▶ 2775] The symbol.
period	Type: System.TimeSpan The period.

Return Value

Type: [IObservable.Object](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IValueSymbol \[▶ 2775\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

Here, the values are polled in a specific time period and sequential Reads are triggered (in opposite to ADS Notification in the latter example)

Observe changing ADS Symbols (Read Polling)

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];
}
```

```
// Reactive Notification Handler
var valueObserver = Observer.Create<object>(val =>
{
    Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToString()));
}
);

// Take 20 Values in an Interval of 500ms
IDisposable subscription = cycleCount.PollValues(TimeSpan.FromMilliseconds(500)).Take(20).Subscribe(valueObserver);

Console.ReadKey(); // Wait for Key press
subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[ValueSymbolExtensions Class \[▶ 1333\]](#)

[PollValues Overload \[▶ 1351\]](#)

[TwinCAT.Ads.Reactive Namespace \[▶ 1249\]](#)

6.3.4.1.4.5 ValueSymbolExtensions.PollValues Method (IValueSymbol, IObservable.Unit., Boolean)

Polls symbol values on trigger signals.

Namespace: [TwinCAT.Ads.Reactive \[▶ 1249\]](#)

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object?> PollValues(
    this IValueSymbol symbol,
    IObservable<Unit> trigger,
    bool ignoreErrors
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IValueSymbol [▶ 2775] The symbol.
trigger	Type: System.IObservable.Unit. The Polling trigger
ignoreErrors	Type: System.Boolean if set to true ADS errors are ignored and a NULL value is returned.

Return Value

Type: [IObservable.Object.](#)
IObservable<System.Object>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IValueSymbol \[▶ 2775\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

If ignoreErrors is not set and ReadRequest errors occur, the observable will be closed with error./> If errors are ignored, the observable will return **NULL** values on erroneous requests.

Reference

[ValueSymbolExtensions Class](#) [► 1333]

[PollValues Overload](#) [► 1351]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

[ValueSymbolExtensions.PollValues2\(IValueSymbol, IObservable.Unit.\)](#) [► 1368]

[ValueSymbolExtensions.PollValues\(IValueSymbol, IObservable.Unit.\)](#) [► 1355]

[ValueSymbolExtensions.PollValues\(IValueSymbol, TimeSpan, Boolean\)](#) [► 1358]

Also see about this

[ValueSymbolExtensions.PollValues2 Method \(IValueSymbol, IObservable.Unit.\)](#) [► 1368]

[ValueSymbolExtensions.PollValues2 Method \(IValueSymbol, TimeSpan\)](#) [► 1369]

6.3.4.1.4.6 ValueSymbolExtensions.PollValues Method (IValueSymbol, TimeSpan, Boolean)

Polls the symbol as value sequence of object values with a specified period time.

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<Object?> PollValues (
    this IValueSymbol symbol,
    TimeSpan period,
    bool ignoreErrors
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IValueSymbol [► 2775] The symbol.
period	Type: System.TimeSpan The period.
ignoreErrors	Type: System.Boolean Ignore errors (NULL values will be returned)

Return Value

Type: [IObservable.Object](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IValueSymbol](#) [► 2775]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

If ignoreErrors is not set and ReadRequest errors occur, the observable will be closed with error ([AdsException](#) [▶ 61]).> If errors are ignored, the observable will return **NULL** values on erroneous requests.

Examples

Here, the values are polled in a specific time period and sequential Reads are triggered (in opposite to ADS Notification in the latter example)

Observe changing ADS Symbols (Read Polling)

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];

    // Reactive Notification Handler
    var valueObserver = Observer.Create<object>(val =>
    {
        Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToString()));
    });

    // Take 20 Values in an Interval of 500ms
    IDisposable subscription = cycleCount.PollValues(TimeSpan.FromMilliseconds(500)).Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[ValueSymbolExtensions Class](#) [▶ 1333]

[PollValues Overload](#) [▶ 1351]

[TwinCAT.Ads.Reactive Namespace](#) [▶ 1249]

[ValueSymbolExtensions.PollValues\(IValueSymbol, IObservable.Unit., Boolean\)](#) [▶ 1357]

[ValueSymbolExtensions.PollValues\(IValueSymbol, TimeSpan\)](#) [▶ 1356]

6.3.4.1.4.7 ValueSymbolExtensions.PollValues Method (ISumRead, IObservable.Unit.)

Polls a series of symbols via a [ISumRead](#) [▶ 1596] command. The SumCommand will read all contained values with every trigger.

Namespace: [TwinCAT.Ads.Reactive](#) [▶ 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultSumValues> PollValues(
    this ISumRead sumRead,
    IObservable<Unit> trigger
)
```

Parameters

sumRead	Type: TwinCAT.Ads.SumCommand.ISumRead [▶ 1596] The SumRead command.
trigger	Type: System.IObservable.Unit . The Polling trigger.

Return Value

Type: [IObservable.ResultSumValues](#) [[▶ 1547](#)].
IObservable<ResultSumValues>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [ISumRead](#) [[▶ 1596](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

The characteristic of this overload is that more than one values can be read with one trigger signal. Each trigger produces only one Read call to the SumRead. Dependent on the configuration of the SumRead this could mean only one ADS Roundtrip (Request/Response). So the advantages are:

- Usage of one trigger (could be a Background thread resource) for all values
- Getting the values of all values in one ADS roundtrip
- Values inside the SumRead are more consistant dependant of target realtime task configuration.

Examples

Demonstration of polling values efficiently via SumRead command.

Observe multiple ADS Symbols via Polling of SumCommand

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];
    IValueSymbol lastExecTime = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo.LastExecTime"];
    List<ISymbol> symbols = new List<ISymbol>() { cycleCount, lastExecTime };

    //Create the SumCommand
    SumSymbolRead sumRead = new SumSymbolRead(client, symbols);

    // Reactive Notification Handler
    var sumCommandObserver = Observer.Create<ResultSumValues>(result =>
    {
        Console.WriteLine($"SumCommand Succeeded: {result.OverallSucceeded}, CycleCount: {result.Values[0]}, LastExecTime: {result.Values[1]}");
    });

    // Take 20 Values/Result in an Interval of 500ms
    IDisposable subscription = sumRead.PollValues(TimeSpan.FromMilliseconds(500)).Take(20).Subscribe(sumCommandObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```


Reference

[ValueSymbolExtensions Class](#) [► 1333]

[PollValues Overload](#) [► 1351]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

[TwinCAT.Ads.SumCommand.ISumRead](#) [► 1596]

[ValueSymbolExtensions.PollValues2\(IValueSymbol, IObservable.Unit.\)](#) [► 1368]

6.3.4.1.4.8 ValueSymbolExtensions.PollValues Method (ISumRead, TimeSpan)

Polls a series of symbols via a [ISumRead2.S.](#) [► 1601] command. The SumCommand will read all contained values with every trigger in one roundtrip.

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultSumValues> PollValues(
    this ISumRead sumRead,
    TimeSpan period
)
```

Parameters

sumRead	Type: TwinCAT.Ads.SumCommand.ISumRead [► 1596] The SumRead command.
period	Type: System.TimeSpan The time period for polling values.

Return Value

Type: [IObservable.ResultSumValues](#) [► 1547].
IObservable<ResultSumValues>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [ISumRead](#) [► 1596]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

The characteristic of this overload is that more than one values can be read with one trigger signal. Each trigger produces only one Read call to the SumRead. Dependent on the configuration of the SumRead this could mean only one ADS Roundtrip (Request/Response). So the advantages are:

- Usage of one trigger (could be a Background thread resource) for all values
- Getting the values of all values in one ADS roundtrip
- Values inside the SumRead are more consistant dependant of target realtime task configuration.

Examples

Demonstration of polling values efficiently via SumRead command.

Observe multiple ADS Symbols via Polling of SumCommand

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];
    IValueSymbol lastExecTime = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo.LastExecTime"];
    List<ISymbol> symbols = new List<ISymbol>() { cycleCount, lastExecTime };

    //Create the SumCommand
    SumSymbolRead sumRead = new SumSymbolRead(client, symbols);

    // Reactive Notification Handler
    var sumCommandObserver = Observer.Create<ResultSumValues>(result =>
    {
        Console.WriteLine($"SumCommand Succeeded: {result.OverallSucceeded}, CycleCount: {result.Values[0]}, LastExecTime: {result.Values[1]}");
    });

    // Take 20 Values/Result in an Interval of 500ms
    IDisposable subscription = sumRead.PollValues(TimeSpan.FromMilliseconds(500)).Take(20).Subscribe(sumCommandObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[ValueSymbolExtensions Class \[► 1333\]](#)

[PollValues Overload \[► 1351\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

[TwinCAT.Ads.SumCommand.ISumRead \[► 1596\]](#)

[ValueSymbolExtensions.PollValues2.S.\(ISumRead2.S., TimeSpan\) \[► 1372\]](#)

6.3.4.1.4.9 ValueSymbolExtensions.PollValues.T. Method (IValueSymbol, IObservable.Unit.)

Poll symbol values as a value sequence on trigger signals (typed)

Namespace: [TwinCAT.Ads.Reactive \[► 1249\]](#)

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> PollValues<T>(
    this IValueSymbol symbol,
    IObservable<Unit> trigger
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IValueSymbol [► 2775] The symbol.
trigger	Type: System.IObservable.Unit. The Polling trigger

Type Parameters

T

Return Value

Type: [IObservable.T](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IValueSymbol](#) [[▶ 2775](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

Read errors will end the observable with [AdsException](#) [[▶ 61](#)] error.

Reference

[ValueSymbolExtensions Class](#) [[▶ 1333](#)]

[PollValues Overload](#) [[▶ 1351](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

[ValueSymbolExtensions.PollValues.T\(IValueSymbol, IObservable.Unit, Boolean\)](#) [[▶ 1364](#)]

[ValueSymbolExtensions.PollValues.T\(IValueSymbol, TimeSpan\)](#) [[▶ 1363](#)]

6.3.4.1.4.10 ValueSymbolExtensions.PollValues.T. Method (IValueSymbol, TimeSpan)

Polls the symbol as value sequence of object values with a specified period time (typed)

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: [TwinCAT.Ads.Reactive](#) (in [TwinCAT.Ads.Reactive.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> PollValues<T>(
    this IValueSymbol symbol,
    TimeSpan period
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IValueSymbol [▶ 2775] The symbol.
period	Type: System.TimeSpan The time period.

Type Parameters

T

Return Value

Type: [IObservable.T](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IValueSymbol](#) [[▶ 2775](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

Here, the values are polled in a specific time period and sequential Reads are triggered (in opposite to ADS Notification in the latter example)

Observe changing ADS Symbols (Read Polling)

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];

    // Reactive Notification Handler
    var valueObserver = Observer.Create<object>(val =>
    {
        Console.WriteLine(string.Format("Instance: {0}, Value: {1}", cycleCount.InstancePath, val.ToString()));
    });

    // Take 20 Values in an Interval of 500ms
    IDisposable subscription = cycleCount.PollValues(TimeSpan.FromMilliseconds(500)).Take(20).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[ValueSymbolExtensions Class](#) [[▶ 1333](#)]

[PollValues Overload](#) [[▶ 1351](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

6.3.4.1.4.11 ValueSymbolExtensions.PollValues.T. Method (IValueSymbol, IObservable.Unit., Boolean)

Poll symbol values on trigger signals (typed)

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> PollValues<T>(
    this IValueSymbol symbol,
    IObservable<Unit> trigger,
    bool ignoreErrors
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IValueSymbol [▶ 2775] The symbol.
trigger	Type: System.IObservable.Unit . The Polling trigger
ignoreErrors	Type: System.Boolean Ignore errors.

Type Parameters

T	The type of the values.
---	-------------------------

Return Value

Type: [IObservable.T](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IValueSymbol](#) [[▶ 2775](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

If ignoreErrors is not set and ReadRequest errors occur, the observable will be closed with error ([AdsException](#) [[▶ 61](#)])./> If errors are ignored, the observable will return **NULL** values on erroneous requests.

Reference

[ValueSymbolExtensions Class](#) [[▶ 1333](#)]

[PollValues Overload](#) [[▶ 1351](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

[ValueSymbolExtensions.PollValues.T.\(IValueSymbol, IObservable.Unit.\)](#) [[▶ 1362](#)]

[ValueSymbolExtensions.PollValues.T.\(IValueSymbol, TimeSpan\)](#) [[▶ 1363](#)]

6.3.4.1.4.12 ValueSymbolExtensions.PollValues.T. Method (IValueSymbol, IObservable.Unit., Func.ResultReadValueAccess2.IValueSymbol, Object., T.)

Poll symbol values on trigger signals (typed)

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<T> PollValues<T>(
    this IValueSymbol symbol,
    IObservable<Unit> trigger,
    Func<ResultReadValueAccess2<IValueSymbol, Object?>, T?> errorHandler
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IValueSymbol [► 2775] The symbol.
trigger	Type: System.IObservable.Unit . The Polling trigger
errorHandler	Type: System.Func.ResultReadValueAccess2 [► 3226].IValueSymbol [► 2775] , Object. , T. The error handler

Type Parameters

T	The type of the values
---	------------------------

Return Value

Type: [IObservable.T](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IValueSymbol \[► 2775\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

If an read error occurs an optional error handler will be called. This decides the return value. If no errorHandler is specified, an [AdsException \[► 61\]](#) is thrown on error.

Examples

Example for polling typed values and usage of an error handler.

Observe changing ADS Symbols (Read Polling)

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];

    // Reactive Notification Handler
    var valueObserver = Observer.Create<ushort>(val =>
    {
        Console.WriteLine($"Instance: {cycleCount}, Value: {val}");
    });

    // Creating an interval trigger.
    IObservable<Unit> interval = Observable.Interval(TimeSpan.FromMilliseconds(500)).StartWith(-1L).
    Select(1 => Unit.Default);
```

```

// Take 20 Values in an Interval of 500ms
IDisposable subscription = cycleCount.PollValues<ushort>(interval,
// Usage of an error handler
(e) =>
{
if (e.ErrorCode == (int) AdsErrorCode.ClientSyncTimeOut)
{
// On Timeout return Value 0!
return 0;
}
else
{
// Finish observer
throw new AdsException($"Internal Error Code: {e.ErrorCode} (Source: {e.Source})");
}
}
)
.Take(20).Subscribe(valueObserver);

Console.ReadKey(); // Wait for Key press
subscription.Dispose(); // Dispose the Subscription
}

```

Reference

[ValueSymbolExtensions Class \[► 1333\]](#)

[PollValues Overload \[► 1351\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

[ValueSymbolExtensions.PollValues.T.\(IValueSymbol, IObservable.Unit.\) \[► 1362\]](#)

[ValueSymbolExtensions.PollValues.T.\(IValueSymbol, TimeSpan\) \[► 1363\]](#)

6.3.4.1.4.13 ValueSymbolExtensions.PollValues.T. Method (IValueSymbol, TimeSpan, Boolean)

Polls the symbol as value sequence of object values with a specified period time (typed)

Namespace: [TwinCAT.Ads.Reactive \[► 1249\]](#)

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public static IObservable<T> PollValues<T>(
    this IValueSymbol symbol,
    TimeSpan period,
    bool ignoreErrors
)

```

Parameters

symbol	Type: TwinCAT.TypeSystem.IValueSymbol [► 2775] The symbol.
period	Type: System.TimeSpan The period.
ignoreErrors	Type: System.Boolean Ignore read errors (default values will be returned).

Type Parameters

T

Return Value

Type: [IObservable.T](#).
[IObservable<System.Object>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IValueSymbol](#) [[▶ 2775](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

If ignoreErrors is not set and ReadRequest errors occur, the observable will be closed with error ([AdsException](#) [[▶ 61](#)]).> If errors are ignored, the observable will return **NULL** values on erroneous requests.

Reference

[ValueSymbolExtensions Class](#) [[▶ 1333](#)]

[PollValues Overload](#) [[▶ 1351](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

6.3.4.1.5 ValueSymbolExtensions.PollValues2 Method

Overload List

	Name	Description
	PollValues2.S. (ISumRead2.S. , IObservable.Unit.) [▶ 1370]	Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger in one roundtrip.
	PollValues2.S. (ISumRead2.S. , TimeSpan) [▶ 1372]	Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger.
	PollValues2(IValueS ymbol, IObservable.Unit.) [▶ 1368]	Poll symbol values as a sequence of annotated results (Value + ErrorCode)
	PollValues2(IValueS ymbol, TimeSpan) [▶ 1369]	Poll symbol values with communication return codes.

Reference

[ValueSymbolExtensions Class](#) [[▶ 1333](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

6.3.4.1.5.1 ValueSymbolExtensions.PollValues2 Method (IValueSymbol, IObservable.Unit.)

Poll symbol values as a sequence of annotated results (Value + ErrorCode)

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultReadValueAccess2<IValueSymbol, Object?>> PollValues2(
    this IValueSymbol symbol,
    IObservable<Unit> trigger
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IValueSymbol [▶ 2775] The symbol to poll
trigger	Type: System.IObservable.Unit . The Polling trigger

Return Value

Type: [IObservable.ResultReadValueAccess2](#) [[▶ 3226](#)].[IValueSymbol](#) [[▶ 2775](#)], [Object](#)..
[IObservable<ResultReadValueAccess>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IValueSymbol](#) [[▶ 2775](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

This reactive method polls the symbol version together with additional information. The distinct result data is [ResultReadValueAccess](#) [[▶ 3209](#)] which contains the value (if succeeded) together with the communication return code. If the ErrorCode is > 0, then the Value property will return the default value. This has the advantage, that the Observable doesn't produces Errors when communication requests return Errors and the Observer can continue. Internally for optimization reasons because it is expected that the polling occurs many times, this reactive method creates the symbol handle before the first polling trigger (one roundtrip), reads the value with each trigger and unregisters the handle on finalize. So the number of communication roundtrips are Triggers + 2.

Reference


[ValueSymbolExtensions Class](#) [[▶ 1333](#)]

[PollValues2 Overload](#) [[▶ 1368](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

Also see about this

 [ResultReadValueAccess Class](#) [[▶ 3209](#)]

 [ValueSymbolExtensions.PollValues Method \(IValueSymbol, TimeSpan, Boolean\)](#) [[▶ 1358](#)]

6.3.4.1.5.2 ValueSymbolExtensions.PollValues2 Method (IValueSymbol, TimeSpan)

Poll symbol values with communication return codes.

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultReadValueAccess2<IValueSymbol, Object?>> PollValues2(
    this IValueSymbol symbol,
    TimeSpan period
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IValueSymbol [▶ 2775] The symbol to poll.
period	Type: System.TimeSpan The polling interval.

Return Value

Type: [IObservable.ResultReadValueAccess2](#) [[▶ 3226](#)].[IValueSymbol](#) [[▶ 2775](#)], [Object](#)..
[IObservable<ResultReadValueAccess>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IValueSymbol](#) [[▶ 2775](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

This reactive method polls the symbol version together with additional information. The distinct result data is [ResultReadValueAccess](#) [[▶ 3209](#)] which contains the value (if succeeded) together with the communication return code. If the ErrorCode is > 0, then the Value property will return the default value. This has the advantage, that the Observable doesn't produces Errors when communication requests return Errors and the Observer can continue. Internally for optimization reasons because it is expected that the polling occurs many times, this reactive method creates the symbol handle before the first polling trigger (one roundtrip), reads the value with each trigger and unregisters the handle on finalize. So the number of communication roundtrips is Triggers + 2.

Reference

[ValueSymbolExtensions Class](#) [[▶ 1333](#)]

[PollValues2 Overload](#) [[▶ 1368](#)]

[TwinCAT.Ads.Reactive Namespace](#) [[▶ 1249](#)]

[ValueSymbolExtensions.PollValues\(IValueSymbol, TimeSpan\)](#) [[▶ 1356](#)]

Also see about this

[ResultReadValueAccess Class](#) [[▶ 3209](#)]

[ValueSymbolExtensions.PollValues Method \(IValueSymbol, TimeSpan, Boolean\)](#) [[▶ 1358](#)]

6.3.4.1.5.3 ValueSymbolExtensions.PollValues2.S. Method (ISumRead2.S., IObservable.Unit.)

Polls a series of symbols via a [ISumRead2.S.](#) [[▶ 1601](#)] command. The SumCommand will read all contained values with every trigger in one roundtrip.

Namespace: [TwinCAT.Ads.Reactive](#) [[▶ 1249](#)]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultSumValues2<S>> PollValues2<S>(
    this ISumRead2<S> sumRead,
    IObservable<Unit> trigger
)
```

Parameters

sumRead	Type: TwinCAT.Ads.SumCommand.ISumRead2 [► 1601].S . The SumRead command.
trigger	Type: System.IObservable.Unit . The Polling trigger.

Type Parameters

S	The source specifier, this could be an ISymbol or InstancePath (string) for example to reference the single values with their source.
---	---

Return Value

Type: [IObservable.ResultSumValues2 \[► 1605\].S](#)..
[IObservable<ResultSumValues>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [ISumRead2 \[► 1601\].S](#).. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

The characteristic of this overload is that more than one values can be read with one trigger signal. Each trigger produces only one Read call to the SumRead. Dependent on the configuration of the SumRead this could mean only one ADS Roundtrip (Request/Response). So the advantages are:

- Usage of one trigger (could be a Background thread resource) for all values
- Getting the values of all values in one ADS roundtrip
- Values inside the SumRead are more consistant dependant of target realtime task configuration.

Examples

Demonstration of polling values efficiently via SumRead command.

Observe multiple ADS Symbols via Polling of SumCommand

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];
    IValueSymbol lastExecTime = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo.LastExecTime"];
    List<ISymbol> symbols = new List<ISymbol>() { cycleCount, lastExecTime };

    //Create the SumCommand
    SumSymbolRead sumRead = new SumSymbolRead(client, symbols);

    // Reactive Notification Handler
    var sumCommandObserver = Observer.Create<ResultSumValues2<ISymbol>>(result =>
```

```

    {
        Console.WriteLine($"SumCommand ErrorCode: {result.ErrorCode}");

        if (result.Succeeded)
        {
            Console.WriteLine($"SumCommand OverallSucceeded: {result.OverallSucceeded}");
            foreach (var subResult in result.ValueResults)
            {
                Console.WriteLine($"Source: {subResult.Source}, ErrorCode: {subResult.ErrorCode}, Value:
{subResult.Value}");
            }
        }
    }
);

// Take 20 Values/Result in an Interval of 500ms
IDisposable subscription = sumRead.PollValues2(TimeSpan.FromMilliseconds(500)).Take(20).Subscribe
(sumCommandObserver);

Console.ReadKey(); // Wait for Key press
subscription.Dispose(); // Dispose the Subscription
}

```

Reference

[ValueSymbolExtensions Class](#) [► 1333]

[PollValues2 Overload](#) [► 1368]

[TwinCAT.Ads.Reactive Namespace](#) [► 1249]

[TwinCAT.Ads.SumCommand.ISumRead2.S.](#) [► 1601]

[ValueSymbolExtensions.PollValues\(ISumRead, IObservable.Unit.\)](#) [► 1359]

6.3.4.1.5.4 ValueSymbolExtensions.PollValues2.S. Method (ISumRead2.S., TimeSpan)

Polls a series of symbols via a [ISumRead2.S.](#) [► 1601] command. The SumCommand will read all contained values with every trigger.

Namespace: [TwinCAT.Ads.Reactive](#) [► 1249]

Assembly: TwinCAT.Ads.Reactive (in TwinCAT.Ads.Reactive.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public static IObservable<ResultSumValues2<S>> PollValues2<S>(
    this ISumRead2<S> sumRead,
    TimeSpan period
)

```

Parameters

sumRead	Type: TwinCAT.Ads.SumCommand.ISumRead2 [► 1601].S. The SumRead command.
period	Type: System.TimeSpan The time period for polling values.

Type Parameters

S	The source specifier, this could be an ISymbol or an InstancePath for example to reference the symbol.
---	--

Return Value

Type: [IObservable.ResultSumValues2 \[► 1605\].S..](#)
 IObservable<ResultSumValues>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [ISumRead2 \[► 1601\].S..](#) When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

The characteristic of this overload is that more than one values can be read with one trigger signal. Each trigger produces only one Read call to the SumRead. Dependent on the configuration of the SumRead this could mean only one ADS Roundtrip (Request/Response). So the advantages are:

- Usage of one trigger (could be a Background thread resource) for all values
- Getting the values of all values in one ADS roundtrip
- Values inside the SumRead are more consistant dependant of target realtime task configuration.

Examples

Demonstration of polling values efficiently via SumRead command.

Observe multiple ADS Symbols via Polling of SumCommand

```
// To Test the Observer run a project on the local PLC System (Port 851)
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    IValueSymbol cycleCount = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo[1].CycleCount"];
    IValueSymbol lastExecTime = (IValueSymbol)symbolLoader.Symbols["TwinCAT_SystemInfoVarList._TaskInfo.LastExecTime"];
    List<ISymbol> symbols = new List<ISymbol>() { cycleCount, lastExecTime };

    //Create the SumCommand
    SumSymbolRead sumRead = new SumSymbolRead(client, symbols);

    // Reactive Notification Handler
    var sumCommandObserver = Observer.Create<ResultSumValues2<ISymbol>>(result =>
    {
        Console.WriteLine($"SumCommand ErrorCode: {result.ErrorCode}");

        if (result.Succeeded)
        {
            Console.WriteLine($"SumCommand OverallSucceeded: {result.OverallSucceeded}");
            foreach (var subResult in result.ValueResults)
            {
                Console.WriteLine($"Source: {subResult.Source}, ErrorCode: {subResult.ErrorCode}, Value: {subResult.Value}");
            }
        }
    });

    // Take 20 Values/Result in an Interval of 500ms
    IDisposable subscription = sumRead.PollValues2(TimeSpan.FromMilliseconds(500)).Take(20).Subscribe(sumCommandObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[ValueSymbolExtensions Class \[► 1333\]](#)

[PollValues2 Overload \[► 1368\]](#)

[TwinCAT.Ads.Reactive Namespace \[► 1249\]](#)

[TwinCAT.Ads.SumCommand.ISumRead \[► 1596\]](#)





[ValueSymbolExtensions.PollValues\(ISumRead, TimeSpan\) \[► 1361\]](#)


6.4 TwinCAT.Ads.Server Namespace

This framework can be used to build custom ADS server application.

Important note: Please contact Beckhoff to receive a reserved ADS port number. Namespace for the TwinCAT ADS Server Component.

Classes


	Class	Description
	AdsServer [► 1375]	Base implementation for an ADS Server.
	AdsServerException [► 1443]	An AdsServerException [► 1443] is thrown on communication errors in the AdsServer [► 1375] class.
	AdsSymbolicServer [► 1468]	AdsServer class that supports Symbolic information and Notifications. Implements the AdsServer [► 1375]
	BaseTickTrigger [► 1513]	Class NotificationTimer. Implements the INotificationTrigger [► 1516]
	ErrorEventArgs [► 1448]	This class implements the event arguments passed by the TcAdsServerExEvent.
	LoopbackNotConnectedException [► 1523]	The Tcp Loopback client is not connected. Implements the AdsServerException [► 1443]
	LoopbackNotRegisteredException [► 1450]	The Tcp Loopback client is not registered. Implements the AdsServerException [► 1443]
	NotificationDataSample [► 1453]	This class implements an ADS Notification Sample. It contains the notification handle and the variable data.
	NotificationSamplesStamp [► 1456]	This class implements an ADS Stamp Header containing multiple ADS Notification Samples (TcAdsStampHeader)
	NotificationTriggerSource [► 1517]	Class NotificationTriggerSource. Implements the IAddNotificationTrigger [► 1515] Implements the IDisposable
	ServerConnectionStateChangedEventArgs [► 1462]	Class ConnectionStateChangedEventArgs (Server Connections)

	Class	Description
	ServerNotConnectedException [▶ 1465]	The AdsServer is not connected. Implements the AdsServerException [▶ 1443]

Interfaces

	Interface	Description
	IAddNotificationTrigger [▶ 1515]	Interface IAddNotificationTrigger
	INotificationTrigger [▶ 1516]	Interface INotificationTrigger

Enumerations

	Enumeration	Description
	ServerConnectionState [▶ 1461]	The Server Connection State

6.4.1 AdsServer Class

Base implementation for an ADS Server.

Inheritance Hierarchy

[System.Object](#)
 TwinCAT.Ads.Server.AdsServer
 TwinCAT.Ads.Server.AdsSymbolicServer [[▶ 1468](#)]

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229







Syntax





C#

```
public abstract class AdsServer : IDisposable
```












The AdsServer type exposes the following members.



















Properties

	Name	Description
	AmsServer [▶ 1384]	Gets the the internal AmsServer [▶ 1384] object.
	IsConnected [▶ 1384]	Gets a value indicating whether AdsServer is connected.
	IsDisconnecting [▶ 1384]	Indicates, that the AdsServer is actually disconnecting.
	IsDisposed [▶ 1385]	Gets a value indicating whether this instance is disposed.
	IsDynamicPort [▶ 1387]	Gets a value indicating this AdsServer has a dynamic/unfixed port.
	Logger [▶ 1385]	Gets the logger object.











	Name	Description
	<u>ServerAddress</u> [▶ 1386]	The AMS address of this server.
	<u>ServerName</u> [▶ 1386]	Gets the name of the server.
	<u>ServerPort</u> [▶ 1387]	Gets the server port.
	<u>ServerVersion</u> [▶ 1387]	Gets the Version of the Server.

Methods


	Name	Description
	<u>AddDeviceNotificationRequest</u> [▶ 1393]	Sends an ADS Add Device Notification request (synchronous).
	<u>AddDeviceNotificationRequestAsync</u> [▶ 1394]	Sends an ADS Add Device Notification request (async)
	<u>AddDeviceNotificationResponseAsync</u> [▶ 1395]	Sends an ADS Add Device Notification response.
	<u>ConnectServer</u> [▶ 1396]	Connect this ADS server to the local ADS router.
	<u>ConnectServerAndWaitAsync</u> [▶ 1396]	Registers the AdsServer at the router asynchronously.
	<u>DeleteDeviceNotificationRequest</u> [▶ 1398]	Sends an ADS Delete Device Notification request (synchronous).
	<u>DeleteDeviceNotificationRequestAsync</u> [▶ 1399]	Sends an ADS Delete Device Notification request (async).
	<u>DeleteDeviceNotificationResponseAsync</u> [▶ 1400]	Sends an ADS Delete Device Notification response.
	<u>DeviceNotificationRequestAsync</u> [▶ 1400]	Sends an ADS Device Notification request asynchronously
	<u>DeviceNotificationRequestSync</u> [▶ 1401]	Sends an ADS Device Notification request (sync)
	<u>Disconnect</u> [▶ 1402]	Disconnects this ADS server from the local ADS router.
	<u>Dispose.</u> [▶ 1403]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.
	<u>Dispose(Boolean)</u> [▶ 1403]	Releases unmanaged and - optionally - managed resources.

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize [▶ 1403]	Finalizes an instance of the AdsServer class. (Overrides Object.Finalize .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetServerName [▶ 1441]	Gets the name of the server.
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnAddDeviceNotificationAsync [▶ 1429]	Called when an ADS Add Device Notification indication is received.
	OnAddDeviceNotificationConfirmationAsync [▶ 1404]	Called when an ADS Add Device Notification confirmation is received.
	OnBeforeConnected [▶ 1430]	Handler function that is called, when the AdsServer is connected, but before calling OnConnected . [▶ 1405].
	OnConnected [▶ 1405]	Handler function that is called, when the AdsServer is connected.
	OnDeleteDeviceNotificationAsync [▶ 1430]	Called when an ADS Delete Device Notification indication is received.
	OnDeleteDeviceNotificationConfirmationAsync [▶ 1431]	Called when an ADS Delete Device Notification confirmation is received.
	OnDisconnect [▶ 1405]	Called when the AdsServer is about to be disconnected.
	OnReadAsync [▶ 1432]	Called when an ADS Read indication is received.
	OnReadConfirmationAsync [▶ 1433]	Called when an ADS Read confirmation is received.
	OnReadDeviceInfoConfirmationAsync [▶ 1434]	Called when an ADS Read Device Info confirmation is received.
	OnReadDeviceStateAsync [▶ 1435]	Called when an ADS Read State indication is received.
	OnReadDeviceStateConfirmationAsync [▶ 1435]	Called when an ADS Read State confirmation is received.
	OnReadWriteAsync [▶ 1436]	Called when an ADS Read Write indication is received.

	Name	Description
	OnReadWriteConfirmationAsync [▶ 1437]	Called when an ADS Read Write confirmation is received.
	OnRouterNotification [▶ 1405]	Handler Function for a Router Notification.
	OnServerConnectionStateChanged [▶ 1406]	Handles the ServerConnectionStateChanged [▶ 1442] event.
	OnWriteAsync [▶ 1438]	Called when an ADS Write indication is received.
	OnWriteConfirmationAsync [▶ 1439]	Called when an ADS Write confirmation is received.
	OnWriteControlAsync [▶ 1440]	Called when an ADS Write Control indication is received.
	OnWriteControlConfirmationAsync [▶ 1441]	Called when an ADS Write Control confirmation is received.
	ReadDeviceInfoRequestAsync [▶ 1409]	Sends an ADS Read Device Info request asynchronously
	ReadDeviceInfoRequestSync [▶ 1409]	Sends an ADS Read Device Info request synchronously.
	ReadDeviceInfoResponseAsync [▶ 1410]	Sends an ADS Read Device Info response.
	ReadDeviceStateRequestAsync [▶ 1412]	Sends an ADS Read State request (asynchronous)
	ReadDeviceStateRequestSync [▶ 1413]	Sends an ADS Read State request (synchronous)
	ReadDeviceStateResponseAsync [▶ 1413]	Sends an ADS Read State response.
	ReadRequest [▶ 1415]	Sends an ADS Read Request.
	ReadRequestAsync [▶ 1416]	Sends an ADS Read Request asynchronously.
	ReadResponseAsync [▶ 1416]	Sends an ADS Read response.
	ReadWriteRequestAsync [▶ 1419]	Sends an ADS Read Write request.

	Name	Description
	ReadWriteRequestSync [▶ 1419]	Sends an ADS Read Write request synchronously
	ReadWriteResponseAsync [▶ 1420]	Sends an ADS Read Write Response.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	WriteControlRequest [▶ 1423]	Sends an ADS Write Control request (synchronous)
	WriteControlRequestAsync [▶ 1424]	Sends an ADS Write Control request (asynchronous).
	WriteControlRequestSync [▶ 1425]	Sends an ADS Write Control request (synchronous).
	WriteControlResponseAsync [▶ 1425]	Sends an ADS Write Control response.
	WriteRequest [▶ 1427]	Sends an ADS Write request synchronously.
	WriteRequestAsync [▶ 1428]	Sends an ADS Write request asynchronously.
	WriteResponseAsync [▶ 1428]	Sends an ADS Write response.

Events

	Name	Description
	ServerConnectionStateChanged [▶ 1442]	The connection status has changed

Fields

	Name	Description
	serverVersion [▶ 1443]	The version of the AdsServer

Remarks

Derived classes should overwrite the AMS indication methods to react on incoming requests. AMS The confirmation methods should be overwritten to receive replies on asynchronous requests sent by this ADS server.

Examples

The following sample shows how to derive from the AdsServer class and create your own Customized ADS Server.

C#

```

class Program
{
    public static void Main(string[] args)
    {
        CreateHostBuilder(args).Build().Run();
    }

    public static IHostBuilder CreateHostBuilder(string[] args) =>
    Host.CreateDefaultBuilder(args)
        .ConfigureServices((hostContext, services) =>
        {
            services.AddHostedService<ServerWorker>();
        });
}

```

C#

```

public class ServerWorker : BackgroundService
{
    private readonly ILogger<ServerWorker> _logger;

    public ServerWorker(ILogger<ServerWorker> logger)
    {
        _logger = logger;
    }

    protected override async Task ExecuteAsync(CancellationToken cancel)
    {
        // Instantiate the server
        AdsSampleServer server = new AdsSampleServer(_logger);
        // Connect the server and wait for cancel
        await server.ConnectServerAndWaitAsync(cancel);
    }
}

```

C#

```

/*
 * Extend the AdsServer class to implement your own ADS server.
 */
public class AdsSampleServer : AdsServer
{
    /// <summary>
    /// Fixed ADS Port (to be changed ...)
    /// </summary>
    const ushort ADS_PORT = 42;

    /// <summary>
    /// Fixed Name for the ADS Port (change this ...)
    /// </summary>
    const string ADS_PORT_NAME = "AdsSampleServer_Port42";

    /// <summary>
    /// Some simple data / ProcessImage
    /// </summary>
    private byte[] _dataBuffer = {1, 2, 3, 4};

    /// <summary>
    /// Ads State
    /// </summary>
    private AdsState _adsState = AdsState.Config;
    /// <summary>
    /// Device State
    /// </summary>
    private ushort _deviceState = 0;

    /// <summary>
    /// Notification dictionary, thread safe
    /// </summary>
    private ConcurrentDictionary<uint, NotificationRequestEntry> _notificationTable = new Concurrent
Dictionary<uint, NotificationRequestEntry>();

    /// <summary>
    /// Simple counter for different Notification handles here.
    /// </summary>
    private uint _currentNotificationHandle = 0;

    /// <summary>

```

```

/// Logger
/// </summary>
private ILogger _logger;

/* Instantiate an ADS server with a fix ADS port assigned by the ADS router.
*/

public AdsSampleServer()
: this(null)
{
}

public AdsSampleServer(ILogger logger) : base(ADS_PORT, ADS_PORT_NAME)
{
base.serverVersion = new Version(0, 0, 1);
_logger = logger;
}

/* Overwrite the indication handlers of the AdsServer class for the services your ADS server
* provides. They are called upon incoming requests. Indications that are not overwritten in
* this class return the ADS DeviceServiceNotSupported error code to the requester.
*/

/* Handler function for Write Indication */
protected override Task<ResultWrite> OnWriteAsync(AmsAddress target, uint invokeId, uint indexGr
oup, uint indexOffset, ReadOnlyMemory<byte> writeData, CancellationToken cancel)
{
ResultWrite result;

switch (indexGroup) /* use index group (and offset) to distinguish between the services
of this server */
{
case 0x10000:
if (writeData.Length == 4)
{
writeData.CopyTo(_dataBuffer);
result = ResultWrite.CreateSuccess();
}
else
{
result = ResultWrite.CreateError(AdsErrorCode.DeviceInvalidSize);
}
break;

case 0x20000: /* used for the PLC Sample */
if (writeData.Length == 4)
{
uint value = BinaryPrimitives.ReadUInt32LittleEndian(writeData.Span);
result = ResultWrite.CreateSuccess();
}
else
{
result = ResultWrite.CreateError(AdsErrorCode.DeviceInvalidSize);
}

break;

default: /* other services are not supported */
result = ResultWrite.CreateError(AdsErrorCode.DeviceServiceNotSupported);
break;
}
return Task.FromResult(result);
}

/* Handler function for Read Indication */
protected override Task<ResultReadBytes> OnReadAsync(AmsAddress target, uint invokeId, uint inde
xGroup, uint indexOffset, int readLength, CancellationToken cancel)
{
ResultReadBytes result;

/* Distinguish between services like in OnWriteAsync */
result = ResultReadBytes.CreateSuccess(_dataBuffer.AsMemory());

// or Error with ErrorCode
// result = ResultReadBytes.CreateError(AdsErrorCode.DeviceNotSupported);
return Task.FromResult(result);
}

/* Handler function for ReadWrite Indication */
protected override Task<ResultReadWriteBytes> OnReadWriteAsync(AmsAddress target, uint invokeId,

```

```

uint indexGroup, uint indexOffset, int readLength, ReadOnlyMemory<byte> writeData, CancellationTok
n cancel)
{
    ResultReadWriteBytes result;

    /* Distinguish between services like in AdsWriteInd */

    if (readLength == 4 && writeData.Length == 4)
    {
        result = ResultReadWriteBytes.CreateSuccess(_dataBuffer.AsMemory(0, 4));
    }
    else
    {
        result = ResultReadWriteBytes.CreateError(AdsErrorCode.DeviceInvalidSize);
    }

    return Task.FromResult(result);
}

/* Handler function for ReadDeviceState Indication */
protected override Task<ResultReadDeviceState> OnReadDeviceStateAsync(AmsAddress target, uint in
vokeId, CancellationToken cancel)
{
    StateInfo state = new StateInfo(_adsState, _deviceState);
    ResultReadDeviceState result = ResultReadDeviceState.CreateSuccess(state);
    return Task.FromResult(result);
}

/* Handler function for WriteControl Indication */
protected override Task<ResultAds> OnWriteControlAsync(AmsAddress target, uint invokeId, AdsStat
e adsState, ushort deviceState, ReadOnlyMemory<byte> data, CancellationToken cancel)
{
    // Set requested ADS and device status

    _adsState = adsState;
    _deviceState = deviceState;

    ResultAds result = ResultAds.CreateSuccess();
    return Task.FromResult(result);
}

/* Handler function for AddDeviceNotification Indication */
protected override Task<ResultHandle> OnAddDeviceNotificationAsync(AmsAddress target, uint invok
eId, uint indexGroup, uint indexOffset, int dataLength, AmsAddress receiver, NotificationSettings se
ttings, CancellationToken cancel)
{
    /* Create a new notification entry and store it in the notification table */
    NotificationRequestEntry notEntry = new NotificationRequestEntry(receiver, indexGroup, indexOffs
et, dataLength, settings);

    _notificationTable.AddOrUpdate(_currentNotificationHandle, notEntry, (key, value) => notEntry);
    ResultHandle result = ResultHandle.CreateSuccess(_currentNotificationHandle);

    _currentNotificationHandle++;
    return Task.FromResult(result);
}

/* Handler function for DeleteDeviceNotification Indication */
protected override Task<ResultAds> OnDeleteDeviceNotificationAsync(AmsAddress target, uint invok
eId, uint hNotification, CancellationToken cancel)
{
    ResultAds result;

    /* check if the requested notification handle is still in the notification table */
    if (_notificationTable.ContainsKey(hNotification))
    {
        NotificationRequestEntry entry = null;
        _notificationTable.TryRemove(hNotification, out entry);
        result = ResultAds.CreateSuccess();
    }
    else // notification handle is not in the notification table -> return an error code
        // to the requester
    {
        result = ResultAds.CreateError(AdsErrorCode.DeviceNotifyHandleInvalid);
    }

    return Task.FromResult(result);
}

```

```

/* Handler function for DeviceNotification Indication */
protected override Task<ResultAds> OnDeviceNotificationAsync(AmsAddress sender, NotificationSamplesStamp[] stampHeaders, CancellationToken cancel)
{
    /*
    * Call notification handlers.
    */
    return Task.FromResult(ResultAds.CreateSuccess());
}

/// <summary>
/// AdsSampleServer Notification request entry
/// </summary>
internal class NotificationRequestEntry
{
    private AmsAddress _sender; // the AmsNetId of the requester
    private uint _indexGroup; // the requested index group
    private uint _indexOffset; // the requested index offset
    private int _length; // the number of bytes to send
    NotificationSettings _settings; // the notification settings

    internal NotificationRequestEntry(AmsAddress sender,
        uint indexGroup,
        uint indexOffset,
        int dataLength,
        NotificationSettings settings)
    {
        _sender = sender;
        _indexGroup = indexGroup;
        _indexOffset = indexOffset;
        _length = dataLength;
        _settings = settings;
    }
}

```










Reference


[TwinCAT.Ads.Server Namespace |> 1374](#)

6.4.1.1 AdsServer Properties

The [AdsServer |> 1375](#) type exposes the following members.

Properties

	Name	Description
	AmsServer > 1384	Gets the the internal AmsServer > 1384 object.
	IsConnected > 1384	Gets a value indicating whether AdsServer > 1375 is connected.
	IsDisconnecting > 1384	Indicates, that the AdsServer > 1375 is actually disconnecting.
	IsDisposed > 1385	Gets a value indicating whether this instance is disposed.
	IsDynamicPort > 1387	Gets a value indicating this AdsServer > 1375 has a dynamic/unfixed port.
	Logger > 1385	Gets the logger object.
	ServerAddress > 1386	The AMS address of this server.
	ServerName > 1386	Gets the name of the server.
	ServerPort > 1387	Gets the server port.

	Name	Description
	ServerVersion [▶ 1387]	Gets the Version of the Server.

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.1.1 AdsServer.AmsServer Property

Gets the the internal AmsServer object.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected AmsServer AmsServer { get; }
```

Property Value

Type: AmsServer
The ams server.

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.1.2 AdsServer.IsConnected Property

Gets a value indicating whether [AdsServer](#) [▶ 1375] is connected.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsConnected { get; }
```

Property Value

Type: [Boolean](#)
true if this instance is connected; otherwise, false.

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.1.3 AdsServer.IsDisconnecting Property

Indicates, that the [AdsServer](#) [▶ 1375] is actually disconnecting.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected bool IsDisconnecting { get; }
```

Property Value

Type: [Boolean](#)

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.1.4 **AdsServer.IsDisposed Property**

Gets a value indicating whether this instance is disposed.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsDisposed { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is disposed; otherwise, false.

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.1.5 **AdsServer.Logger Property**

Gets the logger object.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ILogger? Logger { get; }
```

Property Value

Type: [ILogger](#)

The logger.

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.1.6 AdsServer.ServerAddress Property

The AMS address of this server.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public virtual AmsAddress? ServerAddress { get; }
```

Property Value

Type: [AmsAddress](#) [► 752]

Remarks

The Address consists of [AmsNetId](#) [► 767] and [AmsPort](#) [► 795]. While the [AmsNetId](#) [► 767] is defined by the System, where the **AmsTcplpRouter** is running, the [AmsPort](#) [► 795] is specified by the [AdsServer](#) [► 1375] constructor. By default, the router is running on the same system, but can be configured by [RouterEndPoint](#) [► 2006].

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

[TwinCAT.Ads.AmsNetId](#) [► 767]

[TwinCAT.Ads.AmsPort](#) [► 795]

[AmsConfiguration.RouterEndPoint](#) [► 2006]

6.4.1.1.7 AdsServer.ServerName Property

Gets the name of the server.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string ServerName { get; }
```

Property Value

Type: [String](#)

The name of the server.

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.1.8 AdsServer.ServerPort Property

Gets the server port.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ushort ServerPort { get; }
```

Property Value

Type: [UInt16](#)

The server port.

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.1.9 AdsServer.ServerVersion Property

Gets the Version of the Server.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Version ServerVersion { get; }
```

Property Value

Type: [Version](#)

The version.

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.1.10 AdsServer.IsDynamicPort Property

Gets a value indicating this [AdsServer](#) [► 1375] has a dynamic/unfixed port.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsDynamicPort { get; }
```

Property Value

Type: [Boolean](#)

true if dynamic/unfixed port; otherwise, false.

Remarks

In general all Clients have dynamic ports and inherit from AdsServer.

Reference









[AdsServer Class](#) [► 1375]




















[TwinCAT.Ads.Server Namespace](#) [► 1374]















6.4.1.2 AdsServer Methods






The [AdsServer](#) [► 1375] type exposes the following members.

Methods

	Name	Description
	AddDeviceNotificationRequest [► 1393]	Sends an ADS Add Device Notification request (synchronous).
	AddDeviceNotificationRequestAsync [► 1394]	Sends an ADS Add Device Notification request (async)
	AddDeviceNotificationResponseAsync [► 1395]	Sends an ADS Add Device Notification response.
	ConnectServer [► 1396]	Connect this ADS server to the local ADS router.
	ConnectServerAndWaitAsync [► 1396]	Registers the AdsServer [► 1375] at the router asynchronously.
	DeleteDeviceNotificationRequest [► 1398]	Sends an ADS Delete Device Notification request (synchronous).
	DeleteDeviceNotificationRequestAsync [► 1399]	Sends an ADS Delete Device Notification request (async).
	DeleteDeviceNotificationResponseAsync [► 1400]	Sends an ADS Delete Device Notification response.

	Name	Description
	DeviceNotificationRequestAsync [▶ 1400]	Sends an ADS Device Notification request asynchronously
	DeviceNotificationRequestSync [▶ 1401]	Sends an ADS Device Notification request (sync)
	Disconnect [▶ 1402]	Disconnects this ADS server from the local ADS router.
	Dispose [▶ 1403]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.
	Dispose(Boolean) [▶ 1403]	Releases unmanaged and - optionally - managed resources.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize [▶ 1403]	Finalizes an instance of the AdsServer [▶ 1375] class. (Overrides Object.Finalize .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetServerName [▶ 1441]	Gets the name of the server.
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnAddDeviceNotificationAsync [▶ 1429]	Called when an ADS Add Device Notification indication is received.
	OnAddDeviceNotificationConfirmationAsync [▶ 1404]	Called when an ADS Add Device Notification confirmation is received.
	OnBeforeConnected [▶ 1430]	Handler function that is called, when the AdsServer [▶ 1375] is connected, but before calling OnConnected . [▶ 1405].
	OnConnected [▶ 1405]	Handler function that is called, when the AdsServer [▶ 1375] is connected.
	OnDeleteDeviceNotificationAsync [▶ 1430]	Called when an ADS Delete Device Notification indication is received.
	OnDeleteDeviceNotificationConfirmationAsync [▶ 1431]	Called when an ADS Delete Device Notification confirmation is received.
	OnDisconnect [▶ 1405]	Called when the AdsServer [▶ 1375] is about to be disconnected.
	OnReadAsync [▶ 1432]	Called when an ADS Read indication is received.
	OnReadConfirmationAsync [▶ 1433]	Called when an ADS Read confirmation is received.

	Name	Description
	OnReadDeviceInfoConfirmationAsync [▶ 1434]	Called when an ADS Read Device Info confirmation is received.
	OnReadDeviceStateAsync [▶ 1435]	Called when an ADS Read State indication is received.
	OnReadDeviceStateConfirmationAsync [▶ 1435]	Called when an ADS Read State confirmation is received.
	OnReadWriteAsync [▶ 1436]	Called when an ADS Read Write indication is received.
	OnReadWriteConfirmationAsync [▶ 1437]	Called when an ADS Read Write confirmation is received.
	OnRouterNotification [▶ 1405]	Handler Function for a Router Notification.
	OnServerConnectionStateChanged [▶ 1406]	Handles the ServerConnectionStateChanged [▶ 1442] event.
	OnWriteAsync [▶ 1438]	Called when an ADS Write indication is received.
	OnWriteConfirmationAsync [▶ 1439]	Called when an ADS Write confirmation is received.
	OnWriteControlAsync [▶ 1440]	Called when an ADS Write Control indication is received.
	OnWriteControlConfirmationAsync [▶ 1441]	Called when an ADS Write Control confirmation is received.
	ReadDeviceInfoRequestAsync [▶ 1409]	Sends an ADS Read Device Info request asynchronously
	ReadDeviceInfoRequestSync [▶ 1409]	Sends an ADS Read Device Info request synchronously.
	ReadDeviceInfoResponseAsync [▶ 1410]	Sends an ADS Read Device Info response.
	ReadDeviceStateRequestAsync [▶ 1412]	Sends an ADS Read State request (asynchronous)
	ReadDeviceStateRequestSync [▶ 1413]	Sends an ADS Read State request (synchronous)

	Name	Description
	ReadDeviceStateResponseAsync [▶ 1413]	Sends an ADS Read State response.
	ReadRequest [▶ 1415]	Sends an ADS Read Request.
	ReadRequestAsync [▶ 1416]	Sends an ADS Read Request asynchronously.
	ReadResponseAsync [▶ 1416]	Sends an ADS Read response.
	ReadWriteRequestAsync [▶ 1419]	Sends an ADS Read Write request.
	ReadWriteRequestSync [▶ 1419]	Sends an ADS Read Write request synchronously
	ReadWriteResponseAsync [▶ 1420]	Sends an ADS Read Write Response.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	WriteControlRequest [▶ 1423]	Sends an ADS Write Control request (synchronous)
	WriteControlRequestAsync [▶ 1424]	Sends an ADS Write Control request (asynchronous).
	WriteControlRequestSync [▶ 1425]	Sends an ADS Write Control request (synchronous).
	WriteControlResponseAsync [▶ 1425]	Sends an ADS Write Control response.
	WriteRequest [▶ 1427]	Sends an ADS Write request synchronously.
	WriteRequestAsync [▶ 1428]	Sends an ADS Write request asynchronously.
	WriteResponseAsync [▶ 1428]	Sends an ADS Write response.

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.2.1 `AdsServer.AddDeviceNotificationConfirmationAsync` Method

Called when an ADS Add Device Notification confirmation is received.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
protected virtual Task<AdsErrorCode> AddDeviceNotificationConfirmationAsync(
    AmsAddress sender,
    uint invokeId,
    AdsErrorCode result,
    uint notificationHandle,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [▶ 752] The sender's AMS address
invokeId	Type: System.UInt32 The invoke id provided by this server during the corresponding request
result	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The ADS error code provided by the sender
notificationHandle	Type: System.UInt32 The notification handle provided by the sender
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [[▶ 664](#)].

A task that represents the asynchronous `AddDeviceNotificationConfirmationAsync(AmsAddress, UInt32, AdsErrorCode, UInt32, CancellationToken)` operation. The `Task.TResult` parameter contains the [AdsErrorCode](#) [[▶ 664](#)] as `Result`.

Remarks

Overwrite this method in derived classes to react on ADS Add Device Notification confirmations.

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.2.2 **AdsServer.AddDeviceNotificationIndicationAsync Method**

Called when an ADS Add Device Notification indication is received.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
protected virtual Task<AdsErrorCode> AddDeviceNotificationIndicationAsync(
    AmsAddress sender,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
```



```

    int dataLength,
    NotificationSettings settings,
    CancellationToken cancel
)

```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752] The sender's AMS address
invokeld	Type: System.UInt32 The invokeld provided by the sender
indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index offset of the requested ADS service
dataLength	Type: System.Int32 Number of bytes to be transmitted
settings	Type: TwinCAT.Ads.NotificationSettings [► 1106] The Notification settings.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [► 664].

A task that represents the asynchronous 'AddDeviceNotificationIndication' operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [► 664] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Add Device Notification indications. The default implementation replies with an ADS ServiceNotSupported error code (0x701).

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.3 **AdsServer.AddDeviceNotificationRequest Method**

Sends an ADS Add Device Notification request (synchronous).

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```

protected AdsErrorCode AddDeviceNotificationRequest(
    AmsAddress target,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    int dataLength,
    NotificationSettings settings
)

```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [► 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invokeld for this call. Used to find the matching ADS Confirmation
indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index group of the requested ADS service
dataLength	Type: System.Int32 The number of bytes to be transmitted
settings	Type: TwinCAT.Ads.NotificationSettings [► 1106] The notification settings.

Return Value

Type: [AdsErrorCode](#) [► 664]
The ADS error code for this call.

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.4 AdsServer.AddDeviceNotificationRequestAsync Method

Sends an ADS Add Device Notification request (async)

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: [TwinCAT.Ads.Server](#) (in [TwinCAT.Ads.Server.dll](#)) Version: 6.0.328+39e3229

Syntax**C#**

```
protected Task<AdsErrorCode> AddDeviceNotificationRequestAsync (
    AmsAddress target,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    int dataLength,
    NotificationSettings settings,
    CancellationToken cancel
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [► 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invokeld for this call. Used to find the matching ADS Confirmation
indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index group of the requested ADS service
dataLength	Type: System.Int32 The number of bytes to be transmitted

settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The notification settings.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [▶ 664].

A task that represents the asynchronous `AddDeviceNotificationRequestAsync(AmsAddress, UInt32, UInt32, UInt32, Int32, NotificationSettings, CancellationToken)` operation. The [Task.TResult](#). parameter contains the [AdsErrorCode](#) [▶ 664] as [Result](#).

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.2.5 AdsServer.AddDeviceNotificationResponseAsync Method

Sends an ADS Add Device Notification response.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: `TwinCAT.Ads.Server` (in `TwinCAT.Ads.Server.dll`) Version: 6.0.328+39e3229

Syntax

C#

```
protected Task<AdsErrorCode> AddDeviceNotificationResponseAsync(
    AmsAddress target,
    uint invokeId,
    AdsErrorCode result,
    uint handle,
    CancellationToken cancel
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invoke ID provided by the receiver
result	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The ADS error code for the response
handle	Type: System.UInt32 The notification handle for the added notification
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [▶ 664].

A task that represents the asynchronous `AddDeviceNotificationResponseAsync(AmsAddress, UInt32, AdsErrorCode, UInt32, CancellationToken)` operation. The [Task.TResult](#). parameter contains the [AdsErrorCode](#) [▶ 664] as [Result](#).

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.1.2.6 **AdsServer.ConnectServer Method**

Connect this ADS server to the local ADS router.

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public virtual uint ConnectServer()
```

Return Value

Type: [UInt32](#)

The AmsServer Port.

Exceptions

Exception	Condition
AdsServerException [► 1443]	The connect call has failed!.

Reference

[AdsServer Class \[► 1375\]](#)

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.1.2.7 **AdsServer.ConnectServerAndWaitAsync Method**

Registers the [AdsServer \[► 1375\]](#) at the router asynchronously.

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<AdsErrorCode> ConnectServerAndWaitAsync (
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken
--------	--

Return Value

Type: [Task.AdsErrorCode \[► 664\]](#).

Returns a task object that represents the ConnectServerAndWaitAsync(CancellationToken) operation which returns an [AdsErrorCode \[► 664\]](#)" as result..

Remarks

The connection is hold until a cancel is requested, that means this method will wait until disconnect.

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.8 AdsServer.DeleteDeviceNotificationConfirmationAsync Method

Called when an ADS Delete Device Notification confirmation is received.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
protected virtual Task<AdsErrorCode> DeleteDeviceNotificationConfirmationAsync(  
    AmsAddress sender,  
    uint invokeId,  
    AdsErrorCode result,  
    CancellationToken cancel  
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752] The sender's AMS address
invokeId	Type: System.UInt32 The invoke id provided by this server during the corresponding request
result	Type: TwinCAT.Ads.AdsErrorCode [► 664] The ADS error code provided by the sender
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [► 664].

A task that represents the asynchronous `DeleteDeviceNotificationConfirmationAsync(AmsAddress, UInt32, AdsErrorCode, CancellationToken)` operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [► 664] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Delete Device Notification confirmations.

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.9 AdsServer.DeleteDeviceNotificationIndicationAsync Method

Called when an ADS Delete Device Notification indication is received.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
protected virtual Task<AdsErrorCode> DeleteDeviceNotificationIndicationAsync(
    AmsAddress sender,
    uint invokeId,
    uint hNotification,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752] The sender's AMS address
invokeld	Type: System.UInt32 The invokeld provided by the sender
hNotification	Type: System.UInt32 The notification handle to be deleted
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [► 664].

A task that represents the asynchronous `DeleteDeviceNotificationIndicationAsync(AmsAddress, UInt32, UInt32, CancellationToken)` operation. The `Task.TResult` parameter contains the [AdsErrorCode](#) [► 664] as `Result`.

Remarks

Overwrite this method in derived classes to react on ADS Delete Device Notification indications. The default implementation replies with an ADS ServiceNotSupported error code (0x701).

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.10 AdsServer.DeleteDeviceNotificationRequest Method

Sends an ADS Delete Device Notification request (synchronous).

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected AdsErrorCode DeleteDeviceNotificationRequest(
    AmsAddress target,
    uint invokeId,
    uint hNotification
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [► 752] The receiver's AMS address
--------	--

invokeld	Type: System.UInt32 The invokeld for this call. Used to find the matching ADS Confirmation
hNotification	Type: System.UInt32 The notification ID to be deleted

Return Value

Type: [AdsErrorCode](#) [▶ 664]
The ADS error code for this call.

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.2.11 **AdsServer.DeleteDeviceNotificationRequestAsync Method**

Sends an ADS Delete Device Notification request (async).

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: [TwinCAT.Ads.Server](#) (in [TwinCAT.Ads.Server.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
protected Task<AdsErrorCode> DeleteDeviceNotificationRequestAsync(
    AmsAddress target,
    uint invokeId,
    uint hNotification,
    CancellationToken cancel
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invokeld for this call. Used to find the matching ADS Confirmation
hNotification	Type: System.UInt32 The notification ID to be deleted
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [▶ 664].
A task that represents the asynchronous [DeleteDeviceNotificationRequestAsync\(AmsAddress, UInt32, UInt32, CancellationToken\)](#) operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [▶ 664] as [Result](#).

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.2.12 **AdsServer.DeleteDeviceNotificationResponseAsync** Method

Sends an ADS Delete Device Notification response.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected Task<AdsErrorCode> DeleteDeviceNotificationResponseAsync(
    AmsAddress target,
    uint invokeId,
    AdsErrorCode result,
    CancellationToken cancel
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invoke ID provided by the receiver
result	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The ADS error code for the response
cancel	Type: System.Threading.CancellationToken The cancellation token

Return Value

Type: [Task.AdsErrorCode](#) [[▶ 664](#)].

A task that represents the asynchronous DeleteDeviceNotificationResponseAsync(AmsAddress, UInt32, AdsErrorCode, CancellationToken) operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [[▶ 664](#)] as [Result](#).

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.2.13 **AdsServer.DeviceNotificationRequestAsync** Method

Sends an ADS Device Notification request asynchronously

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected Task<AdsErrorCode> DeviceNotificationRequestAsync(
    AmsAddress target,
    uint invokeId,
    uint numStampHeaders,
    NotificationSamplesStamp[] notificationHeaders,
    CancellationToken cancel
)
```


Parameters

target	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invokeld for this call. Used to find the matching ADS Confirmation
numStampHeaders	Type: System.UInt32 The number of ADS Stamp Headers to be sent
notificationHeaders	Type: .TwinCAT.Ads.Server.NotificationSamplesStamp [▶ 1456]. The array of ADS Stamp Headers to be sent
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [▶ 664].
The ADS error code for this call.

Exceptions

Exception	Condition
ArgumentNullException	target
ArgumentNullException	notificationHeaders

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.2.14 AdsServer.DeviceNotificationRequestSync Method

Sends an ADS Device Notification request (sync)

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected AdsErrorCode DeviceNotificationRequestSync (
    AmsAddress target,
    uint invokeId,
    uint numStampHeaders,
    NotificationSamplesStamp[] notificationHeaders
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invokeld for this call. Used to find the matching ADS Confirmation
numStampHeaders	Type: System.UInt32 The number of ADS Stamp Headers to be sent
notificationHeaders	Type: .TwinCAT.Ads.Server.NotificationSamplesStamp [▶ 1456]. The array of ADS Stamp Headers to be sent

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
 The ADS error code for this call.

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.2.15 AdsServer.Disconnect Method

Disconnects this ADS server from the local ADS router.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public virtual bool Disconnect()
```

Return Value

Type: [Boolean](#)
 true if disconnected, false if the [AdsServer](#) [[▶ 1375](#)] was disconnected before.

Exceptions



Exception	Condition
AdsServerException [▶ 1443]	Thrown if the disconnect call fails.

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.2.16 AdsServer.Dispose Method**Overload List**

	Name	Description
	Dispose. [▶ 1403]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.
	Dispose(Boolean) [▶ 1403]	Releases unmanaged and - optionally - managed resources.

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.2.16.1 AdsServer.Dispose Method

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Dispose()
```

Implements

[IDisposable.Dispose.](#)

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[Dispose Overload](#) [[▶ 1402](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.2.16.2 AdsServer.Dispose Method (Boolean)

Releases unmanaged and - optionally - managed resources.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual void Dispose(  
    bool disposing  
)
```

Parameters

disposing	Type: System.Boolean true to release both managed and unmanaged resources; false to release only unmanaged resources.
-----------	--

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[Dispose Overload](#) [[▶ 1402](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.2.17 AdsServer.Finalize Method

Finalizes an instance of the [AdsServer](#) [[▶ 1375](#)] class.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected override void Finalize()
```

Implements

[Object.Finalize.](#)

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.18 AdsServer.OnAddDeviceNotificationConfirmationAsync Method

Called when an ADS Add Device Notification confirmation is received.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<AdsErrorCode> OnAddDeviceNotificationConfirmationAsync (
    AmsAddress sender,
    uint invokeId,
    AdsErrorCode result,
    uint notificationHandle,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752] The sender's AMS address
invokeld	Type: System.UInt32 The invoke id provided by this server during the corresponding request
result	Type: TwinCAT.Ads.AdsErrorCode [► 664] The ADS error code provided by the sender
notificationHandle	Type: System.UInt32 The notification handle provided by the sender
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [► 664].

A task that represents the asynchronous `OnAddDeviceNotificationConfirmationAsync(AmsAddress, UInt32, AdsErrorCode, UInt32, CancellationToken)` operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [► 664] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Add Device Notification confirmations. The Default implementaton just returns [NoError](#) [► 664]. Because the Confirmation is the last step in the ADS communication an Error will just be logged, but not processed any further. For a functional [AdsServer](#) [► 1375], this confirmation handler doesn't need to be overloaded!

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.19 AdsServer.OnConnected Method

Handler function that is called, when the [AdsServer](#) [► 1375] is connected.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual void OnConnected()
```

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.20 AdsServer.OnDisconnect Method

Called when the [AdsServer](#) [► 1375] is about to be disconnected.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual bool OnDisconnect()
```

Return Value

Type: [Boolean](#)

true if the [AdsServer](#) [► 1375] is disconnected, false otherwise (was disconnected before)

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.21 AdsServer.OnRouterNotification Method

Handler Function for a Router Notification.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public virtual void OnRouterNotification(  
    AmsRouterState state  
)
```

Parameters

state	Type: TwinCAT.Ads.AmsRouterState [▶ 800] The route state.
-------	--

Exceptions

Exception	Condition
NotImplementedException	

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.2.22 AdsServer.OnServerConnectionStateChanged Method

Handles the [ServerConnectionStateChanged](#) [[▶ 1442](#)] event.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
protected virtual void OnServerConnectionStateChanged(
    Object? sender,
    ServerConnectionStateChangedEventArgs e
)
```

Parameters

sender	Type: System.Object The sender.
e	Type: TwinCAT.Ads.Server.ServerConnectionStateChangedEventArgs [▶ 1462] The ServerConnectionStateChangedEventArgs [▶ 1462] instance containing the event data.

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.2.23 AdsServer.ReadConfirmationAsync Method

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934 addedca3e72bc0ea15da1c14

Syntax**C#**

```
protected virtual Task<AdsErrorCode> ReadConfirmationAsync(
    AmsAddress targetAddress,
    uint invokeId,
    AdsErrorCode result,
```

```
ReadOnlyMemory readData,
void cancel
)
```

Parameters

targetAddress	Type: TwinCAT.Ads.AmsAddress [► 752]
invokeld	Type: System.UInt32
result	Type: TwinCAT.Ads.AdsErrorCode [► 664]
readData	Type: ReadOnlyMemory
cancel	Type: System.Void

Return Value

Type: [Task.AdsErrorCode](#) [► 664].

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.24 AdsServer.ReadDeviceInfoConfirmationAsync Method

Called when an ADS Read Device Info confirmation is received.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934 addedca3e72bc0ea15da1c14

Syntax

C#

```
protected virtual Task<AdsErrorCode> ReadDeviceInfoConfirmationAsync(
    AmsAddress sender,
    uint invokeId,
    AdsErrorCode result,
    string name,
    AdsVersion version,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752] The sender's AMS address
invokeld	Type: System.UInt32 The invoke id provided by this server during the corresponding request
result	Type: TwinCAT.Ads.AdsErrorCode [► 664] The ADS error code provided by the sender
name	Type: System.String The sender's name
version	Type: TwinCAT.Ads.AdsVersion [► 743] The sender's version

cancel Type: [System.Threading.CancellationToken](#)
The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [▶ 664].

A task that represents the asynchronous `ReadDeviceInfoConfirmationAsync(AmsAddress, UInt32, AdsErrorCode, String, AdsVersion, CancellationToken)` operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [▶ 664] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Read Device Info confirmations.

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.2.25 **AdsServer.ReadDeviceInfoIndicationAsync Method**

Called when an ADS Read Device Info indication is received.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version:
5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
protected virtual Task<AdsErrorCode> ReadDeviceInfoIndicationAsync(  
    AmsAddress sender,  
    uint invokeId,  
    CancellationToken cancel  
)
```

Parameters

sender Type: [TwinCAT.Ads.AmsAddress](#) [▶ 752]
The sender's / requester's AMS address

invokeId Type: [System.UInt32](#)
The invokeId provided by the sender

cancel Type: [System.Threading.CancellationToken](#)
The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [▶ 664].

A task that represents the asynchronous `OnReadDeviceInfoIndicationAsync(AmsAddress, UInt32, CancellationToken)` operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [▶ 664] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Read Device Info indications. The default implementation replies with an ADS ServiceNotSupported error code (0x701).

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace \[▶ 1374\]](#)

6.4.1.2.26 **AdsServer.ReadDeviceInfoRequestAsync Method**

Sends an ADS Read Device Info request asynchronously

Namespace: [TwinCAT.Ads.Server \[▶ 1374\]](#)

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected Task<AdsErrorCode> ReadDeviceInfoRequestAsync(
    AmsAddress target,
    uint invokeId,
    CancellationToken cancel
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver's AMS address
invokeId	Type: System.UInt32 The invokeId for this call. Used to find the matching ADS Confirmation
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode \[▶ 664\]](#).

A task that represents the asynchronous ReadDeviceInfoRequestAsync(AmsAddress, UInt32, CancellationToken) operation. The [Task.TResult](#) parameter contains the [AdsErrorCode \[▶ 664\]](#) as [Result](#).

Reference

[AdsServer Class \[▶ 1375\]](#)

[TwinCAT.Ads.Server Namespace \[▶ 1374\]](#)

6.4.1.2.27 **AdsServer.ReadDeviceInfoRequestSync Method**

Sends an ADS Read Device Info request synchronously.

Namespace: [TwinCAT.Ads.Server \[▶ 1374\]](#)

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected AdsErrorCode ReadDeviceInfoRequestSync(
    AmsAddress target,
    uint invokeId
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver's AMS address
invokeId	Type: System.UInt32 The invokeId for this call. Used to find the matching ADS Confirmation

Return Value

Type: [AdsErrorCode](#) [► 664]
 The ADS error code for this call.

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.28 AdsServer.ReadDeviceInfoResponseAsync Method

Sends an ADS Read Device Info response.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
protected Task<AdsErrorCode> ReadDeviceInfoResponseAsync (
    AmsAddress target,
    uint invokeId,
    AdsErrorCode result,
    string name,
    AdsVersion? version,
    Cancellation token cancel
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [► 752] The receiver's AMS address
invokeId	Type: System.UInt32 The invoke ID provided by the receiver
result	Type: TwinCAT.Ads.AdsErrorCode [► 664] The ADS error code for the response
name	Type: System.String The name of this ADS server
version	Type: TwinCAT.Ads.AdsVersion [► 743] The version of this ADS server
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [► 664].
 A task that represents the asynchronous `ReadDeviceInfoResponseAsync(AmsAddress, UInt32, AdsErrorCode, String, AdsVersion, CancellationToken)` operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [► 664] as [Result](#).

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.29 **AdsServer.ReadDeviceStateConfirmationAsync Method**

Called when an ADS Read State confirmation is received.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
protected virtual Task<AdsErrorCode> ReadDeviceStateConfirmationAsync(  
    AmsAddress sender,  
    uint invokeId,  
    AdsErrorCode result,  
    AdsState adsState,  
    ushort deviceState,  
    CancellationToken cancel  
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [▶ 752] The sender's AMS address
invokeId	Type: System.UInt32 The invoke id provided by this server during the corresponding request
result	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The ADS error code provided by the sender
adsState	Type: TwinCAT.Ads.AdsState [▶ 729] The ADS state of the sender
deviceState	Type: System.UInt16 The device state of the sender
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [[▶ 664](#)].

A task that represents the asynchronous `ReadDeviceStateConfirmationAsync(AmsAddress, UInt32, AdsErrorCode, AdsState, UInt16, CancellationToken)` operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [[▶ 664](#)] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Read State confirmations.

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.2.30 **AdsServer.ReadDeviceStateIndicationAsync Method**

Called when an ADS Read State indication is received.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
protected virtual Task<AdsErrorCode> ReadDeviceStateIndicationAsync (
    AmsAddress sender,
    uint invokeId,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752] The sender's AMS address
invokeld	Type: System.UInt32 The invokeld provided by the sender
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [► 664].

A task that represents the asynchronous `ReadDeviceStateIndicationAsync(AmsAddress, UInt32, CancellationToken)` operation. The `Task.TResult` parameter contains the [AdsErrorCode](#) [► 664] as `Result`.

Remarks

Overwrite this method in derived classes to react on ADS Read State indications. The default implementation replies with an ADS `ServiceNotSupported` error code (0x701).

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.31 AdsServer.ReadDeviceStateRequestAsync Method

Sends an ADS Read State request (asynchronous)

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: `TwinCAT.Ads.Server` (in `TwinCAT.Ads.Server.dll`) Version: 6.0.328+39e3229

Syntax

C#

```
protected Task<AdsErrorCode> ReadDeviceStateRequestAsync (
    AmsAddress target,
    uint invokeId,
    CancellationToken cancel
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [► 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invokeld for this call. Used to find the matching ADS Confirmation
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [▶ 664].

A task that represents the asynchronous `ReadDeviceStateRequestAsync(AmsAddress, UInt32, CancellationToken)` operation. The `Task.TResult` parameter contains the [AdsErrorCode](#) [▶ 664] as `Result`.

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.2.32 AdsServer.ReadDeviceStateRequestSync Method

Sends an ADS Read State request (synchronous)

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: `TwinCAT.Ads.Server` (in `TwinCAT.Ads.Server.dll`) Version: 6.0.328+39e3229

Syntax**C#**

```
protected AdsErrorCode ReadDeviceStateRequestSync (
    AmsAddress target,
    uint invokeId
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver's AMS address
invokeId	Type: System.UInt32 The invokeId for this call. Used to find the matching ADS Confirmation

Return Value

Type: [AdsErrorCode](#) [▶ 664]

The ADS error code for this call.

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.2.33 AdsServer.ReadDeviceStateResponseAsync Method

Sends an ADS Read State response.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: `TwinCAT.Ads.Server` (in `TwinCAT.Ads.Server.dll`) Version: 6.0.328+39e3229

Syntax**C#**

```
protected Task<AdsErrorCode> ReadDeviceStateResponseAsync (
    AmsAddress target,
    uint invokeId,
    AdsErrorCode result,
    AdsState adsState,
```

```

    ushort deviceState,
    CancellationToken cancel
)

```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invoke ID provided by the receiver
result	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The ADS error code for the response
adsState	Type: TwinCAT.Ads.AdsState [▶ 729] The current ADS state of this ADS server
deviceState	Type: System.UInt16 The device state of this ADS server
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [▶ 664].

A task that represents the asynchronous `ReadDeviceStateResponseAsync(AmsAddress, UInt32, AdsErrorCode, AdsState, UInt16, CancellationToken)` operation. The [Task.TResult](#). parameter contains the [AdsErrorCode](#) [▶ 664] as [Result](#).

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.2.34 AdsServer.ReadIndicationAsync Method

Called when an ADS Read indication is received.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: `TwinCAT.Ads.Server` (in `TwinCAT.Ads.Server.dll`) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```

protected virtual Task<AdsErrorCode> ReadIndicationAsync(
    AmsAddress sender,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    int readLength,
    CancellationToken cancel
)

```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [▶ 752] The sender's AMS address
invokeld	Type: System.UInt32 The invokeld provided by the sender
indexGroup	Type: System.UInt32 The index group of the requested ADS service

indexOffset	Type: System.UInt32 The index offset of the requested ADS service
readLength	Type: System.Int32 The number of bytes to be read
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [▶ 664].
A task that represents the asynchronous ReadIndicationAsync(AmsAddress, UInt32, UInt32, UInt32, Int32, CancellationToken) operation. The [Task.TResult](#). parameter contains the [AdsErrorCode](#) [▶ 664] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Read indications. The default implementation replies with an ADS ServiceNotSupported error code (0x701).

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.2.35 AdsServer.ReadRequest Method

Sends an ADS Read Request.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected AdsErrorCode ReadRequest(
    AmsAddress target,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    int readLength
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invokeld for this call. Used to find the matching ADS Confirmation
indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index group of the requested ADS service
readLength	Type: System.Int32 The number of bytes to be read

Return Value

Type: [AdsErrorCode](#) [▶ 664]
The ADS error code for this call.

Reference[AdsServer Class \[► 1375\]](#)[TwinCAT.Ads.Server Namespace \[► 1374\]](#)**6.4.1.2.36 AdsServer.ReadRequestAsync Method**

Sends an ADS Read Request asynchronously.

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)**Assembly:** TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
protected Task<AdsErrorCode> ReadRequestAsync(
    AmsAddress target,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    int readLength,
    CancellationToken cancel
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [► 752] The receiver's AMS address
invokeId	Type: System.UInt32 The invokeId for this call. Used to find the matching ADS Confirmation
indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index group of the requested ADS service
readLength	Type: System.Int32 The number of bytes to be read
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return ValueType: [Task.AdsErrorCode \[► 664\]](#).

A task that represents the asynchronous ReadRequestAsync(AmsAddress, UInt32, UInt32, UInt32, Int32, CancellationToken) operation. The [Task.TResult](#) parameter contains the [AdsErrorCode \[► 664\]](#) as [Result](#).

Reference[AdsServer Class \[► 1375\]](#)[TwinCAT.Ads.Server Namespace \[► 1374\]](#)**6.4.1.2.37 AdsServer.ReadResponseAsync Method**

Sends an ADS Read response.

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)**Assembly:** TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected Task<AdsErrorCode> ReadResponseAsync (
    AmsAddress target,
    uint invokeId,
    AdsErrorCode result,
    ReadOnlyMemory<byte> data,
    CancellationToken cancel
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invoke ID provided by the receiver
result	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The ADS error code for the response
data	Type: System.ReadOnlyMemory.Byte. The read data buffer
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [▶ 664].

A task that represents the asynchronous `ReadResponseAsync(AmsAddress, UInt32, AdsErrorCode, ReadOnlyMemory.Byte., CancellationToken)` operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [▶ 664] as [Result](#).

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.2.38 AdsServer.ReadWriteConfirmationAsync Method

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: [TwinCAT.Ads.Server](#) (in [TwinCAT.Ads.Server.dll](#)) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
protected virtual Task<AdsErrorCode> ReadWriteConfirmationAsync (
    AmsAddress sender,
    uint invokeId,
    AdsErrorCode result,
    ReadOnlyMemory readData,
    void cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [▶ 752]
invokeld	Type: System.UInt32

result	Type: TwinCAT.Ads.AdsErrorCode [► 664]
readData	Type: ReadOnlyMemory
cancel	Type: System.Void

Return Value

Type: [Task.AdsErrorCode](#) [► 664].

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.39 AdsServer.ReadWriteIndicationAsync Method

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
protected virtual Task<AdsErrorCode> ReadWriteIndicationAsync(  
    AmsAddress sender,  
    uint invokeId,  
    uint indexGroup,  
    uint indexOffset,  
    int readLength,  
    ReadOnlyMemory writeData,  
    void cancel  
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752]
invokeId	Type: System.UInt32
indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
readLength	Type: System.Int32
writeData	Type: ReadOnlyMemory
cancel	Type: System.Void

Return Value

Type: [Task.AdsErrorCode](#) [► 664].

Reference

[AdsServer Class \[▶ 1375\]](#)

[TwinCAT.Ads.Server Namespace \[▶ 1374\]](#)

6.4.1.2.40 AdsServer.ReadWriteRequestAsync Method

Sends an ADS Read Write request.

Namespace: [TwinCAT.Ads.Server \[▶ 1374\]](#)

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected Task<AdsErrorCode> ReadWriteRequestAsync(
    AmsAddress target,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    int readLength,
    ReadOnlyMemory<byte> writeData,
    CancellationToken cancel
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invokeld for this call. Used to find the matching ADS Confirmation
indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index group of the requested ADS service
readLength	Type: System.Int32 The number of bytes to be read
writeData	Type: System.ReadOnlyMemory.Byte . The data to be written
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode \[▶ 664\]](#).

A task that represents the asynchronous 'ReadWriteRequest' operation. The [Task.TResult](#) parameter contains the [AdsErrorCode \[▶ 664\]](#) as [Result](#).

Reference

[AdsServer Class \[▶ 1375\]](#)

[TwinCAT.Ads.Server Namespace \[▶ 1374\]](#)

6.4.1.2.41 AdsServer.ReadWriteRequestSync Method

Sends an ADS Read Write request synchronously

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected AdsErrorCode ReadWriteRequestSync (
    AmsAddress target,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    int readLength,
    ReadOnlySpan<byte> data
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invokeld for this call. Used to find the matching ADS Confirmation
indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index group of the requested ADS service
readLength	Type: System.Int32 The number of bytes to be read
data	Type: System.ReadOnlySpan.Byte . The data to be written

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The ADS error code for this call.

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.2.42 **AdsServer.ReadWriteResponseAsync Method**

Sends an ADS Read Write Response.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected Task<AdsErrorCode> ReadWriteResponseAsync (
    AmsAddress target,
    uint invokeId,
    AdsErrorCode result,
    ReadOnlyMemory<byte> data,
    CancellationToken cancel
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [► 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invoke ID provided by the receiver
result	Type: TwinCAT.Ads.AdsErrorCode [► 664] The ADS error code for the response
data	Type: System.ReadOnlyMemory.Byte. The read data buffer
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [► 664].

A task that represents the asynchronous `ReadWriteResponseAsync(AmsAddress, UInt32, AdsErrorCode, ReadOnlyMemory.Byte., CancellationToken)` operation. The `Task.TResult` parameter contains the [AdsErrorCode](#) [► 664] as `Result`.

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.43 AdsServer.WriteConfirmationAsync Method

Called when an ADS Write confirmation is received. Overwrite this method in derived classes to react on ADS Write confirmations.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax**C#**

```
protected virtual Task<AdsErrorCode> WriteConfirmationAsync(
    AmsAddress sender,
    uint invokeId,
    AdsErrorCode result,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752] The sender's AMS address
invokeld	Type: System.UInt32 The invoke id provided by this server during the corresponding request
result	Type: TwinCAT.Ads.AdsErrorCode [► 664] The ADS error code provided by the sender
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [▶ 664].

A task that represents the asynchronous `WriteConfirmationAsync(AmsAddress, UInt32, AdsErrorCode, CancellationToken)` operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [▶ 664] as [Result](#).

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.2.44 **AdsServer.WriteControlConfirmationAsync Method**

Called when an ADS Write Control confirmation is received.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: `TwinCAT.Ads.Server` (in `TwinCAT.Ads.Server.dll`) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
protected virtual Task<AdsErrorCode> WriteControlConfirmationAsync(  
    AmsAddress sender,  
    uint invokeId,  
    AdsErrorCode result,  
    CancellationToken cancel  
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [▶ 752] The sender's AMS address
invokeId	Type: System.UInt32 The invoke id provided by this server during the corresponding request
result	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The ADS error code provided by the sender
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [▶ 664].

A task that represents the asynchronous `WriteControlConfirmationAsync(AmsAddress, UInt32, AdsErrorCode, CancellationToken)` operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [▶ 664] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Write Control confirmations.

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.2.45 **AdsServer.WriteControlIndicationAsync Method**

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
protected virtual Task<AdsErrorCode> WriteControlIndicationAsync(  
    AmsAddress sender,  
    uint invokeId,  
    AdsState adsState,  
    ushort deviceState,  
    ReadOnlyMemory data,  
    void cancel  
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752]
invokeId	Type: System.UInt32
adsState	Type: TwinCAT.Ads.AdsState [► 729]
deviceState	Type: System.UInt16
data	Type: ReadOnlyMemory
cancel	Type: System.Void

Return Value

Type: [Task.AdsErrorCode](#) [► 664].

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.46 **AdsServer.WriteControlRequest Method**

Sends an ADS Write Control request (synchronous)

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected AdsErrorCode WriteControlRequest(  
    AmsAddress target,  
    uint invokeId,  
    AdsState adsState,  
    ushort deviceState,  
    ReadOnlySpan<byte> data  
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [► 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invokeld for this call. Used to find the matching ADS Confirmation
adsState	Type: TwinCAT.Ads.AdsState [► 729] The requested new ADS state of the receiver.
deviceState	Type: System.UInt16 The requested new ADS state of the receiver.
data	Type: System.ReadOnlySpan.Byte . An additional data buffer for the Control request.

Return Value

Type: [AdsErrorCode](#) [► 664]
The ADS error code for this call.

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.47 AdsServer.WriteControlRequestAsync Method

Sends an ADS Write Control request (asynchronous).

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
protected Task<AdsErrorCode> WriteControlRequestAsync (
    AmsAddress target,
    uint invokeId,
    AdsState adsState,
    ushort deviceState,
    ReadOnlyMemory<byte> data,
    CancellationToken cancel
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [► 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invokeld for this call. Used to find the matching ADS Confirmation
adsState	Type: TwinCAT.Ads.AdsState [► 729] The requested new ADS state of the receiver
deviceState	Type: System.UInt16 The requested new ADS state of the receiver
data	Type: System.ReadOnlyMemory.Byte . The data to be written.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [▶ 664].

A task that represents the asynchronous [WriteControlRequestAsync\(AmsAddress, UInt32, AdsState, UInt16, ReadOnlyMemory.Byte., CancellationToken\)](#) operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [▶ 664] as [Result](#).

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.2.48 AdsServer.WriteControlRequestSync Method

Sends an ADS Write Control request (synchronous).

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected AdsErrorCode WriteControlRequestSync (
    AmsAddress target,
    uint invokeId,
    AdsState adsState,
    ushort deviceState,
    ReadOnlySpan<byte> data
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invokeld for this call. Used to find the matching ADS Confirmation
adsState	Type: TwinCAT.Ads.AdsState [▶ 729] The requested new ADS state of the receiver
deviceState	Type: System.UInt16 The requested new ADS state of the receiver
data	Type: System.ReadOnlySpan.Byte . The data to be written.

Return Value

Type: [AdsErrorCode](#) [▶ 664]

A task that represents the asynchronous [WriteControlRequestAsync\(AmsAddress, UInt32, AdsState, UInt16, ReadOnlyMemory.Byte., CancellationToken\)](#) [▶ 1424] operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [▶ 664] as [Result](#).

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.2.49 AdsServer.WriteControlResponseAsync Method

Sends an ADS Write Control response.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected Task<AdsErrorCode> WriteControlResponseAsync (
    AmsAddress target,
    uint invokeId,
    AdsErrorCode result,
    CancellationToken cancel
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invoke ID provided by the receiver
result	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The ADS error code for the response
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [[▶ 664](#)].

A task that represents the asynchronous `WriteControlResponseAsync(AmsAddress, UInt32, AdsErrorCode, CancellationToken)` operation. The `Task.TResult` parameter contains the [AdsErrorCode](#) [[▶ 664](#)] as `Result`.

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.2.50 AdsServer.WriteIndicationAsync Method

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
protected virtual Task<AdsErrorCode> WriteIndicationAsync (
    AmsAddress target,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory writeData,
    void cancel
)
```

Parameters

target Type: [TwinCAT.Ads.AmsAddress](#) [[▶ 752](#)]

invokeld Type: [System.UInt32](#)

indexGroup	Type: System.UInt32
indexOffset	Type: System.UInt32
writeData	Type: ReadOnlyMemory
cancel	Type: System.Void

Return Value

Type: [Task.AdsErrorCode](#) [▶ 664].

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.2.51 AdsServer.WriteRequest Method

Sends an ADS Write request synchronously.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected AdsErrorCode WriteRequest (
    AmsAddress target,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    ReadOnlySpan<byte> data
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invokeld for this call. Used to find the matching ADS Confirmation
indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index group of the requested ADS service
data	Type: System.ReadOnlySpan.Byte . The data to be written

Return Value

Type: [AdsErrorCode](#) [▶ 664]
The ADS error code for this call.

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.52 **AdsServer.WriteRequestAsync Method**

Sends an ADS Write request asynchronously.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected Task<AdsErrorCode> WriteRequestAsync (
    AmsAddress target,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory<byte> data,
    CancellationToken cancel
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [► 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invokeld for this call. Used to find the matching ADS Confirmation
indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index group of the requested ADS service
data	Type: System.ReadOnlyMemory.Byte . The data to be written
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [► 664].

A task that represents the asynchronous 'WriteRequest' operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [► 664] as [Result](#).

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.53 **AdsServer.WriteResponseAsync Method**

Sends an ADS Write response.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected Task<AdsErrorCode> WriteResponseAsync (
    AmsAddress target,
    uint invokeId,
    AdsErrorCode result,
    CancellationToken cancel
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver's AMS address
invokeld	Type: System.UInt32 The invoke ID provided by the receiver
result	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The ADS error code for the response
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode \[▶ 664\]](#).
A task that represents the asynchronous WriteResponseAsync(AmsAddress, UInt32, AdsErrorCode, CancellationToken) operation. The [Task.TResult](#) parameter contains the [AdsErrorCode \[▶ 664\]](#) as [Result](#).

Reference

[AdsServer Class \[▶ 1375\]](#)

[TwinCAT.Ads.Server Namespace \[▶ 1374\]](#)

6.4.1.2.54 AdsServer.OnAddDeviceNotificationAsync Method

Called when an ADS Add Device Notification indication is received.

Namespace: [TwinCAT.Ads.Server \[▶ 1374\]](#)

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<ResultHandle> OnAddDeviceNotificationAsync (
    AmsAddress sender,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    int dataLength,
    AmsAddress receiver,
    NotificationSettings settings,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [▶ 752] The sender address of the request.
invokeld	Type: System.UInt32 The invoke identifier of the request.

indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index offset of the requested ADS service
dataLength	Type: System.Int32 Number of bytes to be transmitted
receiver	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver of the notifications (the requester of this message)
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultHandle](#) [[▶ 1136](#)].

A task that represents the asynchronous 'AddDeviceNotificationIndication' operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [[▶ 664](#)] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Add Device Notification indications. The default implementation replies with an ADS ServiceNotSupported error code (0x701).

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.2.55 **AdsServer.OnBeforeConnected Method**

Handler function that is called, when the [AdsServer](#) [[▶ 1375](#)] is connected, but before calling [OnConnected](#). [[▶ 1405](#)].

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual void OnBeforeConnected()
```

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.2.56 **AdsServer.OnDeleteDeviceNotificationAsync Method**

Called when an ADS Delete Device Notification indication is received.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<ResultAds> OnDeleteDeviceNotificationAsync(
    AmsAddress sender,
    uint invokeId,
    uint hNotification,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752] The sender address of the request.
invokeld	Type: System.UInt32 The invoke identifier of the request.
hNotification	Type: System.UInt32 The notification handle to be deleted
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds](#) [► 1116].

A task that represents the asynchronous `DeleteDeviceNotificationIndicationAsync(AmsAddress, UInt32, UInt32, CancellationToken)` operation. The `Task.TResult` parameter contains the [AdsErrorCode](#) [► 664] as `Result`.

Remarks

Overwrite this method in derived classes to react on ADS Delete Device Notification indications. The default implementation replies with an ADS ServiceNotSupported error code (0x701).

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.57 AdsServer.OnDeleteDeviceNotificationConfirmationAsync Method

Called when an ADS Delete Device Notification confirmation is received.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<AdsErrorCode> OnDeleteDeviceNotificationConfirmationAsync(
    AmsAddress sender,
    uint invokeId,
    AdsErrorCode result,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752] The sender's AMS address
--------	--

invokeld	Type: System.UInt32 The invoke id provided by this server during the corresponding request
result	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The ADS error code provided by the sender
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [[▶ 664](#)].

A task that represents the asynchronous [OnDeleteDeviceNotificationConfirmationAsync\(AmsAddress, UInt32, AdsErrorCode, CancellationToken\)](#) operation. The [Task.TResult](#). parameter contains the [AdsErrorCode](#) [[▶ 664](#)] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Delete Device Notification confirmations. The Default implementaton just returns [NoError](#) [[▶ 664](#)]. Because the Confirmation is the last step in the ADS communication an Error will just be logged, but not processed any further. For a functional [AdsServer](#) [[▶ 1375](#)], this confirmation handler doesn't need to be overloaded!

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.2.58 AdsServer.OnReadAsync Method

Called when an ADS Read indication is received.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: [TwinCAT.Ads.Server](#) (in [TwinCAT.Ads.Server.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<ResultReadBytes> OnReadAsync (
    AmsAddress sender,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    int readLength,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [▶ 752] The sender address of the request.
invokeld	Type: System.UInt32 The invoke identifier of the request.
indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index offset of the requested ADS service
readLength	Type: System.Int32 The number of bytes to be read

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultReadBytes](#) [▶ 1150].

A task that represents the asynchronous ReadIndicationAsync(AmsAddress, UInt32, UInt32, UInt32, Int32, CancellationToken) operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [▶ 664] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Read indications. The default implementation replies with an ADS ServiceNotSupported error code (0x701).

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.2.59 AdsServer.OnReadConfirmationAsync Method

Called when an ADS Read confirmation is received.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<AdsErrorCode> OnReadConfirmationAsync(
    AmsAddress targetAddress,
    uint invokeId,
    AdsErrorCode result,
    ReadOnlyMemory<byte> readData,
    CancellationToken cancel
)
```

Parameters

targetAddress	Type: TwinCAT.Ads.AmsAddress [▶ 752] The sender's AMS address
invokeld	Type: System.UInt32 The invoke id provided by this server during the corresponding request
result	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The ADS error code provided by the sender
readData	Type: System.ReadOnlyMemory.Byte . The read data buffer
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [▶ 664].

A task that represents the asynchronous OnReadConfirmationAsync(AmsAddress, UInt32, AdsErrorCode, ReadOnlyMemory.Byte., CancellationToken) operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [▶ 664] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Read confirmations. The Default implementaton just returns [NoError \[▶ 664\]](#). Because the Confirmation is the last step in the ADS communication an Error will just be logged, but not processed any further. For a functional [AdsServer \[▶ 1375\]](#), this confirmation handler doesn't need to be overloaded!

Reference

[AdsServer Class \[▶ 1375\]](#)

[TwinCAT.Ads.Server Namespace \[▶ 1374\]](#)

6.4.1.2.60 AdsServer.OnReadDeviceInfoConfirmationAsync Method

Called when an ADS Read Device Info confirmation is received.

Namespace: [TwinCAT.Ads.Server \[▶ 1374\]](#)

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<AdsErrorCode> OnReadDeviceInfoConfirmationAsync(
    AmsAddress sender,
    uint invokeId,
    AdsErrorCode result,
    string name,
    AdsVersion version,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [▶ 752] The sender's AMS address
invokeld	Type: System.UInt32 The invoke id provided by this server during the corresponding request
result	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The ADS error code provided by the sender
name	Type: System.String The sender's name
version	Type: TwinCAT.Ads.AdsVersion [▶ 743] The sender's version
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode \[▶ 664\]](#).

A task that represents the asynchronous `OnReadDeviceInfoConfirmationAsync(AmsAddress, UInt32, AdsErrorCode, String, AdsVersion, CancellationToken)` operation. The [Task.TResult](#) parameter contains the [AdsErrorCode \[▶ 664\]](#) as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Read Device Info confirmations. The Default implementaton just returns [NoError \[▶ 664\]](#). Because the Confirmation is the last step in the ADS communication an Error will just be logged, but not processed any further. For a functional [AdsServer \[▶ 1375\]](#), this confirmation handler doesn't need to be overloaded!

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.61 **AdsServer.OnReadDeviceStateAsync Method**

Called when an ADS Read State indication is received.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<ResultReadDeviceState> OnReadDeviceStateAsync (
    AmsAddress sender,
    uint invokeId,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752] The sender address of the request.
invokeld	Type: System.UInt32 The invoke identifier of the request.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultReadDeviceState](#) [► 1157].

A task that represents the asynchronous [ReadDeviceStateIndicationAsync\(AmsAddress, UInt32, CancellationToken\)](#) operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [► 664] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Read State indications. The default implementation replies with an ADS ServiceNotSupported error code (0x701).

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.62 **AdsServer.OnReadDeviceStateConfirmationAsync Method**

Called when an ADS Read State confirmation is received.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<AdsErrorCode> OnReadDeviceStateConfirmationAsync (
    AmsAddress sender,
    uint invokeId,
```

```

    AdsErrorCode result,
    AdsState adsState,
    ushort deviceState,
    CancellationToken cancel
)

```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752] The sender's AMS address
invokeld	Type: System.UInt32 The invoke id provided by this server during the corresponding request
result	Type: TwinCAT.Ads.AdsErrorCode [► 664] The ADS error code provided by the sender
adsState	Type: TwinCAT.Ads.AdsState [► 729] The ADS state of the sender
deviceState	Type: System.UInt16 The device state of the sender
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [► 664].

A task that represents the asynchronous `OnReadDeviceStateConfirmationAsync(AmsAddress, UInt32, AdsErrorCode, AdsState, UInt16, CancellationToken)` operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [► 664] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Read State confirmations. The Default implementaton just returns [NoError](#) [► 664]. Because the Confirmation is the last step in the ADS communication an Error will just be logged, but not processed any further. For a functional [AdsServer](#) [► 1375], this confirmation handler doesn't need to be overloaded!

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.63 AdsServer.OnReadWriteAsync Method

Called when an ADS Read Write indication is received.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: [TwinCAT.Ads.Server](#) (in [TwinCAT.Ads.Server.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```

protected virtual Task<ResultReadWriteBytes> OnReadWriteAsync(
    AmsAddress sender,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    int readLength,
    ReadOnlyMemory<byte> writeData,
    CancellationToken cancel
)

```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752] The sender address of the request.
invokeld	Type: System.UInt32 The invoke identifier of the request.
indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index offset of the requested ADS service
readLength	Type: System.Int32 Number of bytes to be read
writeData	Type: System.ReadOnlyMemory.Byte . The data to be written
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultReadWriteBytes](#) [► 1170].

A task that represents the asynchronous 'ReadWriteIndication' operation. The [Task.TResult](#). parameter contains the [AdsErrorCode](#) [► 664] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Read Write indications. The default implementation replies with an ADS ServiceNotSupported error code (0x701).

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.2.64 [AdsServer.OnReadWriteConfirmationAsync](#) Method

Called when an ADS Read Write confirmation is received.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: [TwinCAT.Ads.Server](#) (in [TwinCAT.Ads.Server.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<AdsErrorCode> OnReadWriteConfirmationAsync(
    AmsAddress sender,
    uint invokeId,
    AdsErrorCode result,
    ReadOnlyMemory<byte> readData,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752] The sender's AMS address
invokeld	Type: System.UInt32 The invoke id provided by this server during the corresponding request

result	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The ADS error code provided by the sender
readData	Type: System.ReadOnlyMemory.Byte . The read data buffer
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [▶ 664].

A task that represents the asynchronous `OnReadWriteConfirmationAsync(AmsAddress, UInt32, AdsErrorCode, ReadOnlyMemory.Byte., CancellationToken)` operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [▶ 664] as `Result`.

Remarks

Overwrite this method in derived classes to react on ADS Read Write confirmations. The Default implementaton just returns [NoError](#) [▶ 664]. Because the Confirmation is the last step in the ADS communication an Error will just be logged, but not processed any further. For a functional [AdsServer](#) [▶ 1375], this confirmation handler doesn't need to be overloaded!

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.2.65 AdsServer.OnWriteAsync Method

Called when an ADS Write indication is received.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: [TwinCAT.Ads.Server](#) (in [TwinCAT.Ads.Server.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<ResultWrite> OnWriteAsync(
    AmsAddress target,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory<byte> writeData,
    CancellationToken cancel
)
```

Parameters

target	Type: TwinCAT.Ads.AmsAddress [▶ 752] The sender address of the request.
invokeld	Type: System.UInt32 The invoke identifier of the request.
indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index offset of the requested ADS service
writeData	Type: System.ReadOnlyMemory.Byte . The data to be written

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultWrite](#) [[▶ 1187](#)].

A task that represents the asynchronous WriteIndicationAsync(AmsAddress, UInt32, UInt32, UInt32, ReadOnlyMemory.Byte., CancellationToken) operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [[▶ 664](#)] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Write indications. The default implementation replies with an ADS ServiceNotSupported error code (0x701).

Reference

[AdsServer Class](#) [[▶ 1375](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.1.2.66 AdsServer.OnWriteConfirmationAsync Method

Called when an ADS Write confirmation is received.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<AdsErrorCode> OnWriteConfirmationAsync(
    AmsAddress sender,
    uint invokeId,
    AdsErrorCode result,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [▶ 752] The sender's AMS address
invokeld	Type: System.UInt32 The invoke id provided by this server during the corresponding request
result	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The ADS error code provided by the sender
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode](#) [[▶ 664](#)].

A task that represents the asynchronous OnWriteConfirmationAsync(AmsAddress, UInt32, AdsErrorCode, CancellationToken) operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [[▶ 664](#)] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Write confirmations. The Default implementaton just returns [NoError \[▶ 664\]](#). Because the Confirmation is the last step in the ADS communication an Error will just be logged, but not processed any further. For a functional [AdsServer \[▶ 1375\]](#), this confirmation handler doesn't need to be overloaded!

Reference

[AdsServer Class \[▶ 1375\]](#)

[TwinCAT.Ads.Server Namespace \[▶ 1374\]](#)

6.4.1.2.67 AdsServer.OnWriteControlAsync Method

Called when an ADS Write Control indication is received.

Namespace: [TwinCAT.Ads.Server \[▶ 1374\]](#)

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
protected virtual Task<ResultAds> OnWriteControlAsync(
    AmsAddress sender,
    uint invokeId,
    AdsState adsState,
    ushort deviceState,
    ReadOnlyMemory<byte> data,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [▶ 752] The sender address of the request.
invokeld	Type: System.UInt32 The invoke identifier of the request.
adsState	Type: TwinCAT.Ads.AdsState [▶ 729] The requested new ADS state of this ADS device
deviceState	Type: System.UInt16 The requested new device state of this ADS device
data	Type: System.ReadOnlyMemory.Byte. Additional data to write.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultAds \[▶ 1116\]](#).

A task that represents the asynchronous [WriteControlIndicationAsync\(AmsAddress, UInt32, AdsState, UInt16, ReadOnlyMemory.Byte., CancellationToken\)](#) operation. The [Task.TResult.](#) parameter contains the [AdsErrorCode \[▶ 664\]](#) as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Write Control indications. The default implementation replies with an ADS ServiceNotSupported error code (0x701).

Reference[AdsServer Class \[► 1375\]](#)[TwinCAT.Ads.Server Namespace \[► 1374\]](#)**6.4.1.2.68 AdsServer.OnWriteControlConfirmationAsync Method**

Called when an ADS Write Control confirmation is received.

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
protected virtual Task<AdsErrorCode> OnWriteControlConfirmationAsync(
    AmsAddress sender,
    uint invokeId,
    AdsErrorCode result,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752] The sender's AMS address
invokeld	Type: System.UInt32 The invoke id provided by this server during the corresponding request
result	Type: TwinCAT.Ads.AdsErrorCode [► 664] The ADS error code provided by the sender
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.AdsErrorCode \[► 664\]](#).

A task that represents the asynchronous `OnWriteControlConfirmationAsync(AmsAddress, UInt32, AdsErrorCode, CancellationToken)` operation. The `Task.TResult` parameter contains the `AdsErrorCode` [► 664] as `Result`.

Remarks

Overwrite this method in derived classes to react on ADS Write Control confirmations. The Default implementaton just returns `NoError` [► 664]. Because the Confirmation is the last step in the ADS communication an Error will just be logged, but not processed any further. For a functional [AdsServer](#) [► 1375], this confirmation handler doesn't need to be overloaded!

Reference[AdsServer Class \[► 1375\]](#)[TwinCAT.Ads.Server Namespace \[► 1374\]](#)**6.4.1.2.69 AdsServer.GetServerName Method**

Gets the name of the server.

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual string GetServerName()
```

Return Value

Type: [String](#)
The ServerName.

Remarks

Because the ServerName is used as DeviceName in the DeviceInfoStruct, this string must be shorter or equal than 16 bytes in ANSI encoding. The default implementation is '{AmsServer.PortName} ({amsServer.Port})' or for longer PortNames: 'AdsServer_{amsServer.Port}'

Reference


[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.3 AdsServer Events

The [AdsServer](#) [► 1375] type exposes the following members.

Events

	Name	Description
	ServerConnectionState StateChanged [► 1442]	The connection status has changed

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.3.1 AdsServer.ServerConnectionStateChanged Event

The connection status has changed

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: [TwinCAT.Ads.Server](#) (in [TwinCAT.Ads.Server.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<ServerConnectionStateChangedEventArgs> ServerConnectionStateChanged
```

Value

Type: [System.EventHandler.ServerConnectionStateChangedEventArgs](#) [► 1462].

Reference

[AdsServer Class](#) [► 1375]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.1.4 AdsServer Fields

The [AdsServer](#) [▶ 1375] type exposes the following members.

Fields

	Name	Description
	serverVersion [▶ 1443]	The version of the AdsServer

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.1.4.1 AdsServer.serverVersion Field

The version of the AdsServer

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected Version serverVersion
```

Field Value

Type: [Version](#)

Reference

[AdsServer Class](#) [▶ 1375]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.2 AdsServerException Class

An AdsServerException is thrown on communication errors in the [AdsServer](#) [▶ 1375] class.

Inheritance Hierarchy

[System.Object](#)

[System.Exception](#)

[TwinCAT.Ads.Server.AdsServerException](#)

[TwinCAT.Ads.Server.LoopbackNotConnectedException](#) [▶ 1523]

[TwinCAT.Ads.Server.LoopbackNotRegisteredException](#) [▶ 1450]

[TwinCAT.Ads.Server.ServerNotConnectedException](#) [▶ 1465]

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229


Syntax

C#










```
[SerializableAttribute]  
public class AdsServerException : Exception
```

The AdsServerException type exposes the following members.









Constructors

	Name	Description
	AdsServerException [▶ 1445]	Initializes a new instance of the AdsServerException class.


Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	ErrorCode [▶ 1446]	Gets or sets the error code.
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData [▶ 1447]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Overrides Exception.GetObjectData(SerializationInfo, StreamingContext) .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

Reference

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.2.1 **AdsServerException Constructor**

Initializes a new instance of the [AdsServerException](#) [▶ 1443] class.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected AdsServerException (
    SerializationInfo serializationInfo,
    StreamingContext streamingContext
)
```

Parameters

serializationInfo	Type: System.Runtime.Serialization.SerializationInfo The serialization information.
streamingContext	Type: System.Runtime.Serialization.StreamingContext The streaming context.

Reference






[AdsServerException Class](#) [▶ 1443]





[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.2.2 **AdsServerException Properties**

The [AdsServerException](#) [▶ 1443] type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	ErrorCode [▶ 1446]	Gets or sets the error code.
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)

	Name	Description
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Reference

[AdsServerException Class](#) [► 1443]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.2.2.1 AdsServerException.ErrorCode Property

Gets or sets the error code.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: [TwinCAT.Ads.Server](#) (in [TwinCAT.Ads.Server.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsServerErrorCode ErrorCode { get; set; }
```

Property Value

Type: [AmsServerErrorCode](#) [► 2008]

The error code.

Reference





[AdsServerException Class](#) [► 1443]





[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.2.3 AdsServerException Methods

The [AdsServerException](#) [► 1443] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)

	Name	Description
	GetObjectData [▶ 1447]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Overrides Exception.GetObjectData(SerializationInfo, StreamingContext) .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Reference

[AdsServerException Class](#) [▶ 1443]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.2.3.1 AdsServerException.GetObjectData Method

When overridden in a derived class, sets the [SerializationInfo](#) with information about the exception.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: [TwinCAT.Ads.Server](#) (in [TwinCAT.Ads.Server.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public override void GetObjectData(
    SerializationInfo info,
    StreamingContext context
)
```

Parameters

info	Type: System.Runtime.Serialization.SerializationInfo The SerializationInfo that holds the serialized object data about the exception being thrown.
context	Type: System.Runtime.Serialization.StreamingContext The StreamingContext that contains contextual information about the source or destination.

Implements

[ISerializable.GetObjectData\(SerializationInfo, StreamingContext\)](#)

Exceptions

Exception	Condition
ArgumentNullException	info

Reference


[AdsServerException Class](#) [▶ 1443]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.2.4 AdsServerException Events

The [AdsServerException](#) [▶ 1443] type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

Reference

[AdsServerException Class](#) [▶ 1443]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.3 ErrorEventArgs Class

This class implements the event arguments passed by the TcAdsServerExEvent.

Inheritance Hierarchy

[System.Object](#)

[System.EventArgs](#)

[TwinCAT.Ads.Server.ErrorEventArgs](#)

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229



Syntax

C#






```
public class ErrorEventArgs : EventArgs
```


The [ErrorEventArgs](#) type exposes the following members.

Properties

	Name	Description
	Exception [▶ 1449]	Returns the exception that caused the event.
	Message [▶ 1449]	Returns the exception message.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)

	Name	Description
	ToString	Returns a string that represents the current object. (Inherited from Object .)



Reference

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.3.1 ErrorEventArgs Properties

The [ErrorEventArgs](#) [► 1448] type exposes the following members.

Properties

	Name	Description
	Exception [► 1449]	Returns the exception that caused the event.
	Message [► 1449]	Returns the exception message.

Reference

[ErrorEventArgs Class \[► 1448\]](#)

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.3.1.1 ErrorEventArgs.Exception Property

Returns the exception that caused the event.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Exception Exception { get; }
```

Return Value

Type: [Exception](#)

The Exception that caused the event.

Reference

[ErrorEventArgs Class \[► 1448\]](#)

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.3.1.2 ErrorEventArgs.Message Property

Returns the exception message.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Message { get; }
```

Return Value

Type: [String](#)
The Exception message.

Reference







[ErrorEventArgs Class \[► 1448\]](#)

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.3.2 ErrorEventArgs Methods

The [ErrorEventArgs \[► 1448\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ErrorEventArgs Class \[► 1448\]](#)

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.4 LoopbackNotRegisteredException Class

The Tcp Loopback client is not registered. Implements the [AdsServerException \[► 1443\]](#)

Inheritance Hierarchy

[System.Object](#)

[System.Exception](#)

[TwinCAT.Ads.Server.AdsServerException \[► 1443\]](#)

[TwinCAT.Ads.Server.LoopbackNotRegisteredException](#)

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229










Syntax

C#









```
public class LoopbackNotRegisteredException : AdsServerException
```

The LoopbackNotRegisteredException type exposes the following members.


Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	ErrorCode [▶ 1446]	Gets or sets the error code. (Inherited from AdsServerException [▶ 1443].)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData [▶ 1447]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from AdsServerException [▶ 1443].)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Events

	Name	Description
	SerializeObjectState	Obsolete.

	Name	Description
		Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

Reference










[TwinCAT.Ads.Server Namespace](#) [► 1374]

[TwinCAT.Ads.Server.AdsServerException](#) [► 1443]

6.4.4.1 LoopbackNotRegisteredException Properties

The [LoopbackNotRegisteredException](#) [► 1450] type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	ErrorCode [► 1446]	Gets or sets the error code. (Inherited from AdsServerException [► 1443].)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Reference



[LoopbackNotRegisteredException Class](#) [► 1450]







[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.4.2 LoopbackNotRegisteredException Methods

The [LoopbackNotRegisteredException](#) [► 1450] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)

	Name	Description
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData [▶ 1447]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from AdsServerException [▶ 1443].)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Reference


[LoopbackNotRegisteredException Class](#) [▶ 1450]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.4.3 LoopbackNotRegisteredException Events

The [LoopbackNotRegisteredException](#) [▶ 1450] type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

Reference

[LoopbackNotRegisteredException Class](#) [▶ 1450]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.5 NotificationDataSample Class

This class implements an ADS Notification Sample. It contains the notification handle and the variable data.

Inheritance Hierarchy

[System.Object](#)

 TwinCAT.Ads.Server.NotificationDataSample

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229


Syntax

C#




```
public sealed class NotificationDataSample
```

The NotificationDataSample type exposes the following members.





Constructors

	Name	Description
	NotificationDataSample [▶ 1454]	Initializes a new instance of the NotificationDataSample class.

Properties

	Name	Description
	NotificationHandle [▶ 1455]	Gets the notification handle.
	SampleData [▶ 1455]	Gets the Sample Data.
	SampleSize [▶ 1456]	Gets the Sample Size.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.5.1 NotificationDataSample Constructor

Initializes a new instance of the [NotificationDataSample](#) [[▶ 1453](#)] class.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public NotificationDataSample(
    uint handle,
    ReadOnlyMemory<byte> data
)
```

Parameters

handle	Type: System.UInt32 The notification handle.
data	Type: System.ReadOnlyMemory.Byte . The Data content.

Reference




[NotificationDataSample Class](#) [[▶ 1453](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.5.2 NotificationDataSample Properties

The [NotificationDataSample](#) [[▶ 1453](#)] type exposes the following members.

Properties

	Name	Description
	NotificationHandle [▶ 1455]	Gets the notification handle.
	SampleData [▶ 1455]	Gets the Sample Data.
	SampleSize [▶ 1456]	Gets the Sample Size.

Reference

[NotificationDataSample Class](#) [[▶ 1453](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.5.2.1 NotificationDataSample.NotificationHandle Property

Gets the notification handle.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint NotificationHandle { get; }
```

Property Value

Type: [UInt32](#)

Reference

[NotificationDataSample Class](#) [[▶ 1453](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.5.2.2 NotificationDataSample.SampleData Property

Gets the Sample Data.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyMemory<byte> SampleData { get; }
```

Property Value

Type: [ReadOnlyMemory.Byte](#).

Reference[NotificationDataSample Class \[► 1453\]](#)[TwinCAT.Ads.Server Namespace \[► 1374\]](#)**6.4.5.2.3 NotificationDataSample.SampleSize Property**

Gets the Sample Size.





Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)**Assembly:** TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public int SampleSize { get; }
```

Property ValueType: [Int32](#)**Reference**[NotificationDataSample Class \[► 1453\]](#)[TwinCAT.Ads.Server Namespace \[► 1374\]](#)**6.4.5.3 NotificationDataSample Methods**

The [NotificationDataSample \[► 1453\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference[NotificationDataSample Class \[► 1453\]](#)[TwinCAT.Ads.Server Namespace \[► 1374\]](#)**6.4.6 NotificationSamplesStamp Class**

This class implements an ADS Stamp Header containing multiple ADS Notification Samples (TcAdsStampHeader)

Inheritance Hierarchy

System.Object

TwinCAT.Ads.Server.NotificationSamplesStamp

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229



Syntax

C#




```
public sealed class NotificationSamplesStamp
```

The NotificationSamplesStamp type exposes the following members.





Constructors

	Name	Description
	NotificationSamplesStamp(.NotificationDataSample.) [▶ 1458]	Initializes a new instance of the NotificationSamplesStamp class.
	NotificationSamplesStamp(DateTimeOffset, .NotificationDataSample.) [▶ 1458]	Initializes a new instance of the NotificationSamplesStamp class.

Properties

	Name	Description
	NotificationSamples [▶ 1459]	Get the array of notification samples contained in this header.
	SamplesCount [▶ 1461]	Get the number of notification samples contained in this header.
	TimeStamp [▶ 1460]	Gets the time stamp of the NotificationSamplesStamp.



Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

Also see about this

-  [NotificationSamplesStamp Constructor](#) [[▶ 1458](#)]
-  [NotificationSamplesStamp.NumSamples Property](#) [[▶ 1460](#)]

6.4.6.1 NotificationSamplesStamp Constructor

Overload List

	Name	Description
	NotificationSamplesStamp(.NotificationDataSample.) [▶ 1458]	Initializes a new instance of the NotificationSamplesStamp [▶ 1456] class.
	NotificationSamplesStamp(DateTimeOffset, .NotificationDataSample.) [▶ 1458]	Initializes a new instance of the NotificationSamplesStamp [▶ 1456] class.

Reference

[NotificationSamplesStamp Class](#) [[▶ 1456](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

Also see about this

 [NotificationSamplesStamp Class](#) [[▶ 1456](#)]

 [TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.6.1.1 NotificationSamplesStamp Constructor (.NotificationDataSample.)

Initializes a new instance of the [NotificationSamplesStamp](#) [[▶ 1456](#)] class.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public NotificationSamplesStamp(
    NotificationDataSample[] samples
)
```

Parameters

samples	Type: .TwinCAT.Ads.Server.NotificationDataSample [▶ 1453]. The samples.
---------	--

Reference

[NotificationSamplesStamp Class](#) [[▶ 1456](#)]

[NotificationSamplesStamp Overload](#) [[▶ 1458](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.6.1.2 NotificationSamplesStamp Constructor (DateTimeOffset, .NotificationDataSample.)

Initializes a new instance of the [NotificationSamplesStamp](#) [[▶ 1456](#)] class.

Namespace: [TwinCAT.Ads.Server \[▸ 1374\]](#)

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public NotificationSamplesStamp(
    DateTimeOffset timestamp,
    NotificationDataSample[] samples
)
```

Parameters

timestamp	Type: System.DateTimeOffset The timestamp.
samples	Type: .TwinCAT.Ads.Server.NotificationDataSample [▸ 1453] . The notification samples.

Reference

[NotificationSamplesStamp Class \[▸ 1456\]](#)




[NotificationSamplesStamp Overload \[▸ 1458\]](#)

[TwinCAT.Ads.Server Namespace \[▸ 1374\]](#)

6.4.6.2 NotificationSamplesStamp Properties

The [NotificationSamplesStamp \[▸ 1456\]](#) type exposes the following members.

Properties

	Name	Description
	NotificationSamples [▸ 1459]	Get the array of notification samples contained in this header.
	SamplesCount [▸ 1461]	Get the number of notification samples contained in this header.
	TimeStamp [▸ 1460]	Gets the time stamp of the NotificationSamplesStamp [▸ 1456] .

Reference

[NotificationSamplesStamp Class \[▸ 1456\]](#)

[TwinCAT.Ads.Server Namespace \[▸ 1374\]](#)

Also see about this

-  [NotificationSamplesStamp.NumSamples Property \[▸ 1460\]](#)

6.4.6.2.1 NotificationSamplesStamp.NotificationSamples Property

Get the array of notification samples contained in this header.

Namespace: [TwinCAT.Ads.Server \[▸ 1374\]](#)

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public NotificationDataSample[] NotificationSamples { get; }
```

Property Value

Type: [.NotificationDataSample](#) [▶ 1453].

Reference

[NotificationSamplesStamp Class](#) [▶ 1456]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.6.2.2 NotificationSamplesStamp.NumSamples Property

Get the number of notification samples contained in this header.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public int NumSamples { get; }
```

Property Value

Type: [Int32](#)

Reference

[NotificationSamplesStamp Class](#) [▶ 1456]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.6.2.3 NotificationSamplesStamp.TimeStamp Property

Gets the time stamp of the [NotificationSamplesStamp](#) [▶ 1456].

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTimeOffset TimeStamp { get; }
```

Property Value

Type: [DateTimeOffset](#)
The time stamp.

Reference

[NotificationSamplesStamp Class](#) [▶ 1456]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.6.2.4 NotificationSamplesStamp.SamplesCount Property

Get the number of notification samples contained in this header.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int SamplesCount { get; }
```

Property Value

Type: [Int32](#)

Reference





[NotificationSamplesStamp Class](#) [[▶ 1456](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.6.3 NotificationSamplesStamp Methods

The [NotificationSamplesStamp](#) [[▶ 1456](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[NotificationSamplesStamp Class](#) [[▶ 1456](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.7 ServerConnectionState Enumeration

The Server Connection State

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum ServerConnectionState
```

Members

	Member name	Value	Description
	None	0	None / Uninitialized / Disconnect

	Member name	Value	Description
	Disconnected	1	Disconnected State.
	Connected	2	Connected state.

Reference

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.8 ServerConnectionStateChangedEventArgs Class

Class ConnectionStateChangedEventArgs (Server Connections)

Inheritance Hierarchy

[System.Object](#)

[System.EventArgs](#)

[TwinCAT.Ads.Server.ServerConnectionStateChangedEventArgs](#)

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229


Syntax

C#



```
public class ServerConnectionStateChangedEventArgs : EventArgs
```

The ServerConnectionStateChangedEventArgs type exposes the following members.






Constructors


	Name	Description
	ServerConnectionStateChangedEventArgs [► 1463]	Initializes a new instance of the ServerConnectionStateChangedEventArgs class.

Properties

	Name	Description
	Address [► 1463]	The Server address
	State [► 1464]	The Connection state

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)

	Name	Description
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

[System.EventArgs](#)

6.4.8.1 ServerConnectionStateChangedEventArgs Constructor

Initializes a new instance of the [ServerConnectionStateChangedEventArgs](#) [▶ 1462] class.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ServerConnectionStateChangedEventArgs (
    AmsAddress address,
    ServerConnectionState state
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [▶ 752] The address.
state	Type: TwinCAT.Ads.Server.ServerConnectionState [▶ 1461] The state.

Reference



[ServerConnectionStateChangedEventArgs Class](#) [▶ 1462]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.8.2 ServerConnectionStateChangedEventArgs Properties

The [ServerConnectionStateChangedEventArgs](#) [▶ 1462] type exposes the following members.

Properties

	Name	Description
	Address [▶ 1463]	The Server address
	State [▶ 1464]	The Connection state

Reference

[ServerConnectionStateChangedEventArgs Class](#) [▶ 1462]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.8.2.1 ServerConnectionStateChangedEventArgs.Address Property

The Server address

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsAddress Address { get; }
```

Property Value

Type: [AmsAddress](#) [[▶ 752](#)]

Reference

[ServerConnectionStateChangedEventArgs Class](#) [[▶ 1462](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.8.2 ServerConnectionStateChangedEventArgs.State Property

The Connection state

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ServerConnectionState State { get; }
```

Property Value

Type: [ServerConnectionState](#) [[▶ 1461](#)]

Reference






[ServerConnectionStateChangedEventArgs Class](#) [[▶ 1462](#)]


[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.8.3 ServerConnectionStateChangedEventArgs Methods

The [ServerConnectionStateChangedEventArgs](#) [[▶ 1462](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)

	Name	Description
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ServerConnectionStateChangedEventArgs Class \[► 1462\]](#)

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.9 ServerNotConnectedException Class

The AdsServer is not connected. Implements the [AdsServerException \[► 1443\]](#)

Inheritance Hierarchy

[System.Object](#)

[System.Exception](#)

[TwinCAT.Ads.Server.AdsServerException \[► 1443\]](#)

[TwinCAT.Ads.Server.ServerNotConnectedException](#)

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#









```
public class ServerNotConnectedException : AdsServerException
```


The ServerNotConnectedException type exposes the following members.

Constructors









	Name	Description
	ServerNotConnectedException [► 1468]	Initializes a new instance of the ServerNotConnectedException class.

Properties


	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	ErrorCode [► 1446]	Gets or sets the error code. (Inherited from AdsServerException [► 1443] .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)

	Name	Description
	<u>TargetSite</u>	Gets the method that throws the current exception. (Inherited from <u>Exception</u> .)

Methods

	Name	Description
	<u>Equals</u>	Determines whether the specified object is equal to the current object. (Inherited from <u>Object</u> .)
	<u>Finalize</u>	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from <u>Object</u> .)
	<u>GetBaseException</u>	When overridden in a derived class, returns the <u>Exception</u> that is the root cause of one or more subsequent exceptions. (Inherited from <u>Exception</u> .)
	<u>GetHashCode</u>	Serves as the default hash function. (Inherited from <u>Object</u> .)
	<u>GetObjectData</u> [▶ 1447]	When overridden in a derived class, sets the <u>SerializationInfo</u> with information about the exception. (Inherited from <u>AdsServerException</u> [▶ 1443].)
	<u>GetType</u>	Gets the runtime type of the current instance. (Inherited from <u>Exception</u> .)
	<u>MemberwiseClone</u>	Creates a shallow copy of the current <u>Object</u> . (Inherited from <u>Object</u> .)
	<u>ToString</u>	Creates and returns a string representation of the current exception. (Inherited from <u>Exception</u> .)

Events

	Name	Description
	<u>SerializeObjectState</u>	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from <u>Exception</u> .)

Reference





[TwinCAT.Ads.Server Namespace](#) [▶ 1374]






[TwinCAT.Ads.Server.AdsServerException](#) [▶ 1443]

6.4.9.1 ServerNotConnectedException Properties

The [ServerNotConnectedException](#) [▶ 1465] type exposes the following members.

Properties

	Name	Description
	<u>Data</u>	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from <u>Exception</u> .)
	<u>ErrorCode</u> [▶ 1446]	Gets or sets the error code. (Inherited from <u>AdsServerException</u> [▶ 1443].)
	<u>HelpLink</u>	Gets or sets a link to the help file associated with this exception. (Inherited from <u>Exception</u> .)
	<u>HResult</u>	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from <u>Exception</u> .)

	Name	Description
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Reference









[ServerNotConnectedException Class](#) [▶ 1465]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.9.2 ServerNotConnectedException Methods

The [ServerNotConnectedException](#) [▶ 1465] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData [▶ 1447]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from AdsServerException [▶ 1443].)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Reference


[ServerNotConnectedException Class](#) [▶ 1465]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.9.3 ServerNotConnectedException Events

The [ServerNotConnectedException](#) [▶ 1465] type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

Reference

[ServerNotConnectedException Class](#) [► 1465]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.9.4 ServerNotConnectedException Constructor

Initializes a new instance of the [ServerNotConnectedException](#) [► 1465] class.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ServerNotConnectedException()
```

Reference

[ServerNotConnectedException Class](#) [► 1465]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.10 AdsSymbolicServer Class

AdsServer class that supports Symbolic information and Notifications. Implements the [AdsServer](#) [► 1375]

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.Server.AdsServer](#) [► 1375]

[TwinCAT.Ads.Server.AdsSymbolicServer](#)

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public abstract class AdsSymbolicServer : AdsServer,
    ISymbolProvider, ISymbolServer
```

The AdsSymbolicServer type exposes the following members.

Properties

	Name	Description
	AmsServer [► 1384]	Gets the the internal AmsServer [► 1384] object. (Inherited from AdsServer [► 1375].)
	DataTypes [► 1482]	Gets the data types

Name	Description
DefaultValueEncoding [▶ 1482]	Gets/Sets the default value encoding.
InstancePathEncoding [▶ 1483]	Gets or sets the instance path encoding (Default is the Windows CodePage)
IsConnected [▶ 1384]	Gets a value indicating whether AdsServer [▶ 1375] is connected. (Inherited from AdsServer [▶ 1375].)
IsDisconnecting [▶ 1384]	Indicates, that the AdsServer [▶ 1375] is actually disconnecting. (Inherited from AdsServer [▶ 1375].)
IsDisposed [▶ 1385]	Gets a value indicating whether this instance is disposed. (Inherited from AdsServer [▶ 1375].)
IsDynamicPort [▶ 1387]	Gets a value indicating this AdsServer [▶ 1375] has a dynamic/unfixed port. (Inherited from AdsServer [▶ 1375].)
Logger [▶ 1385]	Gets the logger object. (Inherited from AdsServer [▶ 1375].)
PlatformPointerSize [▶ 1483]	Gets the Platform pointer size of this Server
RootNamespaceName [▶ 1484]	Gets the name of the root namespace
ServerAddress [▶ 1386]	The AMS address of this server. (Inherited from AdsServer [▶ 1375].)
ServerName [▶ 1386]	Gets the name of the server. (Inherited from AdsServer [▶ 1375].)
ServerPort [▶ 1387]	Gets the server port. (Inherited from AdsServer [▶ 1375].)
ServerVersion [▶ 1387]	Gets the Version of the Server. (Inherited from AdsServer [▶ 1375].)
Symbols [▶ 1484]	Gets the symbols.

Methods

Name	Description
AddDeviceNotificationRequest [▶ 1393]	Sends an ADS Add Device Notification request (synchronous). (Inherited from AdsServer [▶ 1375].)
AddDeviceNotificationRequestAsync [▶ 1394]	Sends an ADS Add Device Notification request (async) (Inherited from AdsServer [▶ 1375].)
AddDeviceNotificationResponseAsync [▶ 1395]	Sends an ADS Add Device Notification response. (Inherited from AdsServer [▶ 1375].)
ConnectServer [▶ 1396]	Connect this ADS server to the local ADS router. (Inherited from AdsServer [▶ 1375].)
ConnectServerAndWaitAsync [▶ 1396]	Registers the AdsServer [▶ 1375] at the router asynchronously. (Inherited from AdsServer [▶ 1375].)

	Name	Description
	DeleteDeviceNotificationRequest [▶ 1398]	Sends an ADS Delete Device Notification request (synchronous). (Inherited from AdsServer [▶ 1375].)
	DeleteDeviceNotificationRequestAsync [▶ 1399]	Sends an ADS Delete Device Notification request (async). (Inherited from AdsServer [▶ 1375].)
	DeleteDeviceNotificationResponseAsync [▶ 1400]	Sends an ADS Delete Device Notification response. (Inherited from AdsServer [▶ 1375].)
	DeviceNotificationRequestAsync [▶ 1400]	Sends an ADS Device Notification request asynchronously (Inherited from AdsServer [▶ 1375].)
	DeviceNotificationRequestSync [▶ 1401]	Sends an ADS Device Notification request (sync) (Inherited from AdsServer [▶ 1375].)
	Disconnect [▶ 1402]	Disconnects this ADS server from the local ADS router. (Inherited from AdsServer [▶ 1375].)
	Dispose . [▶ 1403]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from AdsServer [▶ 1375].)
	Dispose(Boolean) [▶ 1403]	Releases unmanaged and - optionally - managed resources. (Inherited from AdsServer [▶ 1375].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize [▶ 1403]	Finalizes an instance of the AdsServer [▶ 1375] class. (Inherited from AdsServer [▶ 1375].)
	FireNotificationAsync [▶ 1490]	Fires a single notification to the specified Address.
	FireNotifications [▶ 1490]	Fires outstanding cyclic notifications.
	FireNotificationsAsync [▶ 1491]	Check internal tables for outstanding cyclic Notifications and fire them asynchronously.
	GetDataTypesAsync [▶ 1491]	Gets the data types asynchronously.
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetServerName [▶ 1441]	Gets the name of the server. (Inherited from AdsServer [▶ 1375].)
	GetSymbolsAsync [▶ 1492]	Gets the symbols asynchronously
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	HasValueChanged [▶ 1493]	Determines whether the value of the Symbol has changed.
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)

	Name	Description
	OnAddDeviceNotificationAsync [▶ 1493]	Called when an ADS Add Device Notification indication is received. (Overrides AdsServer.OnAddDeviceNotificationAsync(AmsAddress, UInt32, UInt32, UInt32, Int32, AmsAddress, NotificationSettings, CancellationToken) [▶ 1429].)
	OnAddDeviceNotificationConfirmationAsync [▶ 1404]	Called when an ADS Add Device Notification confirmation is received. (Inherited from AdsServer [▶ 1375].)
	OnBeforeConnected [▶ 1494]	Handler function that is called, when the AdsServer [▶ 1375] is connected, but before calling OnConnected . [▶ 1405]. (Overrides AdsServer.OnBeforeConnected . [▶ 1430].)
	OnConnected [▶ 1495]	Handler function that is called, when the AdsServer [▶ 1375] is connected. (Overrides AdsServer.OnConnected . [▶ 1405].)
	OnCreateSymbols [▶ 1509]	Handler function to create the Initial Symbols of the server.
	OnDeleteDeviceNotificationAsync [▶ 1495]	Called when a DeleteDeviceNotification request is indicated. (Overrides AdsServer.OnDeleteDeviceNotificationAsync(AmsAddress, UInt32, UInt32, CancellationToken) [▶ 1430].)
	OnDeleteDeviceNotificationConfirmationAsync [▶ 1431]	Called when an ADS Delete Device Notification confirmation is received. (Inherited from AdsServer [▶ 1375].)
	OnDisconnect [▶ 1496]	Called when the AdsServer [▶ 1375] is about to be disconnected. (Overrides AdsServer.OnDisconnect . [▶ 1405].)
	OnFireNotifications [▶ 1496]	Handler function for firing notifications to the specified targets.
	OnFireNotificationsAsync [▶ 1497]	Handler function for firing notifications to the specified targets.
	OnGetValue [▶ 1497]	Handler function to determine the (stored) value of the symbol from the internal caches of the AdsSymbolicServer .
	OnReadAsync [▶ 1498]	Called when an ADS Read indication is received. (Overrides AdsServer.OnReadAsync(AmsAddress, UInt32, UInt32, UInt32, Int32, CancellationToken) [▶ 1432].)
	OnReadConfirmationAsync [▶ 1433]	Called when an ADS Read confirmation is received. (Inherited from AdsServer [▶ 1375].)
	OnReadDeviceInfoConfirmationAsync [▶ 1434]	Called when an ADS Read Device Info confirmation is received. (Inherited from AdsServer [▶ 1375].)
	OnReadDeviceStateAsync [▶ 1435]	Called when an ADS Read State indication is received. (Inherited from AdsServer [▶ 1375].)
	OnReadDeviceStateConfirmationAsync [▶ 1435]	Called when an ADS Read State confirmation is received. (Inherited from AdsServer [▶ 1375].)
	OnReadIglloAsync [▶ 1499]	Handler function for handling pure IG/IO Read indications

	Name	Description
	OnReadRawValue [▶ 1500]	Handler function for Reading the internal value data in raw format.
	OnReadWriteAsync [▶ 1500]	Called when an ADS Read Write indication is received. (Overrides AdsServer.OnReadWriteAsync(AmsAddress, UInt32, UInt32, UInt32, Int32, ReadOnlyMemory.Byte., CancellationToken) [▶ 1436].)
	OnReadWriteConfirmationAsync [▶ 1437]	Called when an ADS Read Write confirmation is received. (Inherited from AdsServer [▶ 1375].)
	OnReadWritelgloAsync [▶ 1501]	Handler function for handling of pure IG/IO calls.
	OnRouterNotification [▶ 1405]	Handler Function for a Router Notification. (Inherited from AdsServer [▶ 1375].)
	OnRpcInvoke [▶ 1502]	Handler function called when an RpcInvoke occurs.
	OnServerConnectionStateChanged [▶ 1406]	Handles the ServerConnectionStateChanged [▶ 1442] event. (Inherited from AdsServer [▶ 1375].)
	OnSetValue [▶ 1503]	Handler function to store a new Symbol value within internal caches of the AdsSymbolicServer .
	OnWriteAsync [▶ 1503]	Called when an ADS Write indication is received. (Overrides AdsServer.OnWriteAsync(AmsAddress, UInt32, UInt32, UInt32, ReadOnlyMemory.Byte., CancellationToken) [▶ 1438].)
	OnWriteConfirmationAsync [▶ 1439]	Called when an ADS Write confirmation is received. (Inherited from AdsServer [▶ 1375].)
	OnWriteControlAsync [▶ 1440]	Called when an ADS Write Control indication is received. (Inherited from AdsServer [▶ 1375].)
	OnWriteControlConfirmationAsync [▶ 1441]	Called when an ADS Write Control confirmation is received. (Inherited from AdsServer [▶ 1375].)
	OnWritelgloAsync [▶ 1504]	Handler function for handling of pure Write IG/IO Indications.
	OnWriteRawValue [▶ 1505]	Handler function for writing the symbol value data in raw format.
	ReadDeviceInfoRequestAsync [▶ 1409]	Sends an ADS Read Device Info request asynchronously (Inherited from AdsServer [▶ 1375].)
	ReadDeviceInfoRequestSync [▶ 1409]	Sends an ADS Read Device Info request synchronously. (Inherited from AdsServer [▶ 1375].)
	ReadDeviceInfoResponseAsync [▶ 1410]	Sends an ADS Read Device Info response. (Inherited from AdsServer [▶ 1375].)

	Name	Description
	ReadDeviceStateRequestAsync [► 1412]	Sends an ADS Read State request (asynchronous) (Inherited from AdsServer [► 1375].)
	ReadDeviceStateRequestSync [► 1413]	Sends an ADS Read State request (synchronous) (Inherited from AdsServer [► 1375].)
	ReadDeviceStateResponseAsync [► 1413]	Sends an ADS Read State response. (Inherited from AdsServer [► 1375].)
	ReadRequest [► 1415]	Sends an ADS Read Request. (Inherited from AdsServer [► 1375].)
	ReadRequestAsync [► 1416]	Sends an ADS Read Request asynchronously. (Inherited from AdsServer [► 1375].)
	ReadResponseAsync [► 1416]	Sends an ADS Read response. (Inherited from AdsServer [► 1375].)
	ReadWriteRequestAsync [► 1419]	Sends an ADS Read Write request. (Inherited from AdsServer [► 1375].)
	ReadWriteRequestSync [► 1419]	Sends an ADS Read Write request synchronously (Inherited from AdsServer [► 1375].)
	ReadWriteResponseAsync [► 1420]	Sends an ADS Read Write Response. (Inherited from AdsServer [► 1375].)
	ResetCachedSymbolicData [► 1506]	Resets the cached symbolic data.
	SetValue(ISymbol, Object) [► 1507]	Sets the value of the symbol.
	SetValue(String, Object) [► 1507]	Sets the value of the symbol.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryGetDataTypes [► 1508]	Tries to get the symbols from the device target.
	TryGetSymbols [► 1509]	Tries to geth the symbols from the device target.
	WriteControlRequest [► 1423]	Sends an ADS Write Control request (synchronous) (Inherited from AdsServer [► 1375].)
	WriteControlRequestAsync [► 1424]	Sends an ADS Write Control request (asynchronous). (Inherited from AdsServer [► 1375].)
	WriteControlRequestSync [► 1425]	Sends an ADS Write Control request (synchronous). (Inherited from AdsServer [► 1375].)
	WriteControlResponseAsync [► 1425]	Sends an ADS Write Control response. (Inherited from AdsServer [► 1375].)

	Name	Description
	WriteRequest [▶ 1427]	Sends an ADS Write request synchronously. (Inherited from AdsServer [▶ 1375].)
	WriteRequestAsync [▶ 1428]	Sends an ADS Write request asynchronously. (Inherited from AdsServer [▶ 1375].)
	WriteResponseAsync [▶ 1428]	Sends an ADS Write response. (Inherited from AdsServer [▶ 1375].)

Events

	Name	Description
	ServerConnectionStateChanged [▶ 1442]	The connection status has changed (Inherited from AdsServer [▶ 1375].)

Fields

	Name	Description
	handleTable [▶ 1510]	The handle table (InstancePath --> Symbol handle)
	notificationTable [▶ 1511]	The notification table
	notificationTrigger [▶ 1511]	The notification trigger Source
	serverVersion [▶ 1443]	The version of the AdsServer (Inherited from AdsServer [▶ 1375].)
	symbolFactory [▶ 1511]	The symbol factory
	symbolNameMarshaller [▶ 1512]	The SymbolName marshaler
	symbolVersion [▶ 1512]	The symbol version as respond by SymbolVersion [▶ 1231]

Remarks

DataTypes and hierarchical trees of symbols can be downloaded from this by symbolic information access. The symbolic access is necessary for the following functionality:

- Browsing of Symbols and DataTypes
- Type Safe Read/Write access of Process Values of AdsServers

This symbolic access is for example used by the following TwinCAT tools:

- TwinCAT Scope
- TwinCAT Powershell Management module [TcXaeMgmt](#)

Examples

The following sample shows how to derive from the [AdsServer](#) [▶ 1375] class and create your own Customized ADS Server.

C#

```

public class SymbolicTestServer : AdsSymbolicServer
{
    /// <summary>
    /// AmsPort of the SymbolicTestServer
    /// </summary>
    /// <remarks>
    /// User Server Ports must be in between
    /// AmsPort.CUSTOMER_FIRST (25000) <= PORT <= AmsPort.CUSTOMER_LAST (25999)
    /// to not conflict with Beckhoff prerreserved servers!
    /// </remarks>
    static ushort s_Port = 25000;

    /// <summary>
    /// Dictionary containing the Values of the Symbols (Symbol -- Value)
    /// </summary>
    Dictionary<ISymbol, object> _symbolValues = new Dictionary<ISymbol, object>();

    /// <summary>
    /// Gets the SymbolValue Dictionary of the <see cref="SymbolicTestServer"/>.
    /// </summary>
    /// <value>The symbol values.</value>
    public IDictionary<ISymbol, object> SymbolValues => _symbolValues;

    /// <summary>
    /// Symbolic Marshaler for Values
    /// </summary>
    SymbolicAnyTypeMarshaler _symbolMarshaler = new SymbolicAnyTypeMarshaler();

    /// <summary>
    /// Initializes a new instance of the <see cref="SymbolicTestServer"/> class.
    /// </summary>
    public SymbolicTestServer()
    : base(s_Port, "SymbolicTestServer", null)
    {
        /// AMS Router endpoint can be changed via environment variables which is
        /// beneficial in containerized setups where the AMS router is not listening
        /// at the default loopback address and port
        IPAddress ipEndpoint;
        if (!
IPAddress.TryParse(System.Environment.GetEnvironmentVariable("ENV_AmsConfiguration__LoopbackAddress"
), out ipEndpoint))
        {
            ipEndpoint = IPAddress.Loopback;
        }

        int port;
        if (!
int.TryParse(System.Environment.GetEnvironmentVariable("ENV_AmsConfiguration__LoopbackPort"), out po
rt))
        {
            port = 48898;
        }

        AmsConfiguration.RouterEndPoint = new IPEndPoint(ipEndpoint, port);
    }

    IDisposable _changeValueObserver = null;

    /// <summary>
    /// Called when [connected].
    /// </summary>
    protected override void OnConnected()
    {
        this.AddSymbols()
            .AddNotificationTrigger();

        // An Observable.Interval is used to simulate changed values (on a 1 Second base)
        IObservable<long> changeValueTrigger = Observable.Interval(TimeSpan.FromSeconds(1.0), Scheduler.
Default);
        _changeValueObserver = changeValueTrigger.Subscribe(toggleValues);
        base.OnConnected();
    }

    protected override bool OnDisconnect()
    {
        {
            if (_changeValueObserver != null)
                _changeValueObserver.Dispose();
        }
    }
}

```

```

return base.OnDisconnect();
}

private void toggleValues(long count)
{
SetValue("Globals.int1", (short)count);
}

/// <summary>
/// Creates an Notification trigger for the Notifications base tick.
/// </summary>
private SymbolicTestServer AddNotificationTrigger()
{
base.notificationTrigger.Add(new BaseTickTrigger(TimeSpan.FromMilliseconds(100)));
return this;
}

/// <summary>
/// Create the Symbolic information DataAreas, DataTypes and Symbols.
/// </summary>
private SymbolicTestServer AddSymbols()
{
// Create some Primitive types
PrimitiveType dtBool = new PrimitiveType("BOOL", typeof(bool)); // 1-Byte size
PrimitiveType dtInt = new PrimitiveType("INT", typeof(short)); // 2-Byte size
PrimitiveType dtDInt = new PrimitiveType("DINT", typeof(int)); // 4-Byte size

// Create an TwinCAT specific Type PCCH (for testing purposes)
// Which is used for interop to C++ TCOM Modules
// It defines a pointer to an UTF8 String, where the Length of the data is specified
// With a LengthIs parameter
PrimitiveType dtByte = new PrimitiveType("BYTE", typeof(byte)); // Necessary for the PCCH Type (
which is in fact an Alias to POINTER TO BYTE)!
PCCHType dtPCCH = PCCHType.Create();

// StringType (fixed size)
StringType dtString = new StringType(80, Encoding.Unicode);

// Struct Type
// Associated to this struct Type, we use a corresponding 'ValueType' within our ServerValues
// to hold the StructInstance value. This class/value is used to marshal/unmarshal struct types.
// See the C# class 'MyStruct' at the bottom of this example.
StructType dtStruct = new StructType("MYSTRUCT", null)
    // Add Members
    .AddAligned(new Member("name", dtString))
    .AddAligned(new Member("a", dtBool))
    .AddAligned(new Member("b", dtInt))
    .AddAligned(new Member("c", dtDInt));

// Create an RPC Invokable Type
// Firstly the methods ...

// INT Method1([in] INT i1, [in] i2)
RpcMethod rpc1 = new RpcMethod("Method1")
    //Add Parameters
    .AddParameter("i1", dtInt, MethodParamFlags.In)
    .AddParameter("i2", dtInt, MethodParamFlags.In)
    //Set Return Type
    .SetReturnType(dtInt);

// INT Method2([in] INT in1, [out] INT out1)
RpcMethod rpc2 = new RpcMethod("Method2")
    .AddParameter("in1", dtInt, MethodParamFlags.In)
    .AddParameter("out1", dtInt, MethodParamFlags.Out)
    .SetReturnType(dtInt);

// STRING[80] Method3([in] INT len, [in][LengthIs = 1] PCCH str)
RpcMethod rpc3 = new RpcMethod("Method3")
    .AddParameter("len", dtInt, MethodParamFlags.In)
    // Referencing to parameter 'len' via LengthIs Attribute
    .AddParameter("str", dtPCCH, MethodParamFlags.In, 1)
    .SetReturnType(dtString);

// STRING[80] Method4([in] INT len, [out][LengthIs = 1] PCCH str)
RpcMethod rpc4 = new RpcMethod("Method4")
    .AddParameter("len", dtInt, MethodParamFlags.In)
    .AddParameter("str", dtPCCH, MethodParamFlags.Out, 1)
    .SetReturnType(dtString);

```

```

// STRING[80] Method5([in] INT len, [out][LengthIs = 1] PCCH str)
RpcMethod rpc5 = new RpcMethod("Method5")
    .AddParameter("len", dtInt, MethodParamFlags.In)
    .AddParameter("str", dtPCCH, MethodParamFlags.Out, 1)
    .SetReturnType(dtString);

// ... Create the RpcStructType itself and bind the Members/Methods
StructType dtStructRpc = new StructType("MYRPCSTRUCT");
dtStructRpc
    // Add Members (Fields)
    .AddAligned(new Member("name", dtString))
    .AddAligned(new Member("a", dtBool))
    .AddAligned(new Member("b", dtInt))
    .AddAligned(new Member("c", dtDInt))
    // Add Methods
    .AddMethod(rpc1)
    .AddMethod(rpc2);

// Create an Array Type
// Create the Dimensions
IDimensionCollection dims = new DimensionCollection()
    // Add Dimension
    .AddDimension(new Dimension(0, 4))
    .AddDimension(new Dimension(0, 2));

// Create the Array Type itself
ArrayType dtArray = new ArrayType(dtInt, dims);

// Create an Enumeration Type
// Define the Enum Fields/Values
EnumValueCollection<int> enumValues = new EnumValueCollection<int>()
    // Add Enumeration Fields/Entry
    .AddValue("None", 0)
    .AddValue("Red", 1)
    .AddValue("Yellow", 2)
    .AddValue("Green", 3);

// Create the EnumType
EnumType<int> dtEnum = new EnumType<int>("MYENUM", dtDInt, enumValues);

// Alias Type
AliasType dtAlias = new AliasType("MYALIAS", dtEnum);

//Pointer Type
PointerType dtPointer = new PointerType(dtInt, this.PlatformPointerSize);

// Reference Type
ReferenceType dtReference = new ReferenceType(dtInt, this.PlatformPointerSize);

// Add the Types to the internal SymbolFactory
base.symbolFactory
    .AddType(dtBool)
    .AddType(dtInt)
    .AddType(dtDInt)
    .AddType(dtString)
    .AddType(dtStruct)
    .AddType(dtArray)
    .AddType(dtEnum)
    .AddType(dtAlias)
    .AddType(dtPointer)
    .AddType(dtReference)
    .AddType(dtStructRpc)
    .AddType(dtByte)
    .AddType(dtPCCH);

// Define some ProcessImage DataAreas (virtual) used for IndexGroup/IndexOffset Alignments
DataArea general = new DataArea("General", 0x01, 0x1000, 0x1000);
DataArea globals = new DataArea("Globals", 0x02, 0x1000, 0x1000);
DataArea statics = new DataArea("Statics", 0x03, 0x1000, 0x1000);

// Add the DataAreas to the SymbolFactory
base.symbolFactory
    .AddDataArea(general)
    .AddDataArea(globals)
    .AddDataArea(statics);

// Create the Symbols (Symbol Instances)
// With DataType and define its DataArea in the (virtual)ProcessImage
base.symbolFactory
    .AddSymbol("Globals.bool1", dtBool, globals)

```

```

.AddSymbol("Globals.int1", dtInt, globals)
.AddSymbol("Globals.dint1", dtDInt, globals)
.AddSymbol("Globals.string1", dtString, globals)
.AddSymbol("Globals.myStruct1", dtStruct, globals)
.AddSymbol("Globals.myArray1", dtArray, globals)
.AddSymbol("Globals.myEnum1", dtEnum, globals)
.AddSymbol("Globals.myAlias1", dtAlias, globals)
.AddSymbol("Globals.pointer1", dtPointer, globals)
.AddSymbol("Globals.reference1", dtReference, globals)
.AddSymbol("Globals.rpcInvoke1", dtStructRpc, globals)
.AddSymbol("Main.bool1", dtBool, general)
.AddSymbol("Main.int1", dtInt, general)
.AddSymbol("Main.dint1", dtDInt, general)
.AddSymbol("Main.string1", dtString, general)
.AddSymbol("Main.myStruct1", dtStruct, general)
.AddSymbol("Main.myArray1", dtArray, general)
.AddSymbol("Main.myEnum1", dtEnum, general)
.AddSymbol("Main.myAlias1", dtAlias, general)
.AddSymbol("Main.pointer1", dtPointer, general)
.AddSymbol("Main.reference1", dtReference, general)
.AddSymbol("Main.rpcInvoke1", dtStructRpc, general);

// Here we set the initial values of our Symbol instances
_symbolValues.Add(this.Symbols["Globals.bool1"], true);
_symbolValues.Add(this.Symbols["Globals.int1"], (short)42);
_symbolValues.Add(this.Symbols["Globals.dint1"], 42);
_symbolValues.Add(this.Symbols["Globals.string1"], "Hello world!");
_symbolValues.Add(this.Symbols["Globals.myStruct1"], new MyStruct("Globals.myStruct1", true, 42,
99));
_symbolValues.Add(this.Symbols["Globals.myArray1"], new short[4, 2]); ;
_symbolValues.Add(this.Symbols["Globals.myEnum1"], "Yellow");
_symbolValues.Add(this.Symbols["Globals.myAlias1"], "Red");
_symbolValues.Add(this.Symbols["Globals.pointer1"], 0);
_symbolValues.Add(this.Symbols["Globals.reference1"], 0);
_symbolValues.Add(this.Symbols["Globals.rpcInvoke1"], new MyStruct("Globals.rpcInvoke1", false,
555, 666));
_symbolValues.Add(this.Symbols["Main.bool1"], true);
_symbolValues.Add(this.Symbols["Main.int1"], (short)42);
_symbolValues.Add(this.Symbols["Main.dint1"], 42);
_symbolValues.Add(this.Symbols["Main.string1"], "Hello world!");
_symbolValues.Add(this.Symbols["Main.myStruct1"], new MyStruct("Main.myStruct1", true, 42, 99));
_symbolValues.Add(this.Symbols["Main.myArray1"], new short[4, 2]);
_symbolValues.Add(this.Symbols["Main.myEnum1"], "Yellow");
_symbolValues.Add(this.Symbols["Main.myAlias1"], "Red");
_symbolValues.Add(this.Symbols["Main.pointer1"], 0);
_symbolValues.Add(this.Symbols["Main.reference1"], 0);
_symbolValues.Add(this.Symbols["Main.rpcInvoke1"], new MyStruct("Main.rpcInvoke1", false, 555, 6
66));

return this;
}

/// <summary>
/// Called when an ADS Read State indication is received.
/// </summary>
/// <param name="sender">The sender address of the request.</param>
/// <param name="invokeId">The invoke identifier of the request.</param>
/// <param name="cancel">The cancellation token.</param>
/// <returns>A task that represents the asynchronous <see cref="M:TwinCAT.Ads.Server.AdsServer.R
eadDeviceStateIndicationAsync(TwinCAT.Ads.AmsAddress, System.UInt32, System.Threading.CancellationToker
n)" /> operation. The <see cref="T:System.Threading.Tasks.Task`1" /
> parameter contains the <see cref="T:TwinCAT.Ads.AdsErrorCode" /> as
/// <see cref="P:System.Threading.Tasks.Task`1.Result" />.</returns>
/// <remarks>Overwrite this method in derived classes to react on ADS Read State indications.
/// The default implementation replies with an ADS ServiceNotSupported error code (0x701).</
remarks>
protected override Task<ResultReadDeviceState> OnReadDeviceStateAsync(AmsAddress sender, uint in
vokeId, CancellationTokens cancel)
{
    AdsState adsState = AdsState.Run;
    ushort deviceState = 0;
    StateInfo state = new StateInfo(adsState, deviceState);
    ResultReadDeviceState result = ResultReadDeviceState.CreateSuccess(state);
    return Task.FromResult(result);
}

/// <summary>
/// Handler function for Reading the internal value data in raw format.
/// </summary>
/// <param name="symbol">The symbol.</param>

```

```

/// <param name="span">The span.</param>
/// <returns>AdsErrorCode.</returns>
/// <remarks>This method is called, when a Read request was received to read the symbols value.
/// Implement this handler to Read and marshal the value data.</remarks>
protected override AdsErrorCode OnReadRawValue(ISymbol symbol, Span<byte> span)
{
    object value;
    if (_symbolValues.TryGetValue(symbol, out value))
    {
        int bytes = 0;
        if (_symbolMarshaler.TryMarshal(symbol, value, span, out bytes))
            return AdsErrorCode.NoError;
        else
            return AdsErrorCode.DeviceInvalidSize;
    }
    else
        return AdsErrorCode.DeviceSymbolNotFound;
}

/// <summary>
/// Handler function for writing the symbol value data in raw format.
/// </summary>
/// <param name="symbol">The symbol.</param>
/// <param name="span">The span.</param>
/// <returns>AdsErrorCode.</returns>
/// <remarks>This method is called, when a Write request was received to overwrite the symbols v
alue.
/// Implement this handler to overwrite the Ads Servers internal value data.</remarks>
protected override AdsErrorCode OnWriteRawValue(ISymbol symbol, ReadOnlySpan<byte> span)
{
    object value;
    _symbolMarshaler.Unmarshal(symbol, span, null, out value);
    SetValue(symbol, value);
    return AdsErrorCode.NoError;
}

/// <summary>
/// Handler function to store a new Symbol value within internal caches of the <see cref="T:Twin
CAT.Ads.Server.AdsSymbolicServer" />.
/// </summary>
/// <param name="symbol">The symbol.</param>
/// <param name="value">The value.</param>
/// <param name="valueChanged">if set to <c>true</c>, the value was changed.</param>
/// <returns>AdsErrorCode.</returns>
protected override AdsErrorCode OnSetValue(ISymbol symbol, object value, out bool valueChanged)
{
    object oldValue = null;
    valueChanged = false;
    if (_symbolValues.TryGetValue(symbol, out oldValue))
    {
        if (!value.Equals(oldValue))
        {
            valueChanged = true;
        }
        return AdsErrorCode.NoError;
    }
    return AdsErrorCode.DeviceSymbolNotFound;
}

/// <summary>
/// Handler function to determine the (stored) value of the symbol from the internal caches of t
he <see cref="T:TwinCAT.Ads.Server.AdsSymbolicServer" />.
/// </summary>
/// <param name="symbol">The symbol.</param>
/// <returns>System.Object.</returns>
protected override AdsErrorCode OnGetValue(ISymbol symbol, out object value)
{
    if (_symbolValues.TryGetValue(symbol, out value))
    {
        return AdsErrorCode.NoError;
    }
    return AdsErrorCode.DeviceSymbolNotFound;
}

/// <summary>
/// Handler function called when an RpcInvoke occurs.
/// </summary>
/// <param name="structInstance">The structure instance.</param>
/// <param name="method">The method.</param>

```

```

    /// <param name="parameterValues">The parameter values.</param>
    /// <param name="returnValue">The return value.</param>
    /// <returns>AdsErrorCode.</returns>
    /// <remarks>Overwrite this method to react on RpcInvokes within your custom <see cref="AdsSymbolicServer" />.
    /// The default implementation returns <see cref="AdsErrorCode.DeviceServiceNotSupported" />.</remarks>
    protected override AdsErrorCode OnRpcInvoke(IInterfaceInstance structInstance, IRpcMethod method, object[] parameterValues, out object returnValue)
    {
        // Here we implement or handler for the RPC Call.

        object val;
        // Select the right RpcStructInstance and get its value object.
        if (_symbolValues.TryGetValue(structInstance, out val))
        {
            MyStruct myStructValue = val as MyStruct;

            if (myStructValue != null)
            {
                // For demo simplification, we choose the Method simply by name.
                // This could be done in a more generic way, e.g with Reflection or whatever custom infrastructure.
                switch (method.Name)
                {
                    case "Method1":
                    {
                        returnValue = myStructValue.Method1((short)parameterValues[0], (short)parameterValues[1]);
                        return AdsErrorCode.NoError;
                    }
                    case "Method2":
                    {
                        returnValue = myStructValue.Method2((short)parameterValues[0], out var out1);
                        parameterValues[1] = out1;
                        return AdsErrorCode.NoError;
                    }
                    default:
                        returnValue = null;
                        return AdsErrorCode.DeviceServiceNotSupported;
                }
            }
            else
            {
                returnValue = null;
                return AdsErrorCode.DeviceServiceNotSupported;
            }
        }
        else
        {
            returnValue = null;
            return AdsErrorCode.DeviceServiceNotSupported;
        }
    }
}

// Necessary helper struct to use the .NET Default Interop marshaler which is used by the
// SymbolicAnyTypeMarshaler to Marshal/Unmarshal Struct values.
// The Layout must exactly map the 'dtMyStruct' DataType definition, so that
// the .NET Interop default marshaler is able to 'blit' the value in its own data buffers.
// MyStruct also implements the RpcInvoke Methods in this example.
[StructLayout(LayoutKind.Explicit, CharSet = CharSet.Unicode, Pack = 1)]
public class MyStruct
{
    // Constructor
    public MyStruct(string name, bool a, short b, int c)
    {
        this.name = name;
        this.a = a;
        this.b = b;
        this.c = c;
    }

    [FieldOffset(0)]
    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = 81)]
    public string name;
    [FieldOffset(162)]
    [MarshalAs(UnmanagedType.U1)] // Boolean is Marshaled as UnmanagedType.I4 otherwise
    public bool a;
    [FieldOffset(163)]

```



```
// [MarshalAs(UnmanagedType.I2)] (Default)
public short b;
[FieldOffset(165)]
// [MarshalAs(UnmanagedType.I4)] (Default)
public int c;

// Definition of the RpcMethods

public short Method1(short i1, short i2)
{
// Just return the addition of both inputs
return (short)(i1 + i2);
}
public short Method2(short i1, out short i2)
{
i2 = (short)(i1 + 1);
return (short)(i1 + 2);
}
}
```

Reference

[TwinCAT.Ads.Server Namespace \[▶ 1374\]](#)

[TwinCAT.Ads.Server.AdsServer \[▶ 1375\]](#)

6.4.10.1 AdsSymbolicServer Properties

The [AdsSymbolicServer \[▶ 1468\]](#) type exposes the following members.

Properties

Name	Description
AmsServer [▶ 1384]	Gets the the internal AmsServer [▶ 1384] object. (Inherited from AdsServer [▶ 1375] .)
DataTypes [▶ 1482]	Gets the data types
DefaultValueEncoding [▶ 1482]	Gets/Sets the default value encoding.
InstancePathEncoding [▶ 1483]	Gets or sets the instance path encoding (Default is the Windows CodePage)
IsConnected [▶ 1384]	Gets a value indicating whether AdsServer [▶ 1375] is connected. (Inherited from AdsServer [▶ 1375] .)
IsDisconnecting [▶ 1384]	Indicates, that the AdsServer [▶ 1375] is actually disconnecting. (Inherited from AdsServer [▶ 1375] .)
IsDisposed [▶ 1385]	Gets a value indicating whether this instance is disposed. (Inherited from AdsServer [▶ 1375] .)
IsDynamicPort [▶ 1387]	Gets a value indicating this AdsServer [▶ 1375] has a dynamic/unfixed port. (Inherited from AdsServer [▶ 1375] .)
Logger [▶ 1385]	Gets the logger object. (Inherited from AdsServer [▶ 1375] .)
PlatformPointerSize [▶ 1483]	Gets the Platform pointer size of this Server
RootNamespaceName [▶ 1484]	Gets the name of the root namespace
ServerAddress [▶ 1386]	The AMS address of this server. (Inherited from AdsServer [▶ 1375] .)

	Name	Description
	ServerName [▶ 1386]	Gets the name of the server. (Inherited from AdsServer [▶ 1375] .)
	ServerPort [▶ 1387]	Gets the server port. (Inherited from AdsServer [▶ 1375] .)
	ServerVersion [▶ 1387]	Gets the Version of the Server. (Inherited from AdsServer [▶ 1375] .)
	Symbols [▶ 1484]	Gets the symbols.

Reference

[AdsSymbolicServer Class](#) [\[▶ 1468\]](#)

[TwinCAT.Ads.Server Namespace](#) [\[▶ 1374\]](#)

6.4.10.1.1 AdsSymbolicServer.DataTypes Property

Gets the data types

Namespace: [TwinCAT.Ads.Server](#) [\[▶ 1374\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataValueCollection<IDataType> DataTypes { get; }
```

Property Value

Type: [IDataValueCollection](#) [\[▶ 2486\]](#).[IDataType](#) [\[▶ 2475\]](#).

The data types.

Implements

[ISymbolServer.DataTypes](#) [\[▶ 2720\]](#)

Remarks

This property reads the DataTypes synchronously, if the data is not available yet. For performance reasons, the asynchronous counterpart [GetDataTypesAsync\(CancellationToken\)](#) [\[▶ 2722\]](#) should be preferred for the first call.

Reference

[AdsSymbolicServer Class](#) [\[▶ 1468\]](#)

[TwinCAT.Ads.Server Namespace](#) [\[▶ 1374\]](#)

[ISymbolServer.GetDataTypesAsync\(CancellationToken\)](#) [\[▶ 2722\]](#)

6.4.10.1.2 AdsSymbolicServer.DefaultValueEncoding Property

Gets/Sets the default value encoding.

Namespace: [TwinCAT.Ads.Server](#) [\[▶ 1374\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Encoding DefaultValueEncoding { get; set; }
```

Property Value

Type: [Encoding](#)

The default value encoding.

Implements

[ISymbolServer.DefaultValueEncoding](#) [[▶ 2721](#)]

Remarks

This property must be set/changed before the Server is connected to have an effect!

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.1.3 **AdsSymbolicServer.InstancePathEncoding Property**

Gets or sets the instance path encoding (Default is the Windows CodePage)

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Encoding InstancePathEncoding { get; set; }
```

Property Value

Type: [Encoding](#)

The instance path encoding.

Remarks

This property must be set/changed before the Server is connected to have an effect!

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.1.4 **AdsSymbolicServer.PlatformPointerSize Property**

Gets the Platform pointer size of this Server

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int PlatformPointerSize { get; }
```

Property Value

Type: [Int32](#)

The size of the platform pointer.

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.1.5 **AdsSymbolicServer.RootNamespaceName** Property

Gets the name of the root namespace

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string RootNamespaceName { get; }
```

Property Value

Type: [String](#)

The namespace.

Implements

[ISymbolProvider.RootNamespaceName](#) [[▶ 2718](#)]

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.1.6 **AdsSymbolicServer.Symbols** Property

Gets the symbols.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISymbolCollection<ISymbol> Symbols { get; }
```

Property Value

Type: [ISymbolCollection](#) [[▶ 2700](#)].[ISymbol](#) [[▶ 2691](#)].

The symbols.

Implements

[ISymbolServer.Symbols](#) [[▶ 2721](#)]

Remarks

This property reads the Symbol information synchronously, if the data is not available yet. For performance reasons, the asynchronous counterpart [GetSymbolsAsync\(CancellationToken\)](#) [[▶ 2723](#)] should be preferred for the first call.

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

[ISymbolServer.GetSymbolsAsync\(CancellationToken\)](#) [[▶ 2723](#)]

6.4.10.2 AdsSymbolicServer Methods

The [AdsSymbolicServer](#) [[▶ 1468](#)] type exposes the following members.

Methods

	Name	Description
	AddDeviceNotificationRequest [▶ 1393]	Sends an ADS Add Device Notification request (synchronous). (Inherited from AdsServer [▶ 1375].)
	AddDeviceNotificationRequestAsync [▶ 1394]	Sends an ADS Add Device Notification request (async) (Inherited from AdsServer [▶ 1375].)
	AddDeviceNotificationResponseAsync [▶ 1395]	Sends an ADS Add Device Notification response. (Inherited from AdsServer [▶ 1375].)
	ConnectServer [▶ 1396]	Connect this ADS server to the local ADS router. (Inherited from AdsServer [▶ 1375].)
	ConnectServerAndWaitAsync [▶ 1396]	Registers the AdsServer [▶ 1375] at the router asynchronously. (Inherited from AdsServer [▶ 1375].)
	DeleteDeviceNotificationRequest [▶ 1398]	Sends an ADS Delete Device Notification request (synchronous). (Inherited from AdsServer [▶ 1375].)
	DeleteDeviceNotificationRequestAsync [▶ 1399]	Sends an ADS Delete Device Notification request (async). (Inherited from AdsServer [▶ 1375].)
	DeleteDeviceNotificationResponseAsync [▶ 1400]	Sends an ADS Delete Device Notification response. (Inherited from AdsServer [▶ 1375].)

	Name	Description
	DeviceNotificationRequestAsync [▶ 1400]	Sends an ADS Device Notification request asynchronously (Inherited from AdsServer [▶ 1375].)
	DeviceNotificationRequestSync [▶ 1401]	Sends an ADS Device Notification request (sync) (Inherited from AdsServer [▶ 1375].)
	Disconnect [▶ 1402]	Disconnects this ADS server from the local ADS router. (Inherited from AdsServer [▶ 1375].)
	Dispose. [▶ 1403]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from AdsServer [▶ 1375].)
	Dispose(Boolean) [▶ 1403]	Releases unmanaged and - optionally - managed resources. (Inherited from AdsServer [▶ 1375].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize [▶ 1403]	Finalizes an instance of the AdsServer [▶ 1375] class. (Inherited from AdsServer [▶ 1375].)
	FireNotificationAsync [▶ 1490]	Fires a single notification to the specified Address.
	FireNotifications [▶ 1490]	Fires outstanding cyclic notifications.
	FireNotificationsAsync [▶ 1491]	Check internal tables for outstanding cyclic Notifications and fire them asynchronously.
	GetDataTypesAsync [▶ 1491]	Gets the data types asynchronously.
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetServerName [▶ 1441]	Gets the name of the server. (Inherited from AdsServer [▶ 1375].)
	GetSymbolsAsync [▶ 1492]	Gets the symbols asynchronously
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	HasValueChanged [▶ 1493]	Determines whether the value of the Symbol has changed.
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnAddDeviceNotificationAsync [▶ 1493]	Called when an ADS Add Device Notification indication is received. (Overrides AdsServer.OnAddDeviceNotificationAsync(AmsAddress, UInt32, UInt32, UInt32, Int32, AmsAddress, NotificationSettings, CancellationToken) [▶ 1429].)
	OnAddDeviceNotificationConfirmationAsync [▶ 1404]	Called when an ADS Add Device Notification confirmation is received. (Inherited from AdsServer [▶ 1375].)
	OnBeforeConnected [▶ 1494]	Handler function that is called, when the AdsServer [▶ 1375] is connected, but before calling OnConnected . [▶ 1405]. (Overrides AdsServer.OnBeforeConnected . [▶ 1430].)
	OnConnected [▶ 1495]	Handler function that is called, when the AdsServer [▶ 1375] is connected. (Overrides AdsServer.OnConnected . [▶ 1405].)

	Name	Description
	OnCreateSymbols [▶ 1509]	Handler function to create the Initial Symbols of the server.
	OnDeleteDeviceNotificationAsync [▶ 1495]	Called when a DeleteDeviceNotification request is indicated. (Overrides AdsServer.OnDeleteDeviceNotificationAsync(AmsAddress, UInt32, UInt32, CancellationToken) [▶ 1430].)
	OnDeleteDeviceNotificationConfirmationAsync [▶ 1431]	Called when an ADS Delete Device Notification confirmation is received. (Inherited from AdsServer [▶ 1375].)
	OnDisconnect [▶ 1496]	Called when the AdsServer [▶ 1375] is about to be disconnected. (Overrides AdsServer.OnDisconnect. [▶ 1405].)
	OnFireNotifications [▶ 1496]	Handler function for firing notifications to the specified targets.
	OnFireNotificationsAsync [▶ 1497]	Handler function for firing notifications to the specified targets.
	OnGetValue [▶ 1497]	Handler function to determine the (stored) value of the symbol from the internal caches of the AdsSymbolicServer [▶ 1468].
	OnReadAsync [▶ 1498]	Called when an ADS Read indication is received. (Overrides AdsServer.OnReadAsync(AmsAddress, UInt32, UInt32, UInt32, Int32, CancellationToken) [▶ 1432].)
	OnReadConfirmationAsync [▶ 1433]	Called when an ADS Read confirmation is received. (Inherited from AdsServer [▶ 1375].)
	OnReadDeviceInfoConfirmationAsync [▶ 1434]	Called when an ADS Read Device Info confirmation is received. (Inherited from AdsServer [▶ 1375].)
	OnReadDeviceStateAsync [▶ 1435]	Called when an ADS Read State indication is received. (Inherited from AdsServer [▶ 1375].)
	OnReadDeviceStateConfirmationAsync [▶ 1435]	Called when an ADS Read State confirmation is received. (Inherited from AdsServer [▶ 1375].)
	OnReadIglloAsync [▶ 1499]	Handler function for handling pure IG/IO Read indications
	OnReadRawValue [▶ 1500]	Handler function for Reading the internal value data in raw format.
	OnReadWriteAsync [▶ 1500]	Called when an ADS Read Write indication is received. (Overrides AdsServer.OnReadWriteAsync(AmsAddress, UInt32, UInt32, UInt32, Int32, ReadOnlyMemory<Byte>, CancellationToken) [▶ 1436].)
	OnReadWriteConfirmationAsync [▶ 1437]	Called when an ADS Read Write confirmation is received. (Inherited from AdsServer [▶ 1375].)
	OnReadWriteIglloAsync [▶ 1501]	Handler function for handling of pure IG/IO calls.

Name	Description
OnRouterNotification [▶ 1405]	Handler Function for a Router Notification. (Inherited from AdsServer [▶ 1375].)
OnRpclInvoke [▶ 1502]	Handler function called when an RpclInvoke occurs.
OnServerConnectionStateChanged [▶ 1406]	Handles the ServerConnectionStateChanged [▶ 1442] event. (Inherited from AdsServer [▶ 1375].)
OnSetValue [▶ 1503]	Handler function to store a new Symbol value within internal caches of the AdsSymbolicServer [▶ 1468].
OnWriteAsync [▶ 1503]	Called when an ADS Write indication is received. (Overrides AdsServer.OnWriteAsync(AmsAddress, UInt32, UInt32, UInt32, ReadOnlyMemory.Byte., CancellationToken) [▶ 1438].)
OnWriteConfirmationAsync [▶ 1439]	Called when an ADS Write confirmation is received. (Inherited from AdsServer [▶ 1375].)
OnWriteControlAsync [▶ 1440]	Called when an ADS Write Control indication is received. (Inherited from AdsServer [▶ 1375].)
OnWriteControlConfirmationAsync [▶ 1441]	Called when an ADS Write Control confirmation is received. (Inherited from AdsServer [▶ 1375].)
OnWriteIglloAsync [▶ 1504]	Handler function for handling of pure Write IG/IO Indications.
OnWriteRawValue [▶ 1505]	Handler function for writing the symbol value data in raw format.
ReadDeviceInfoRequestAsync [▶ 1409]	Sends an ADS Read Device Info request asynchronously (Inherited from AdsServer [▶ 1375].)
ReadDeviceInfoRequestSync [▶ 1409]	Sends an ADS Read Device Info request synchronously. (Inherited from AdsServer [▶ 1375].)
ReadDeviceInfoResponseAsync [▶ 1410]	Sends an ADS Read Device Info response. (Inherited from AdsServer [▶ 1375].)
ReadDeviceStateRequestAsync [▶ 1412]	Sends an ADS Read State request (asynchronous) (Inherited from AdsServer [▶ 1375].)
ReadDeviceStateRequestSync [▶ 1413]	Sends an ADS Read State request (synchronous) (Inherited from AdsServer [▶ 1375].)
ReadDeviceStateResponseAsync [▶ 1413]	Sends an ADS Read State response. (Inherited from AdsServer [▶ 1375].)
ReadRequest [▶ 1415]	Sends an ADS Read Request. (Inherited from AdsServer [▶ 1375].)

	Name	Description
	ReadRequestAsync [▶ 1416]	Sends an ADS Read Request asynchronously. (Inherited from AdsServer [▶ 1375].)
	ReadResponseAsync [▶ 1416]	Sends an ADS Read response. (Inherited from AdsServer [▶ 1375].)
	ReadWriteRequestAsync [▶ 1419]	Sends an ADS Read Write request. (Inherited from AdsServer [▶ 1375].)
	ReadWriteRequestSync [▶ 1419]	Sends an ADS Read Write request synchronously (Inherited from AdsServer [▶ 1375].)
	ReadWriteResponseAsync [▶ 1420]	Sends an ADS Read Write Response. (Inherited from AdsServer [▶ 1375].)
	ResetCachedSymbolicData [▶ 1506]	Resets the cached symbolic data.
	SetValue(ISymbol, Object) [▶ 1507]	Sets the value of the symbol.
	SetValue(String, Object) [▶ 1507]	Sets the value of the symbol.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryGetDataTypes [▶ 1508]	Tries to get the symbols from the device target.
	TryGetSymbols [▶ 1509]	Tries to geth the symbols from the device target.
	WriteControlRequest [▶ 1423]	Sends an ADS Write Control request (synchronous) (Inherited from AdsServer [▶ 1375].)
	WriteControlRequestAsync [▶ 1424]	Sends an ADS Write Control request (asynchronous). (Inherited from AdsServer [▶ 1375].)
	WriteControlRequestSync [▶ 1425]	Sends an ADS Write Control request (synchronous). (Inherited from AdsServer [▶ 1375].)
	WriteControlResponseAsync [▶ 1425]	Sends an ADS Write Control response. (Inherited from AdsServer [▶ 1375].)
	WriteRequest [▶ 1427]	Sends an ADS Write request synchronously. (Inherited from AdsServer [▶ 1375].)
	WriteRequestAsync [▶ 1428]	Sends an ADS Write request asynchronously. (Inherited from AdsServer [▶ 1375].)
	WriteResponseAsync [▶ 1428]	Sends an ADS Write response. (Inherited from AdsServer [▶ 1375].)

Reference

[AdsSymbolicServer Class](#) [▶ 1468]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.10.2.1 **AdsSymbolicServer.FireNotificationAsync** Method

Fires a single notification to the specified Address.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected Task<AdsErrorCode> FireNotificationAsync(
    AmsAddress address,
    uint handle,
    CancellationToken cancel
)
```

Parameters

address	Type: TwinCAT.Ads.AmsAddress [▶ 752] The address.
handle	Type: System.UInt32 The handle.
cancel	Type: System.Threading.CancellationToken The cancel.

Return Value

Type: [Task.AdsErrorCode](#) [[▶ 664](#)].
AdsErrorCode.

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.2.2 **AdsSymbolicServer.FireNotifications** Method

Fires outstanding cyclic notifications.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode[] FireNotifications(
    long tick
)
```

Parameters

tick	Type: System.Int64 The tick.
------	---

Return Value

Type: [.AdsErrorCode](#) [[▶ 664](#)].
AdsErrorCode[].

Reference[AdsSymbolicServer Class \[► 1468\]](#)[TwinCAT.Ads.Server Namespace \[► 1374\]](#)**6.4.10.2.3 AdsSymbolicServer.FireNotificationsAsync Method**

Check internal tables for outstanding cyclic Notifications and fire them asynchronously.

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)**Assembly:** TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public Task<AdsErrorCode[]> FireNotificationsAsync(
    long tick,
    CancellationToken cancel
)
```

Parameters

tick	Type: System.Int64 The tick.
cancel	Type: System.Threading.CancellationToken The cancel.

Return ValueType: [Task..AdsErrorCode \[► 664\]](#)..

Task.

Reference[AdsSymbolicServer Class \[► 1468\]](#)[TwinCAT.Ads.Server Namespace \[► 1374\]](#)**6.4.10.2.4 AdsSymbolicServer.GetDataTypesAsync Method**

Gets the data types asynchronously.

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)**Assembly:** TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public Task<ResultDataTypes> GetDataTypesAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultDataTypes](#) [▸ 2872].

A task that represents the asynchronous 'GetDataTypes' operation. The [ResultDataTypes](#) [▸ 2872] parameter contains the data types ([DataTypes](#) [▸ 2874]) and the [ErrorCode](#) [▸ 1120] after execution.

Implements

[ISymbolServer.GetDataTypesAsync\(CancellationTokens\)](#) [▸ 2722]

Exceptions

Exception	Condition
NotImplementedException	

Reference

[AdsSymbolicServer Class](#) [▸ 1468]

[TwinCAT.Ads.Server Namespace](#) [▸ 1374]

[ISymbolServer.DataTypes](#) [▸ 2720]

6.4.10.2.5 AdsSymbolicServer.GetSymbolsAsync Method

Gets the symbols asynchronously

Namespace: [TwinCAT.Ads.Server](#) [▸ 1374]

Assembly: [TwinCAT.Ads.SymbolicServer](#) (in [TwinCAT.Ads.SymbolicServer.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultSymbols> GetSymbolsAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultSymbols](#) [▸ 2880].

A task that represents the asynchronous 'GetDataTypes' operation. The [ResultSymbols](#) [▸ 2880] parameter contains the data types ([Symbols](#) [▸ 2886]) and the [ErrorCode](#) [▸ 1120] after execution.

Implements

[ISymbolServer.GetSymbolsAsync\(CancellationTokens\)](#) [▸ 2723]

Exceptions

Exception	Condition
NotImplementedException	

Reference

[AdsSymbolicServer Class](#) [▶ 1468]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

[ISymbolServer.Symbols](#) [▶ 2721]

6.4.10.2.6 AdsSymbolicServer.HasValueChanged Method

Determines whether the value of the Symbol has changed.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual bool HasValueChanged(
    ISymbol symbol
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol.
--------	--

Return Value

Type: [Boolean](#)

true if the value has changed; otherwise, false.

Remarks

This Method is used internally to find out, if a symbol was changed since the last notification trigger. Collecting Notifications and send them in cycles enforces that the information about changed values is cached somehow. In the case of the [AdsSymbolicServer](#) [▶ 1468] Symbols with outstanding notifications will have a flag in the internal notification tables.

Reference

[AdsSymbolicServer Class](#) [▶ 1468]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.10.2.7 AdsSymbolicServer.OnAddDeviceNotificationAsync Method

Called when an ADS Add Device Notification indication is received.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected override Task<ResultHandle> OnAddDeviceNotificationAsync (
    AmsAddress sender,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    int dataLength,
    AmsAddress receiver,
    NotificationSettings settings,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [▶ 752] The sender address of the request.
invokeld	Type: System.UInt32 The invoke identifier of the request.
indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index offset of the requested ADS service
dataLength	Type: System.Int32 Number of bytes to be transmitted
receiver	Type: TwinCAT.Ads.AmsAddress [▶ 752] The receiver of the notifications (the requester of this message)
settings	Type: TwinCAT.Ads.NotificationSettings [▶ 1106] The Notification settings.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultHandle](#) [▶ 1136].

A task that represents the asynchronous 'AddDeviceNotificationIndication' operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [▶ 664] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Add Device Notification indications. The default implementation replies with an ADS ServiceNotSupported error code (0x701).

Reference

[AdsSymbolicServer Class](#) [▶ 1468]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.10.2.8 AdsSymbolicServer.OnBeforeConnected Method

Handler function that is called, when the [AdsServer](#) [▶ 1375] is connected, but before calling [OnConnected](#). [▶ 1405].

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: [TwinCAT.Ads.SymbolicServer](#) (in [TwinCAT.Ads.SymbolicServer.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
protected override void OnBeforeConnected()
```

Reference

[AdsSymbolicServer Class \[► 1468\]](#)

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.10.2.9 AdsSymbolicServer.OnConnected Method

Handler function that is called, when the [AdsServer \[► 1375\]](#) is connected.

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected override void OnConnected()
```

Reference

[AdsSymbolicServer Class \[► 1468\]](#)

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.10.2.10 AdsSymbolicServer.OnDeleteDeviceNotificationAsync Method

Called when a DeleteDeviceNotification request is indicated.

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected override Task<ResultAds> OnDeleteDeviceNotificationAsync(
    AmsAddress sender,
    uint invokeId,
    uint notificationHandle,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752] The sender address of the request.
invokeId	Type: System.UInt32 The invoke identifier of the request.
notificationHandle	Type: System.UInt32 The notification handle.
cancel	Type: System.Threading.CancellationToken The cancellation token that can be used by other objects or threads to receive notice of cancellation.

Return Value

Type: [Task.ResultAds \[▸ 1116\]](#).
Task<ResultAds>.

Reference

[AdsSymbolicServer Class \[▸ 1468\]](#)

[TwinCAT.Ads.Server Namespace \[▸ 1374\]](#)

6.4.10.2.11 AdsSymbolicServer.OnDisconnect Method

Called when the [AdsServer \[▸ 1375\]](#) is about to be disconnected.

Namespace: [TwinCAT.Ads.Server \[▸ 1374\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
protected override bool OnDisconnect()
```

Return Value

Type: [Boolean](#)
true if the [AdsServer \[▸ 1375\]](#) is disconnected, false otherwise (was disconnected before)

Reference

[AdsSymbolicServer Class \[▸ 1468\]](#)

[TwinCAT.Ads.Server Namespace \[▸ 1374\]](#)

6.4.10.2.12 AdsSymbolicServer.OnFireNotifications Method

Handler function for firing notifications to the specified targets.

Namespace: [TwinCAT.Ads.Server \[▸ 1374\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
protected AdsErrorCode OnFireNotifications(
    AmsAddress a,
    NotificationDataSample[] stamps
)
```

Parameters

a	Type: TwinCAT.Ads.AmsAddress [▸ 752] a.
stamps	Type: .TwinCAT.Ads.Server.NotificationDataSample [▸ 1453] . The stamps.

Return Value

Type: [AdsErrorCode \[▸ 664\]](#)
AdsErrorCode.

Reference[AdsSymbolicServer Class \[► 1468\]](#)[TwinCAT.Ads.Server Namespace \[► 1374\]](#)**6.4.10.2.13 AdsSymbolicServer.OnFireNotificationsAsync Method**

Handler function for firing notifications to the specified targets.

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)**Assembly:** TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
protected Task<AdsErrorCode> OnFireNotificationsAsync(
    AmsAddress a,
    NotificationDataSample[] stamps,
    CancellationToken cancel
)
```

Parameters

a	Type: TwinCAT.Ads.AmsAddress [► 752] a.
stamps	Type: .TwinCAT.Ads.Server.NotificationDataSample [► 1453] . The stamps.
cancel	Type: System.Threading.CancellationToken The cancel.

Return Value

Type: [Task.AdsErrorCode \[► 664\]](#).
Task<AdsErrorCode>.

Reference[AdsSymbolicServer Class \[► 1468\]](#)[TwinCAT.Ads.Server Namespace \[► 1374\]](#)**6.4.10.2.14 AdsSymbolicServer.OnGetValue Method**

Handler function to determine the (stored) value of the symbol from the internal caches of the [AdsSymbolicServer \[► 1468\]](#).

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)**Assembly:** TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
protected abstract AdsErrorCode OnGetValue(
    ISymbol symbol,
    out Object?? value
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol.
value	Type: System.Object . The value.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
AdsErrorCode

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.2.15 AdsSymbolicServer.OnReadAsync Method

Called when an ADS Read indication is received.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
protected override Task<ResultReadBytes> OnReadAsync (
    AmsAddress sender,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    int readLength,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [▶ 752] The sender address of the request.
invokeld	Type: System.UInt32 The invoke identifier of the request.
indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index offset of the requested ADS service
readLength	Type: System.Int32 The number of bytes to be read
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultReadBytes](#) [[▶ 1150](#)].

A task that represents the asynchronous ReadIndicationAsync(AmsAddress, UInt32, UInt32, UInt32, Int32, CancellationToken) operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [[▶ 664](#)] as [Result](#).

Remarks

Overwrite this method in derived classes to react on ADS Read indications. The default implementation replies with an ADS ServiceNotSupported error code (0x701).

Reference

[AdsSymbolicServer Class \[► 1468\]](#)

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.10.2.16 AdsSymbolicServer.OnReadIgoAsync Method

Handler function for handling pure IG/IO Read indications

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
protected virtual Task<ResultReadBytes> OnReadIgoAsync(
    AmsAddress sender,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    int readLength,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752] The sender address of the request.
invokeld	Type: System.UInt32 The invoke identifier of the request.
indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index offset of the requested ADS service
readLength	Type: System.Int32 The number of bytes to be read
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultReadBytes \[► 1150\]](#).

A task that represents the asynchronous ReadIndicationAsync(AmsAddress, UInt32, UInt32, UInt32, Int32, CancellationToken) operation. The [Task.TResult](#) parameter contains the [AdsErrorCode \[► 664\]](#) as [Result](#).

Remarks

Overwrite this method in derived classes to react on pure IG/IO ADS Read indications.

Reference

[AdsSymbolicServer Class \[► 1468\]](#)

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.10.2.17 AdsSymbolicServer.OnReadRawValue Method

Handler function for Reading the internal value data in raw format.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected abstract AdsErrorCode OnReadRawValue (
    ISymbol symbol,
    Span<byte> span
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol.
span	Type: System.Span.Byte . The span.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

AdsErrorCode.

Remarks

This method is called, when a Read request was received to read the symbols value. Implement this handler to Read and marshal the value data.

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.2.18 AdsSymbolicServer.OnReadWriteAsync Method

Called when an ADS Read Write indication is received.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected override Task<ResultReadWriteBytes> OnReadWriteAsync (
    AmsAddress sender,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    int readLength,
    ReadOnlyMemory<byte> writeData,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [▶ 752] The sender address of the request.
--------	--

invokeld	Type: System.UInt32 The invoke identifier of the request.
indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index offset of the requested ADS service
readLength	Type: System.Int32 Number of bytes to be read
writeData	Type: System.ReadOnlyMemory.Byte . The data to be written
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultReadWriteBytes](#) [▶ 1170].

A task that represents the asynchronous 'ReadWriteIndication' operation. The [Task.TResult](#). parameter contains the [AdsErrorCode](#) [▶ 664] as Result.

Remarks

Overwrite this method in derived classes to react on ADS Read Write indications. The default implementation replies with an ADS ServiceNotSupported error code (0x701).

Reference

[AdsSymbolicServer Class](#) [▶ 1468]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.10.2.19 AdsSymbolicServer.OnReadWriteIgIoAsync Method

Handler function for handling of pure IG/IO calls.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<ResultReadWriteBytes> OnReadWriteIgIoAsync(
    AmsAddress sender,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    int readLength,
    ReadOnlyMemory<byte> writeData,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [▶ 752] The sender address of the request.
invokeld	Type: System.UInt32 The invoke identifier of the request.
indexGroup	Type: System.UInt32 The index group of the requested ADS service

indexOffset	Type: System.UInt32 The index offset of the requested ADS service
readLength	Type: System.Int32 Number of bytes to be read
writeData	Type: System.ReadOnlyMemory.Byte . The data to be written
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultReadWriteBytes](#) [► 1170].

A task that represents the asynchronous 'OnReadWriteIgcloAsync' operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [► 664] as [Result](#).

Exceptions

Exception	Condition
NotImplementedException	

Remarks

Overwrite this method in derived classes to react on pure IG/IO ADS ReadWrite indications. To send the ADS answer/response back to the sender, the base class implementation must be called after

Reference

[AdsSymbolicServer Class](#) [► 1468]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.10.2.20 AdsSymbolicServer.OnRpcInvoke Method

Handler function called when an RpcInvoke occurs.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: [TwinCAT.Ads.SymbolicServer](#) (in [TwinCAT.Ads.SymbolicServer.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual AdsErrorCode OnRpcInvoke(
    IInterfaceInstance structInstance,
    IRpcMethod method,
    Object[] values,
    out Object?? returnValue
)
```

Parameters

structInstance	Type: TwinCAT.TypeSystem.IInterfaceInstance [► 3014] The structure instance.
method	Type: TwinCAT.TypeSystem.IRpcMethod [► 2625] The method.
values	Type: .System.Object . The values.

returnValue	Type: System.Object . The return value.
-------------	--

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
AdsErrorCode.

Remarks

Overwrite this method to react on RpcInvokes within your custom [AdsSymbolicServer](#) [[▶ 1468](#)]. The default implementation returns [DeviceServiceNotSupported](#) [[▶ 664](#)].

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.2.21 AdsSymbolicServer.OnSetValue Method

Handler function to store a new Symbol value within internal caches of the [AdsSymbolicServer](#) [[▶ 1468](#)].

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected abstract AdsErrorCode OnSetValue(
    ISymbol symbol,
    Object value,
    out bool valueChanged
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol.
value	Type: System.Object The value.
valueChanged	Type: System.Boolean . if set to true, the value was changed.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
AdsErrorCode.

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.2.22 AdsSymbolicServer.OnWriteAsync Method

Called when an ADS Write indication is received.

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected override Task<ResultWrite> OnWriteAsync(
    AmsAddress sender,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory<byte> writeData,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752] The sender address of the request.
invokeld	Type: System.UInt32 The invoke identifier of the request.
indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index offset of the requested ADS service
writeData	Type: System.ReadOnlyMemory.Byte. The data to be written
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite \[► 1187\]](#).

A task that represents the asynchronous WriteIndicationAsync(AmsAddress, UInt32, UInt32, UInt32, ReadOnlyMemory.Byte., CancellationToken) operation. The [Task.TResult.](#) parameter contains the [AdsErrorCode \[► 664\]](#) as Result.

Remarks

Overwrite this method in derived classes to react on ADS Write indications. The default implementation replies with an ADS ServiceNotSupported error code (0x701).

Reference

[AdsSymbolicServer Class \[► 1468\]](#)

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.10.2.23 AdsSymbolicServer.OnWriteIgIoAsync Method

Handler function for handling of pure Write IG/IO Indications.

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<ResultWrite> OnWriteIgIoAsync (
    AmsAddress sender,
    uint invokeId,
    uint indexGroup,
    uint indexOffset,
    ReadOnlyMemory<byte> writeData,
    CancellationToken cancel
)
```

Parameters

sender	Type: TwinCAT.Ads.AmsAddress [► 752] The sender address of the request.
invokeld	Type: System.UInt32 The invoke identifier of the request.
indexGroup	Type: System.UInt32 The index group of the requested ADS service
indexOffset	Type: System.UInt32 The index offset of the requested ADS service
writeData	Type: System.ReadOnlyMemory.Byte . The data to be written
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWrite](#) [► 1187].

A task that represents the asynchronous `WriteIndicationAsync(AmsAddress, UInt32, UInt32, UInt32, ReadOnlyMemory.Byte., CancellationToken)` operation. The [Task.TResult](#) parameter contains the [AdsErrorCode](#) [► 664] as [Result](#).

Remarks

Overwrite this method in derived classes to react on pure ADS Write indications by IG/IO. The Default implementation returns a `DeviceServiceNotSupported` error.

Reference

[AdsSymbolicServer Class](#) [► 1468]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.10.2.24 AdsSymbolicServer.OnWriteRawValue Method

Handler function for writing the symbol value data in raw format.

Namespace: [TwinCAT.Ads.Server](#) [► 1374]

Assembly: `TwinCAT.Ads.SymbolicServer` (in `TwinCAT.Ads.SymbolicServer.dll`) Version: 6.0.328+39e3229

Syntax

C#

```
protected abstract AdsErrorCode OnWriteRawValue (
    ISymbol symbol,
    ReadOnlySpan<byte> span
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▸ 2691] The symbol.
span	Type: System.ReadOnlySpan.Byte . The span.

Return Value

Type: [AdsErrorCode](#) [[▸ 664](#)]
AdsErrorCode.

Remarks

This method is called, when a Write request was received to overwrite the symbols value. Implement this handler to overwrite the Ads Servers internal value data.

Reference

[AdsSymbolicServer Class](#) [[▸ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▸ 1374](#)]

6.4.10.25 AdsSymbolicServer.ResetCachedSymbolicData Method

Resets the cached symbolic data.

Namespace: [TwinCAT.Ads.Server](#) [[▸ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public void ResetCachedSymbolicData()
```

Implements

[ISymbolServer.ResetCachedSymbolicData](#). [[▸ 2724](#)]

Reference

[AdsSymbolicServer Class](#) [[▸ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▸ 1374](#)]

6.4.10.26 AdsSymbolicServer.SetValue Method**Overload List**

	Name	Description
	SetValue(ISymbol, Object) [▸ 1507]	Sets the value of the symbol.
	SetValue(String, Object) [▸ 1507]	Sets the value of the symbol.

Reference

[AdsSymbolicServer Class](#) [[▸ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.10.2.26.1 **AdsSymbolicServer.SetValue Method (ISymbol, Object)**

Sets the value of the symbol.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode SetValue(  
    ISymbol symbol,  
    Object value  
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol.
value	Type: System.Object The value.

Return Value

Type: [AdsErrorCode](#) [▶ 664]

Remarks

Call this if the Server has to store a new Value of the symbol internally.

Reference

[AdsSymbolicServer Class](#) [▶ 1468]

[SetValue Overload](#) [▶ 1506]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.10.2.26.2 **AdsSymbolicServer.SetValue Method (String, Object)**

Sets the value of the symbol.

Namespace: [TwinCAT.Ads.Server](#) [▶ 1374]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode SetValue(  
    string instancePath,  
    Object value  
)
```

Parameters

instancePath	Type: System.String The instance path.
--------------	---

value	Type: System.Object The value.
-------	---

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

Exceptions

Exception	Condition
SymbolException [▶ 2933]	Symbol '{instancePath}' not found! - null

Remarks

Call this if the Server has to store a new Value of the symbol internally.

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[SetValue Overload](#) [[▶ 1506](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.2.27 AdsSymbolicServer.TryGetDataTypes Method

Tries to get the symbols from the device target.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public AdsErrorCode TryGetDataTypes (
    out IDataValueCollection<IDataType>? dataTypes
)
```

Parameters

dataTypes	Type: TwinCAT.TypeSystem.IDataValueCollection [▶ 2486]. IDataType [▶ 2475]. The data types.
-----------	--

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

AdsErrorCode.

Implements

[ISymbolServer.TryGetDataTypes\(IDataValueCollection.IDataType..\)](#) [[▶ 2723](#)]

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.2.28 **AdsSymbolicServer.TryGetSymbols Method**

Tries to geth the symbols from the device target.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryGetSymbols(  
    out ISymbolCollection<ISymbol>? symbols  
)
```

Parameters

symbols	Type: TwinCAT.TypeSystem.ISymbolCollection [▶ 2700]. ISymbol [▶ 2691]. The symbols.
---------	--

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

AdsErrorCode.

Implements

[ISymbolServer.TryGetSymbols\(ISymbolCollection.ISymbol..\)](#) [[▶ 2724](#)]

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.2.29 **AdsSymbolicServer.OnCreateSymbols Method**

Handler function to create the Initial Symbols of the server.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual void OnCreateSymbols()
```

Remarks

Override this to create the initial symbols during AdsServer Connection

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.3 **AdsSymbolicServer Events**

The [AdsSymbolicServer](#) [[▶ 1468](#)] type exposes the following members.

Events

	Name	Description
	ServerConnectionStateChanged [▶ 1442]	The connection status has changed (Inherited from AdsServer [▶ 1375].)

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.4 AdsSymbolicServer Fields

The [AdsSymbolicServer](#) [[▶ 1468](#)] type exposes the following members.

Fields

	Name	Description
	handleTable [▶ 1510]	The handle table (InstancePath --> Symbol handle)
	notificationTable [▶ 1511]	The notification table
	notificationTrigger [▶ 1511]	The notification trigger Source
	serverVersion [▶ 1443]	The version of the AdsServer (Inherited from AdsServer [▶ 1375].)
	symbolFactory [▶ 1511]	The symbol factory
	symbolNameMarshaller [▶ 1512]	The SymbolName marshaler
	symbolVersion [▶ 1512]	The symbol version as respond by SymbolVersion [▶ 1231]

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.4.1 AdsSymbolicServer.handleTable Field

The handle table (InstancePath --> Symbol handle)

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
protected HandleTable<string??> handleTable
```

Field Value

Type: [HandleTable.String](#).

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.4.2 AdsSymbolicServer.notificationTable Field

The notification table

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: [TwinCAT.Ads.SymbolicServer](#) (in [TwinCAT.Ads.SymbolicServer.dll](#)) Version: 6.0.328+39e3229

Syntax**C#**

```
protected NotificationTable? notificationTable
```

Field Value

Type: [NotificationTable](#)

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.4.3 AdsSymbolicServer.notificationTrigger Field

The notification trigger Source

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: [TwinCAT.Ads.SymbolicServer](#) (in [TwinCAT.Ads.SymbolicServer.dll](#)) Version: 6.0.328+39e3229

Syntax**C#**

```
protected NotificationTriggerSource? notificationTrigger
```

Field Value

Type: [NotificationTriggerSource](#) [[▶ 1517](#)]

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.4.4 AdsSymbolicServer.symbolFactory Field

The symbol factory

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: [TwinCAT.Ads.SymbolicServer](#) (in [TwinCAT.Ads.SymbolicServer.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
protected ServerSymbolFactory? symbolFactory
```

Field Value

Type: [ServerSymbolFactory](#) [[▶ 3242](#)]

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.4.5 AdsSymbolicServer.symbolNameMarshaler Field

The SymbolName marshaler

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected StringMarshaler? symbolNameMarshaler
```

Field Value

Type: [StringMarshaler](#)

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.10.4.6 AdsSymbolicServer.symbolVersion Field

The symbol version as respond by [SymbolVersion](#) [[▶ 1231](#)]

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected uint symbolVersion
```

Field Value

Type: [UInt32](#)

Reference

[AdsSymbolicServer Class](#) [[▶ 1468](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.11 BaseTickTrigger Class

Class NotificationTimer. Implements the [INotificationTrigger](#) [[▶ 1516](#)]

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.Server.BaseTickTrigger](#)

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public class BaseTickTrigger : INotificationTrigger,
    IObservable<long>
```

The BaseTickTrigger type exposes the following members.

Constructors

	Name	Description
	BaseTickTrigger [▶ 1513]	Initializes a new instance of the BaseTickTrigger class.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Subscribe [▶ 1514]	Subscribes the specified observer.
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Remarks

This observable creates a BaseTick Trigger for the [AdsSymbolicServer](#) [[▶ 1468](#)]. The observable long parameter is just a tick counter. The fired ticks are aquidistant as defined in the constructor.

Reference

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

[TwinCAT.Ads.Server.INotificationTrigger](#) [[▶ 1516](#)]

6.4.11.1 BaseTickTrigger Constructor

Initializes a new instance of the [BaseTickTrigger](#) [[▶ 1513](#)] class.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public BaseTickTrigger(
    TimeSpan baseTick
)
```

Parameters

baseTick	Type: System.TimeSpan The base tick.
----------	---

Reference

[BaseTickTrigger Class](#) [► [1513](#)]

[TwinCAT.Ads.Server Namespace](#) [► [1374](#)]

6.4.11.2 BaseTickTrigger Methods

The [BaseTickTrigger](#) [► [1513](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Subscribe [► 1514]	Subscribes the specified observer.
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[BaseTickTrigger Class](#) [► [1513](#)]

[TwinCAT.Ads.Server Namespace](#) [► [1374](#)]

6.4.11.2.1 BaseTickTrigger.Subscribe Method

Subscribes the specified observer.

Namespace: [TwinCAT.Ads.Server](#) [► [1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDisposable Subscribe(
    IObservable<long> observer
)
```

Parameters

observer	Type: System.IObserver.Int64 . The observer.
----------	---

Return Value

Type: [IDisposable](#)
A reference to an interface that allows observers to finish receiving notifications.

Implements

[IObservable.T..Subscribe\(IObserver.T.\)](#)

Reference

[BaseTickTrigger Class \[▶ 1513\]](#)

[TwinCAT.Ads.Server Namespace \[▶ 1374\]](#)

6.4.12 IAddNotificationTrigger Interface

Interface [IAddNotificationTrigger](#)

Namespace: [TwinCAT.Ads.Server \[▶ 1374\]](#)

Assembly: [TwinCAT.Ads.SymbolicServer \(in TwinCAT.Ads.SymbolicServer.dll\) Version: 6.0.328+39e3229](#)

Syntax

C#

```
public interface IAddNotificationTrigger
```

The [IAddNotificationTrigger](#) type exposes the following members.

Methods

	Name	Description
	Add [▶ 1516]	Adds the specified observer.

Remarks

Fluent interface to Add observable notification triggers to the implementing class.

Reference

[TwinCAT.Ads.Server Namespace \[▶ 1374\]](#)

6.4.12.1 IAddNotificationTrigger Methods

The [IAddNotificationTrigger \[▶ 1515\]](#) type exposes the following members.

Methods

	Name	Description
	Add [▶ 1516]	Adds the specified observer.

Reference[IAddNotificationTrigger Interface \[► 1515\]](#)[TwinCAT.Ads.Server Namespace \[► 1374\]](#)**6.4.12.1 IAddNotificationTrigger.Add Method**

Adds the specified observer.

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)**Assembly:** TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
IAddNotificationTrigger Add(
    INotificationTrigger observable
)
```

Parameters

observable	Type: TwinCAT.Ads.Server.INotificationTrigger [► 1516] The trigger observable.
------------	---

Return ValueType: [IAddNotificationTrigger \[► 1515\]](#)

IAddNotificationTrigger.

Reference[IAddNotificationTrigger Interface \[► 1515\]](#)[TwinCAT.Ads.Server Namespace \[► 1374\]](#)**6.4.13 INotificationTrigger Interface**

Interface INotificationTrigger

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)**Assembly:** TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public interface INotificationTrigger : IObservable<long>
```

The INotificationTrigger type exposes the following members.

Methods

	Name	Description
	Subscribe	Notifies the provider that an observer is to receive notifications. (Inherited from IObservable.Int64..)

Remarks

The notification trigger produces observable events, when the [AdsSymbolicServer \[► 1468\]](#) has to be checked for sending actual notifications. It can be realised e.g. as a base tick.

Reference

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.13.1 INotificationTrigger Methods

The [INotificationTrigger \[► 1516\]](#) type exposes the following members.

Methods

	Name	Description
	Subscribe	Notifies the provider that an observer is to receive notifications. (Inherited from IObservable.Int64..)

Reference

[INotificationTrigger Interface \[► 1516\]](#)

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.14 NotificationTriggerSource Class

Class [NotificationTriggerSource](#). Implements the [IAddNotificationTrigger \[► 1515\]](#) Implements the [IDisposable](#)

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.Server.NotificationTriggerSource](#)

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)

Assembly: [TwinCAT.Ads.SymbolicServer](#) (in [TwinCAT.Ads.SymbolicServer.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public class NotificationTriggerSource : IAddNotificationTrigger,
    IDisposable
```

The [NotificationTriggerSource](#) type exposes the following members.

Methods

	Name	Description
	Add [► 1519]	Adds the specified observer.
	Connect [► 1519]	Connects this instance (Start observing notification triggers)
	Disconnect [► 1520]	Disconnects this instance (stop triggering notifications)
	Dispose. [► 1520]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.
	Dispose(Boolean) [► 1521]	Releases unmanaged and - optionally - managed resources.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize [► 1521]	Finalizes an instance of the NotificationTriggerSource class. (Overrides Object.Finalize .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)

	Name	Description
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnNotification [▶ 1522]	Called when [notification].
	OnNotificationAsyn c [▶ 1522]	Called when [notification asynchronous].
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

[TwinCAT.Ads.Server.IAddNotificationTrigger](#) [▶ 1515]

[System.IDisposable](#)

6.4.14.1 NotificationTriggerSource Methods

The [NotificationTriggerSource](#) [▶ 1517] type exposes the following members.

Methods

	Name	Description
	Add [▶ 1519]	Adds the specified observer.
	Connect [▶ 1519]	Connects this instance (Start observing notification triggers)
	Disconnect [▶ 1520]	Disconnects this instance (stop triggering notifications)
	Dispose . [▶ 1520]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.
	Dispose(Boolean) [▶ 1521]	Releases unmanaged and - optionally - managed resources.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize [▶ 1521]	Finalizes an instance of the NotificationTriggerSource [▶ 1517] class. (Overrides Object.Finalize .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnNotification [▶ 1522]	Called when [notification].
	OnNotificationAsyn c [▶ 1522]	Called when [notification asynchronous].
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[NotificationTriggerSource Class](#) [▶ 1517]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.14.1.1 NotificationTriggerSource.Add Method

Adds the specified observer.

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IAddNotificationTrigger Add(
    INotificationTrigger observer
)
```

Parameters

observer	Type: TwinCAT.Ads.Server.INotificationTrigger [▶ 1516] The observer.
----------	---

Return Value

Type: [IAddNotificationTrigger](#) [[▶ 1515](#)]

IAddNotificationTrigger.

Implements

[IAddNotificationTrigger.Add\(INotificationTrigger\)](#) [[▶ 1516](#)]

Exceptions

Exception	Condition
ObjectDisposedException	NotificationTriggerSource

Reference

[NotificationTriggerSource Class](#) [[▶ 1517](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.14.1.2 NotificationTriggerSource.Connect Method

Connects this instance (Start observing notification triggers)

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Connect()
```

Exceptions

Exception	Condition
ObjectDisposedException	NotificationTriggerSource

Reference

[NotificationTriggerSource Class](#) [[▶ 1517](#)]

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.14.1.3 NotificationTriggerSource.Disconnect Method

Disconnects this instance (stop triggering notifications)

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Disconnect()
```

Exceptions

Exception	Condition
ObjectDisposedException	NotificationTriggerSource

Reference

[NotificationTriggerSource Class \[► 1517\]](#)

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.14.1.4 NotificationTriggerSource.Dispose Method

Overload List

	Name	Description
	Dispose. [► 1520]	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.
	Dispose(Boolean) [► 1521]	Releases unmanaged and - optionally - managed resources.

Reference

[NotificationTriggerSource Class \[► 1517\]](#)

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.14.1.4.1 NotificationTriggerSource.Dispose Method

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Dispose()
```

Implements

[IDisposable.Dispose.](#)

Reference

[NotificationTriggerSource Class \[► 1517\]](#)

[Dispose Overload \[► 1520\]](#)

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.14.1.4.2 NotificationTriggerSource.Dispose Method (Boolean)

Releases unmanaged and - optionally - managed resources.

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual void Dispose(  
    bool disposing  
)
```

Parameters

disposing	Type: System.Boolean true to release both managed and unmanaged resources; false to release only unmanaged resources.
-----------	--

Reference

[NotificationTriggerSource Class \[► 1517\]](#)

[Dispose Overload \[► 1520\]](#)

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.14.1.5 NotificationTriggerSource.Finalize Method

Finalizes an instance of the [NotificationTriggerSource \[► 1517\]](#) class.

Namespace: [TwinCAT.Ads.Server \[► 1374\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected override void Finalize()
```

Implements

[Object.Finalize.](#)

Reference

[NotificationTriggerSource Class \[► 1517\]](#)

[TwinCAT.Ads.Server Namespace \[► 1374\]](#)

6.4.14.1.6 NotificationTriggerSource.OnNotification Method

Called when [notification].

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual void OnNotification(
    long idx
)
```

Parameters

idx	Type: System.Int64 The index.
-----	--

Reference

[NotificationTriggerSource Class](#) [[▶ 1517](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.14.1.7 NotificationTriggerSource.OnNotificationAsync Method

Called when [notification asynchronous].

Namespace: [TwinCAT.Ads.Server](#) [[▶ 1374](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task OnNotificationAsync(
    long idx,
    CancellationToken cancel
)
```

Parameters

idx	Type: System.Int64 The index.
cancel	Type: System.Threading.CancellationToken The cancel.

Return Value

Type: [Task](#)
Task.

Reference

[NotificationTriggerSource Class](#) [[▶ 1517](#)]

[TwinCAT.Ads.Server Namespace](#) [[▶ 1374](#)]

6.4.15 LoopbackNotConnectedException Class

The Tcp Loopback client is not connected. Implements the [AdsServerException](#) [▶ 1443]

Inheritance Hierarchy

System.Object
 System.Exception
 TwinCAT.Ads.Server.AdsServerException [▶ 1443]
 TwinCAT.Ads.Server.LoopbackNotConnectedException

Namespace: TwinCAT.Ads.Server [▶ 1374]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public class LoopbackNotConnectedException : AdsServerException
```

The LoopbackNotConnectedException type exposes the following members.

Properties

Name	Description
Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
ErrorCode [▶ 1446]	Gets or sets the error code. (Inherited from AdsServerException [▶ 1443].)
HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
Message	Gets a message that describes the current exception. (Inherited from Exception .)
Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Methods

Name	Description
Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
GetHashCode	Serves as the default hash function. (Inherited from Object .)
GetObjectData [▶ 1447]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from AdsServerException [▶ 1443].)

	Name	Description
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

Reference

[TwinCAT.Ads.Server Namespace](#) [► 1374]

[TwinCAT.Ads.Server.AdsServerException](#) [► 1443]

6.4.15.1 LoopbackNotConnectedException Properties

The [LoopbackNotConnectedException](#) [► 1523] type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	ErrorCode [► 1446]	Gets or sets the error code. (Inherited from AdsServerException [► 1443].)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Reference

[LoopbackNotConnectedException Class](#) [► 1523]

[TwinCAT.Ads.Server Namespace](#) [► 1374]

6.4.15.2 LoopbackNotConnectedException Methods

The [LoopbackNotConnectedException](#) [► 1523] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData [▶ 1447]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from AdsServerException [▶ 1443].)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Reference

[LoopbackNotConnectedException Class](#) [▶ 1523]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]

6.4.15.3 LoopbackNotConnectedException Events

The [LoopbackNotConnectedException](#) [▶ 1523] type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

Reference

[LoopbackNotConnectedException Class](#) [▶ 1523]

[TwinCAT.Ads.Server Namespace](#) [▶ 1374]




6.5 TwinCAT.Ads.SumCommand Namespace

ADS offers powerful and fast communication to exchange any kind of information. It's possible to read single variables or complete arrays and structures with each one single ADS-API call. ADS Sum-Commands offer to read/write with one single ADS call multiple variables which are not structured within a linear memory, effectively reducing roundtrips.

Classes

	Class	Description
	ISumCommandExtension [▶ 1591]	Extension class for ISumCommand [▶ 1527] derived classes.
	ResultSumCommand [▶ 1533]	Result class for an asynchronous TwinCAT.Ads.SumCommand:
	ResultSumHandles [▶ 1538]	Result class for an asynchronous SumCreateHandles>.
	ResultSumHandles2 [▶ 1540]	Result class for an asynchronous SumCreateHandles.
	ResultSumReadRaw [▶ 1544]	Result class for an asynchronous SumRead that contains blocks of raw memory data (unmarshaled data).
	ResultSumValues [▶ 1547]	Result class for an asynchronous ISumRead [▶ 1596]. The Values are returned as objects.
	ResultSumValues2.S [▶ 1605]	Extended result class for an asynchronous ISumRead2.S [▶ 1601].
 	SumAnyTypeRead [▶ 1608]	Symbolic ADS Sum read access (ANY_TYPE marshalling) on already loaded symbols.
 	SumCreateHandles [▶ 1550]	SumCommandBase for getting variable handles by a set of InstancePaths
 	SumHandleRead [▶ 1557]	Read (primitive, Any) values by Handle SumCommandBase.
 	SumHandleWrite [▶ 1561]	Write any (primitive) values by Handle SumCommandBase.
 	SumInstancePathAnyTypeRead [▶ 1615]	SumRead Command that uses the instancePath as symbol ID / Address and returns objects as Values (ANY Type marshaling)
 	SumReleaseHandles [▶ 1565]	Release Handles SumCommandBase.
 	SumSymbolRead [▶ 1572]	Symbolic ADS Sum read access (automatic marshalling)
 	SumSymbolWrite [▶ 1582]	Class for ADS Sum symbolic Write Access.

Interfaces

	Interface	Description
	ISumCommand [▶ 1527]	Interface for SumCommands (Combined commands)
	ISumRead [▶ 1596]	Interface ISumRead Implements the ISumCommand [▶ 1527]
	ISumRead2.S. [▶ 1601]	Interface ISumRead2 (supports extended Sum SubResults) Implements the ISumCommand [▶ 1527]

Enumerations

	Enumeration	Description
	SumAccessMode [▶ 1608]	Indicates how to handle each single sub-sum-command (IGIO, ByHandle, ByName)
	SumCommandErrorStrategy [▶ 1614]	The Error strategy for ISumCommand [▶ 1527]
	SumCommandMode [▶ 1615]	Indicates, which (extended) SumCommand to use.

Also see about this

-  [ISumSymbolRead Interface](#) [▶ 1587]

6.5.1 ISumCommand Interface

Interface for SumCommands (Combined commands)

Namespace: [TwinCAT.Ads.SumCommand](#) [▶ 1525]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229








Syntax



C#

```
public interface ISumCommand
```

The ISumCommand type exposes the following members.

Properties

	Name	Description
	Connection [▶ 1531]	The connection used for communication.
	ErrorStrategy [▶ 1531]	Gets the error strategy of the ISumCommand
	Executed [▶ 1529]	Gets a value indicating whether this ISumCommand was already executed.
	Failed [▶ 1529]	Gets a value indicating whether this ISumCommand failed.
	FailedCount [▶ 1532]	Gets the count of failed subcommands.
	Result [▶ 1530]	Gets the AdsErrorCode [▶ 664] of the main SumCommandBase ADS Request
	SubResults [▶ 1530]	Gets the sub results of the single Sub Requests.

	Name	Description
	Succeeded [► 1531]	Gets a value indicating whether this ISumCommand is succeeded.
	SucceededCount [► 1532]	Gets the count of succeeded subcommands.

Extension Methods

	Name	Description
	AllFailed [► 1592]	Gets a value indicating, that all SubCommands failed. (Defined by ISumCommandExtension [► 1591].)
	AllSucceeded [► 1593]	Gets a value indicating, that all SubCommands succeeded. (Defined by ISumCommandExtension [► 1591].)
	FirstSubError [► 1594]	Gets the first SubError that is not NoError [► 664] (Defined by ISumCommandExtension [► 1591].)
	OneFailed [► 1594]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Defined by ISumCommandExtension [► 1591].)
	OneSucceeded [► 1595]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Defined by ISumCommandExtension [► 1591].)
	OverallError [► 1595]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Defined by ISumCommandExtension [► 1591].)

Remarks

ADS offers powerful and fast communication to exchange any kind of information. It's possible to read single variables or complete arrays and structures with each one single ADS-API call. The ADS Sum Command command offers to read with one single ADS call multiple variables which are not structured within a linear memory. As a result the ADS caller application (like scada Systems etc.) can extremely speed up cyclic polling : Sample :

- Until now : Polling 4000 single variables which are not in a linear area (like array / structure / fixed PLC address) would cause 4000 single Ads-ReadReq with each 1-2 ms protocol time. As a result the scanning of these variables take 4000ms-8000ms.
- New Ads-Command allows to read multiple variables with one single ADS-ReadReq : 4000 single variables are handled with e.g. 8 single Ads-ReadReq (each call requesting 500 variables) with each 1-2 ms protocol time. As a result the scanning of these variables take just few 10ms.

REQUIREMENTS AND IMPORTANT LIMITATIONS: Note that ADS is just a transport layer, but there could be important side effects. So read this requirements and take care on limitations:

- **Version of target ADS Device** - - ADS itself is just the transport layer, but the requested ADS device has to support the ADS-Command.
- **Bytes length of requested data** - - Requesting a large list of values from variables is fine, but the requested data of the Ads-response (the data-byte-length) have to pass the AMS Router (size by default a 2048kb) So the caller has to limit the requested variables based on calculation of requested data-byte-length.
- **Number of Sub-ADS calls : Highly recommended to max. 500 !** - - If the PLC is processing one ADS request, it will completely work on this single ADS request BEFORE starting next PLC cycle. As a result one single ADS request with 200.000 sub-Ads-requests would cause that PLC would collect and copy 200.000 variables into one single ADS response, before starting next PLC. So this large number of ads-sub-commands will jitter the PLC execution ! **We highly recommend to not request more than 500 Ads-Sub commands**










Reference

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

6.5.1.1 ISumCommand Properties

The [ISumCommand \[▸ 1527\]](#) type exposes the following members.

Properties

	Name	Description
	Connection [▸ 1531]	The connection used for communication.
	ErrorStrategy [▸ 1531]	Gets the error strategy of the ISumCommand [▸ 1527]
	Executed [▸ 1529]	Gets a value indicating whether this ISumCommand [▸ 1527] was already executed.
	Failed [▸ 1529]	Gets a value indicating whether this ISumCommand [▸ 1527] failed.
	FailedCount [▸ 1532]	Gets the count of failed subcommands.
	Result [▸ 1530]	Gets the AdsErrorCode [▸ 664] of the main SumCommandBase ADS Request
	SubResults [▸ 1530]	Gets the sub results of the single Sub Requests.
	Succeeded [▸ 1531]	Gets a value indicating whether this ISumCommand [▸ 1527] is succeeded.
	SucceededCount [▸ 1532]	Gets the count of succeeded subcommands.

Reference

[ISumCommand Interface \[▸ 1527\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▸ 1525\]](#)

6.5.1.1.1 ISumCommand.Executed Property

Gets a value indicating whether this [ISumCommand \[▸ 1527\]](#) was already executed.

Namespace: [TwinCAT.Ads.SumCommand \[▸ 1525\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool Executed { get; }
```

Property Value

Type: [Boolean](#)

true if executed; otherwise, false.

Reference

[ISumCommand Interface \[▸ 1527\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▸ 1525\]](#)

6.5.1.1.2 ISumCommand.Failed Property

Gets a value indicating whether this [ISumCommand \[▸ 1527\]](#) failed.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool Failed { get; }
```

Property Value

Type: [Boolean](#)

true if failed; otherwise, false.

Reference

[ISumCommand Interface](#) [[▶ 1527](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.1.1.3 ISumCommand.Result Property

Gets the [AdsErrorCode](#) [[▶ 664](#)] of the main SumCommandBase ADS Request

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode Result { get; }
```

Property Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The result.

Reference

[ISumCommand Interface](#) [[▶ 1527](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.1.1.4 ISumCommand.SubResults Property

Gets the sub results of the single Sub Requests.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode[] SubResults { get; }
```

Property Value

Type: [.AdsErrorCode](#) [[▶ 664](#)].

The sub results.

Reference

[ISumCommand Interface \[► 1527\]](#)

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

6.5.1.1.5 ISumCommand.Succeeded Property

Gets a value indicating whether this [ISumCommand \[► 1527\]](#) is succeeded.

Namespace: [TwinCAT.Ads.SumCommand \[► 1525\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool Succeeded { get; }
```

Property Value

Type: [Boolean](#)

true if succeeded; otherwise, false.

Reference

[ISumCommand Interface \[► 1527\]](#)

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

6.5.1.1.6 ISumCommand.Connection Property

The connection used for communication.

Namespace: [TwinCAT.Ads.SumCommand \[► 1525\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IAdsConnection Connection { get; }
```

Property Value

Type: [IAdsConnection \[► 876\]](#)

Remarks

This can be the AdsConnection or AdsClient object.

Reference

[ISumCommand Interface \[► 1527\]](#)

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

6.5.1.1.7 ISumCommand.ErrorStrategy Property

Gets the error strategy of the [ISumCommand \[► 1527\]](#)

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
SumCommandErrorStrategy ErrorStrategy { get; }
```

Property Value

Type: [SumCommandErrorStrategy](#) [[▶ 1614](#)]

The error strategy.

Reference

[ISumCommand Interface](#) [[▶ 1527](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.1.1.8 ISumCommand.FailedCount Property

Gets the count of failed subcommands.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int FailedCount { get; }
```

Property Value

Type: [Int32](#)

The failed count.

Reference

[ISumCommand Interface](#) [[▶ 1527](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.1.1.9 ISumCommand.SucceededCount Property

Gets the count of succeeded subcommands.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int SucceededCount { get; }
```

Property Value

Type: [Int32](#)

The succeeded count.

Reference

[ISumCommand Interface \[▸ 1527\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▸ 1525\]](#)

6.5.1.2 ISumCommand Methods

The [ISumCommand \[▸ 1527\]](#) type exposes the following members.

Extension Methods

	Name	Description
	AllFailed [▸ 1592]	Gets a value indicating, that all SubCommands failed. (Defined by ISumCommandExtension [▸ 1591] .)
	AllSucceeded [▸ 1593]	Gets a value indicating, that all SubCommands succeeded. (Defined by ISumCommandExtension [▸ 1591] .)
	FirstSubError [▸ 1594]	Gets the first SubError that is not NoError [▸ 664] (Defined by ISumCommandExtension [▸ 1591] .)
	OneFailed [▸ 1594]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Defined by ISumCommandExtension [▸ 1591] .)
	OneSucceeded [▸ 1595]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Defined by ISumCommandExtension [▸ 1591] .)
	OverallError [▸ 1595]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Defined by ISumCommandExtension [▸ 1591] .)

Reference

[ISumCommand Interface \[▸ 1527\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▸ 1525\]](#)

6.5.2 ResultSumCommand Class

Result class for an asynchronous [TwinCAT.Ads.SumCommand \[▸ 1525\]](#):

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.ResultAds \[▸ 1116\]](#)

[TwinCAT.Ads.SumCommand.ResultSumCommand](#)

[TwinCAT.Ads.SumCommand.ResultSumHandles \[▸ 1538\]](#)

[TwinCAT.Ads.SumCommand.ResultSumReadRaw \[▸ 1544\]](#)

[TwinCAT.Ads.SumCommand.ResultSumValues \[▸ 1547\]](#)

[TwinCAT.Ads.SumCommand.ResultSumValues2.S. \[▸ 1605\]](#)

Namespace: [TwinCAT.Ads.SumCommand \[▸ 1525\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229










Syntax

C#








```
public class ResultSumCommand : ResultAds
```

The [ResultSumCommand](#) type exposes the following members.

Properties

	Name	Description
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	FirstSubError [▶ 1535]	Gets the first SubError that is not NoError [▶ 664]
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	OverallError [▶ 1536]	Gets the Overall (combined error) from SumCommand AND SubCommands.
	OverallFailed [▶ 1536]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results).
	OverallSucceeded [▶ 1537]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands)
	SubErrors [▶ 1535]	Gets the Error codes [▶ 664] for the single SumCommands.
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Remarks

The result contains the overall ErrorCode of the SumCommand [OverallError](#) [[▶ 1536](#)] and all the Suberrors of the SubRequests [SubErrors](#) [[▶ 1535](#)].










Reference

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.2.1 ResultSumCommand Properties

The [ResultSumCommand](#) [[▶ 1533](#)] type exposes the following members.

Properties

	Name	Description
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	FirstSubError [▶ 1535]	Gets the first SubError that is not NoError [▶ 664]
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	OverallError [▶ 1536]	Gets the Overall (combined error) from SumCommand AND SubCommands.
	OverallFailed [▶ 1536]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results).
	OverallSucceeded [▶ 1537]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands)
	SubErrors [▶ 1535]	Gets the Error codes [▶ 664] for the single SumCommands.
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Reference

[ResultSumCommand Class](#) [[▶ 1533](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.2.1.1 ResultSumCommand.SubErrors Property

Gets the [Error codes](#) [[▶ 664](#)] for the single SumCommands.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode[]? SubErrors { get; protected set; }
```

Property Value

Type: [.AdsErrorCode](#) [[▶ 664](#)].

Reference

[ResultSumCommand Class](#) [[▶ 1533](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.2.1.2 ResultSumCommand.FirstSubError Property

Gets the first SubError that is not [NoError](#) [[▶ 664](#)]

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode FirstSubError { get; }
```

Property Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The first subError.

Reference

[ResultSumCommand Class](#) [[▶ 1533](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.2.1.3 ResultSumCommand.OverallError Property

Gets the Overall (combined error) from SumCommand AND SubCommands.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode OverallError { get; }
```

Property Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

The combined error or [NoError](#) [[▶ 664](#)]

Remarks

This will return the ErrorCode of the Sumcommand (if failed) or the first failed subcommand.

Reference

[ResultSumCommand Class](#) [[▶ 1533](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.2.1.4 ResultSumCommand.OverallFailed Property

Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results).

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool OverallFailed { get; }
```

Property Value

Type: [Boolean](#)

true if [overall failed]; otherwise, false.

Reference

[ResultSumCommand Class \[► 1533\]](#)

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

6.5.2.1.5 ResultSumCommand.OverallSucceeded Property

Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands)

Namespace: [TwinCAT.Ads.SumCommand \[► 1525\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool OverallSucceeded { get; }
```

Property Value

Type: [Boolean](#)

true if [sub results succeeded]; otherwise, false.

Reference








[ResultSumCommand Class \[► 1533\]](#)

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

6.5.2.2 ResultSumCommand Methods

The [ResultSumCommand \[► 1533\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [► 1128]	Sets the error state of this ResultAds [► 1116] (Inherited from ResultAds [► 1116] .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultSumCommand Class \[► 1533\]](#)

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

6.5.3 ResultSumHandles Class

Result class for an asynchronous SumCreateHandles>.

Inheritance Hierarchy

System.Object

TwinCAT.Ads.ResultAds [[▶ 1116](#)]

TwinCAT.Ads.SumCommand.ResultSumCommand [[▶ 1533](#)]

TwinCAT.Ads.SumCommand.ResultSumHandles

TwinCAT.Ads.SumCommand.ResultSumHandles2 [[▶ 1540](#)]

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229











Syntax

C#



```
public class ResultSumHandles : ResultSumCommand
```






The ResultSumHandles type exposes the following members.

Properties

	Name	Description
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	FirstSubError [▶ 1535]	Gets the first SubError that is not NoError [▶ 664] (Inherited from ResultSumCommand [▶ 1533].)
	Handles [▶ 1539]	The registered Symbol handles.
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	OverallError [▶ 1536]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Inherited from ResultSumCommand [▶ 1533].)
	OverallFailed [▶ 1536]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Inherited from ResultSumCommand [▶ 1533].)
	OverallSucceeded [▶ 1537]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Inherited from ResultSumCommand [▶ 1533].)
	SubErrors [▶ 1535]	Gets the Error codes [▶ 664] for the single SumCommands. (Inherited from ResultSumCommand [▶ 1533].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)

	Name	Description
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)











Reference

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.3.1 ResultSumHandles Properties

The [ResultSumHandles](#) [[▶ 1538](#)] type exposes the following members.

Properties

	Name	Description
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	FirstSubError [▶ 1535]	Gets the first SubError that is not NoError [▶ 664] (Inherited from ResultSumCommand [▶ 1533].)
	Handles [▶ 1539]	The registered Symbol handles.
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	OverallError [▶ 1536]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Inherited from ResultSumCommand [▶ 1533].)
	OverallFailed [▶ 1536]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Inherited from ResultSumCommand [▶ 1533].)
	OverallSucceeded [▶ 1537]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Inherited from ResultSumCommand [▶ 1533].)
	SubErrors [▶ 1535]	Gets the Error codes [▶ 664] for the single SumCommands. (Inherited from ResultSumCommand [▶ 1533].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Reference

[ResultSumHandles Class](#) [[▶ 1538](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.3.1.1 ResultSumHandles.Handles Property

The registered Symbol handles.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▸ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint[]? Handles { get; }
```

Property Value

Type: [.UInt32](#).

Reference








[ResultSumHandles Class](#) [[▸ 1538](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▸ 1525](#)]

6.5.3.2 ResultSumHandles Methods

The [ResultSumHandles](#) [[▸ 1538](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▸ 1128]	Sets the error state of this ResultAds [▸ 1116] (Inherited from ResultAds [▸ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultSumHandles Class](#) [[▸ 1538](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▸ 1525](#)]

6.5.4 ResultSumHandles2 Class

Result class for an asynchronous SumCreateHandles.

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.ResultAds](#) [[▸ 1116](#)]

[TwinCAT.Ads.SumCommand.ResultSumCommand](#) [[▸ 1533](#)]

[TwinCAT.Ads.SumCommand.ResultSumHandles](#) [[▸ 1538](#)]

TwinCAT.Ads.SumCommand.ResultSumHandles2

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#












```
public class ResultSumHandles2 : ResultSumHandles
```

The ResultSumHandles2 type exposes the following members.




Constructors





	Name	Description
	ResultSumHandles2 [▶ 1542]	Initializes a new instance of the ResultSumHandles2 class.

Properties

	Name	Description
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	FirstSubError [▶ 1535]	Gets the first SubError that is not NoError [▶ 664] (Inherited from ResultSumCommand [▶ 1533].)
	Handles [▶ 1539]	The registered Symbol handles. (Inherited from ResultSumHandles [▶ 1538].)
	InstancePaths [▶ 1543]	The symbol/instance paths belonging to the handles and the subErrors.
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	OverallError [▶ 1536]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Inherited from ResultSumCommand [▶ 1533].)
	OverallFailed [▶ 1536]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Inherited from ResultSumCommand [▶ 1533].)
	OverallSucceeded [▶ 1537]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Inherited from ResultSumCommand [▶ 1533].)
	SubErrors [▶ 1535]	Gets the Error codes [▶ 664] for the single SumCommands. (Inherited from ResultSumCommand [▶ 1533].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)

	Name	Description
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.4.1 ResultSumHandles2 Constructor

Initializes a new instance of the [ResultSumHandles2](#) [[▶ 1540](#)] class.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public ResultSumHandles2(
    AdsErrorCode complete,
    AdsErrorCode[]? subErrors,
    uint[]? handles,
    string[]? instancePaths
)
```

Parameters

complete	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The overall ADS error codes.
subErrors	Type: .TwinCAT.Ads.AdsErrorCode [▶ 664]. The single ADS SubErrors.
handles	Type: .System.UInt32 . The registered handles.
instancePaths	Type: .System.String . The instance paths.

Reference


[ResultSumHandles2 Class](#) [[▶ 1540](#)]











[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.4.2 ResultSumHandles2 Properties

The [ResultSumHandles2](#) [[▶ 1540](#)] type exposes the following members.

Properties

	Name	Description
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)

	Name	Description
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	FirstSubError [▶ 1535]	Gets the first SubError that is not NoError [▶ 664] (Inherited from ResultSumCommand [▶ 1533].)
	Handles [▶ 1539]	The registered Symbol handles. (Inherited from ResultSumHandles [▶ 1538].)
	InstancePaths [▶ 1543]	The symbol/instance paths belonging to the handles and the subErrors.
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	OverallError [▶ 1536]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Inherited from ResultSumCommand [▶ 1533].)
	OverallFailed [▶ 1536]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Inherited from ResultSumCommand [▶ 1533].)
	OverallSucceeded [▶ 1537]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Inherited from ResultSumCommand [▶ 1533].)
	SubErrors [▶ 1535]	Gets the Error codes [▶ 664] for the single SumCommands. (Inherited from ResultSumCommand [▶ 1533].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Reference

[ResultSumHandles2 Class](#) [[▶ 1540](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.4.2.1 ResultSumHandles2.InstancePaths Property

The symbol/instance paths belonging to the handles and the subErrors.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string[]? InstancePaths { get; }
```

Property Value

Type: [.String](#).

Reference








[ResultSumHandles2 Class](#) [[▶ 1540](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.4.3 ResultSumHandles2 Methods

The [ResultSumHandles2](#) [[▶ 1540](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultSumHandles2 Class](#) [[▶ 1540](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.5 ResultSumReadRaw Class

Result class for an asynchronous SumRead that contains blocks of raw memory data (unmarshaled data).

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.ResultAds](#) [[▶ 1116](#)]

[TwinCAT.Ads.SumCommand.ResultSumCommand](#) [[▶ 1533](#)]

[TwinCAT.Ads.SumCommand.ResultSumReadRaw](#)

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229





Syntax







C#

```
public class ResultSumReadRaw : ResultSumCommand
```








The ResultSumReadRaw type exposes the following members.

Properties

	Name	Description
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	FirstSubError [▶ 1535]	Gets the first SubError that is not NoError [▶ 664] (Inherited from ResultSumCommand [▶ 1533].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)

	Name	Description
	OverallError [▶ 1536]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Inherited from ResultSumCommand [▶ 1533].)
	OverallFailed [▶ 1536]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Inherited from ResultSumCommand [▶ 1533].)
	OverallSucceeded [▶ 1537]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Inherited from ResultSumCommand [▶ 1533].)
	ReadBlocks [▶ 1546]	List of the read raw memory blocks read.
	SubErrors [▶ 1535]	Gets the Error codes [▶ 664] for the single SumCommands. (Inherited from ResultSumCommand [▶ 1533].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)






Reference






[TwinCAT.Ads.SumCommand Namespace](#) [▶ 1525]

6.5.5.1 ResultSumReadRaw Properties

The [ResultSumReadRaw](#) [▶ 1544] type exposes the following members.

Properties

	Name	Description
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	FirstSubError [▶ 1535]	Gets the first SubError that is not NoError [▶ 664] (Inherited from ResultSumCommand [▶ 1533].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	OverallError [▶ 1536]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Inherited from ResultSumCommand [▶ 1533].)

	Name	Description
	OverallFailed [▶ 1536]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Inherited from ResultSumCommand [▶ 1533].)
	OverallSucceeded [▶ 1537]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Inherited from ResultSumCommand [▶ 1533].)
	ReadBlocks [▶ 1546]	List of the read raw memory blocks read.
	SubErrors [▶ 1535]	Gets the Error codes [▶ 664] for the single SumCommands. (Inherited from ResultSumCommand [▶ 1533].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)

Reference

[ResultSumReadRaw Class](#) [▶ [1544](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [▶ [1525](#)]

6.5.5.1.1 ResultSumReadRaw.ReadBlocks Property

List of the read raw memory blocks read.

Namespace: [TwinCAT.Ads.SumCommand](#) [▶ [1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IList<ReadOnlyMemory<byte>>? ReadBlocks { get; }
```

Property Value

Type: [IList.ReadOnlyMemory.Byte](#).

Reference





[ResultSumReadRaw Class](#) [▶ [1544](#)]




[TwinCAT.Ads.SumCommand Namespace](#) [▶ [1525](#)]

6.5.5.2 ResultSumReadRaw Methods

The [ResultSumReadRaw](#) [▶ [1544](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)

	Name	Description
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultSumReadRaw Class](#) [[▶ 1544](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.6 ResultSumValues Class

Result class for an asynchronous [ISumRead](#) [[▶ 1596](#)]. The Values are returned as objects.

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.ResultAds](#) [[▶ 1116](#)]

[TwinCAT.Ads.SumCommand.ResultSumCommand](#) [[▶ 1533](#)]

[TwinCAT.Ads.SumCommand.ResultSumValues](#)

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229









Syntax



C#

```
public class ResultSumValues : ResultSumCommand
```








The ResultSumValues type exposes the following members.

Properties

	Name	Description
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	FirstSubError [▶ 1535]	Gets the first SubError that is not NoError [▶ 664] (Inherited from ResultSumCommand [▶ 1533].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	OverallError [▶ 1536]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Inherited from ResultSumCommand [▶ 1533].)
	OverallFailed [▶ 1536]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Inherited from ResultSumCommand [▶ 1533].)
	OverallSucceeded [▶ 1537]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Inherited from ResultSumCommand [▶ 1533].)
	SubErrors [▶ 1535]	Gets the Error codes [▶ 664] for the single SumCommands. (Inherited from ResultSumCommand [▶ 1533].)

	Name	Description
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)
	Values [▶ 1549]	The Read data as marshalled values

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference









[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]



[TwinCAT.Ads.SumCommand.ResultSumValues2.S.](#) [[▶ 1605](#)]

6.5.6.1 ResultSumValues Properties

The [ResultSumValues](#) [[▶ 1547](#)] type exposes the following members.

Properties

	Name	Description
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	FirstSubError [▶ 1535]	Gets the first SubError that is not NoError [▶ 664] (Inherited from ResultSumCommand [▶ 1533].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	OverallError [▶ 1536]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Inherited from ResultSumCommand [▶ 1533].)
	OverallFailed [▶ 1536]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Inherited from ResultSumCommand [▶ 1533].)
	OverallSucceeded [▶ 1537]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Inherited from ResultSumCommand [▶ 1533].)
	SubErrors [▶ 1535]	Gets the Error codes [▶ 664] for the single SumCommands. (Inherited from ResultSumCommand [▶ 1533].)

	Name	Description
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116] .)
	Values [▶ 1549]	The Read data as marshalled values

Reference

[ResultSumValues Class \[▶ 1547\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▶ 1525\]](#)

6.5.6.1 ResultSumValues.Values Property

The Read data as marshalled values

Namespace: [TwinCAT.Ads.SumCommand \[▶ 1525\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object[]? Values { get; }
```

Property Value

Type: [.Object](#).

Reference








[ResultSumValues Class \[▶ 1547\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▶ 1525\]](#)

6.5.6.2 ResultSumValues Methods

The [ResultSumValues \[▶ 1547\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116] .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultSumValues Class \[▶ 1547\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▶ 1525\]](#)

6.5.7 SumCreateHandles Class

SumCommandBase for getting variable handles by a set of InstancePaths

Inheritance Hierarchy

[System.Object](#)

SumCommandWrapper.SumReadWrite.

TwinCAT.Ads.SumCommand.SumCreateHandles

Namespace: [TwinCAT.Ads.SumCommand \[▶ 1525\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229





Syntax

C#



```
public class SumCreateHandles : SumCommandWrapper<SumReadWrite>
```







The SumCreateHandles type exposes the following members.

Constructors

	Name	Description
	SumCreateHandles(IAdsConnection, IList.String.) [▶ 1552]	Initializes a new instance of the SumCreateHandles class.
	SumCreateHandles(IAdsConnection, .String.) [▶ 1553]	Initializes a new instance of the SumCreateHandles class.
	SumCreateHandles(IAdsConnection, IList.String., SumCommandErrorStrategy) [▶ 1554]	Initializes a new instance of the SumCreateHandles class.
	SumCreateHandles(IAdsConnection, .String., SumCommandErrorStrategy) [▶ 1554]	Initializes a new instance of the SumCreateHandles class.

Methods

	Name	Description
	CreateHandles [▶ 1555]	Creates the ADS handles.
	CreateHandlesAsync [▶ 1556]	Create handles asynchronously.

	Name	Description
	<u>Equals</u>	Determines whether the specified object is equal to the current object. (Inherited from <u>Object</u> .)
	<u>Finalize</u>	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from <u>Object</u> .)
	<u>GetHashCode</u>	Serves as the default hash function. (Inherited from <u>Object</u> .)
	<u>GetType</u>	Gets the <u>Type</u> of the current instance. (Inherited from <u>Object</u> .)
	<u>MemberwiseClone</u>	Creates a shallow copy of the current <u>Object</u> . (Inherited from <u>Object</u> .)
	<u>ToString</u>	Returns a string that represents the current object. (Inherited from <u>Object</u> .)

Examples

Usage of Sum commands with handles (CreateHandles, Read, Write, ReleaseHandles)

```

/// <summary>
/// Defines the entry point of the application.
/// </summary>
/// <param name="args">The arguments.</param>
static async void Main(string[] args)
{
    Console.WriteLine("");
    Console.WriteLine("Press [Enter] for start:");
    Console.ReadLine();

    // Parse the command-line arguments
    AmsAddress address = ArgParser.Parse(args);

    CancellationTokenSource cancelSource = new CancellationTokenSource();
    CancellationToken cancel = cancelSource.Token;

    using (AdsClient client = new AdsClient())
    {
        // Connect the AdsClient to the device target.
        client.Connect(address);

        string[] instancePathList = {
            "GVL.bVar",
            "GVL.iCount",
            "TwinCAT_SystemInfoVarList._AppInfo.ProjectName"
        };
        SumCreateHandles createHandlesCommand = new SumCreateHandles(client, instancePathList);

        var resultCreateHandles = await createHandlesCommand.CreateHandlesAsync(cancel);

        if (resultCreateHandles.Succeeded)
        {
            uint[] handles = resultCreateHandles.Handles;
            Type[] valueTypes = new Type[]
            {
                typeof(bool),
                typeof(short),
                typeof(string)
            };
            SumHandleRead readCommand = new SumHandleRead(client, handles, valueTypes);
            var resultRead = await readCommand.ReadAsync(cancel);

            if (resultRead.Succeeded)
            {
                object[] readValues = resultRead.Values;

                for (int i = 0; i < instancePathList.Length; i++)
                {
                    Console.WriteLine("Symbol: {0} (Value: {1}, Type: {2})", instancePathList[i], readValues[i].ToString(), valueTypes[i].Name);
                }

                // Sum Command Write
                SumHandleWrite writeCommand = new SumHandleWrite(client, handles, valueTypes);
                object[] writeValues = new object[] { true, (short)42, "MyNewProjectName" };

```

```

        await writeCommand.WriteAsync(writeValues, cancel);
    }

    SumReleaseHandles releaseCommand = new SumReleaseHandles(client, handles);
    await releaseCommand.ReleaseHandlesAsync(cancel);
}
}

Console.WriteLine("");
Console.WriteLine("Press [Enter] for leave:");
Console.ReadLine();
}

```

Reference

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

[TwinCAT.Ads.SumCommand.ISumCommand \[► 1527\]](#)





[TwinCAT.Ads.SumCommand.SumReleaseHandles \[► 1565\]](#)

[TwinCAT.Ads.SumCommand.SumHandleRead \[► 1557\]](#)

[TwinCAT.Ads.SumCommand.SumHandleWrite \[► 1561\]](#)

6.5.7.1 SumCreateHandles Constructor

Overload List

	Name	Description
	SumCreateHandles(IAdsConnection, IList.String.) [► 1552]	Initializes a new instance of the SumCreateHandles [► 1550] class.
	SumCreateHandles(IAdsConnection, .String.) [► 1553]	Initializes a new instance of the SumCreateHandles [► 1550] class.
	SumCreateHandles(IAdsConnection, IList.String, SumCommandErrorStrategy) [► 1554]	Initializes a new instance of the SumCreateHandles [► 1550] class.
	SumCreateHandles(IAdsConnection, .String, SumCommandErrorStrategy) [► 1554]	Initializes a new instance of the SumCreateHandles [► 1550] class.

Reference

[SumCreateHandles Class \[► 1550\]](#)

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

6.5.7.1.1 SumCreateHandles Constructor (IAdsConnection, IList.String.)

Initializes a new instance of the [SumCreateHandles \[► 1550\]](#) class.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SumCreateHandles(
    IAdsConnection connection,
    IList<string> instancePaths
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
instancePaths	Type: System.Collections.Generic.IList.String . The instance paths.

Remarks

Uses the [Strict](#) [[▶ 1614](#)].

Reference

[SumCreateHandles Class](#) [[▶ 1550](#)]

[SumCreateHandles Overload](#) [[▶ 1552](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.7.1.2 SumCreateHandles Constructor (IAdsConnection, .String.)

Initializes a new instance of the [SumCreateHandles](#) [[▶ 1550](#)] class.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SumCreateHandles(
    IAdsConnection connection,
    string[] instancePaths
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
instancePaths	Type: .System.String . The instance paths.

Remarks

Uses the [Strict](#) [[▶ 1614](#)].

Reference

[SumCreateHandles Class](#) [[▶ 1550](#)]

[SumCreateHandles Overload \[► 1552\]](#)

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

6.5.7.1.3 SumCreateHandles Constructor (IAdsConnection, IList.String., SumCommandErrorStrategy)

Initializes a new instance of the [SumCreateHandles \[► 1550\]](#) class.

Namespace: [TwinCAT.Ads.SumCommand \[► 1525\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SumCreateHandles(
    IAdsConnection connection,
    IList<string> instancePaths,
    SumCommandErrorStrategy errorStrategy
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
instancePaths	Type: System.Collections.Generic.IList.String. The instance paths.
errorStrategy	Type: TwinCAT.Ads.SumCommand.SumCommandErrorStrategy [► 1614] The error strategy.

Reference

[SumCreateHandles Class \[► 1550\]](#)

[SumCreateHandles Overload \[► 1552\]](#)

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

6.5.7.1.4 SumCreateHandles Constructor (IAdsConnection, .String., SumCommandErrorStrategy)

Initializes a new instance of the [SumCreateHandles \[► 1550\]](#) class.

Namespace: [TwinCAT.Ads.SumCommand \[► 1525\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SumCreateHandles(
    IAdsConnection connection,
    string[] instancePaths,
    SumCommandErrorStrategy errorStrategy
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
------------	---

instancePaths	Type: <u>.System.String</u> . The instance paths.
errorStrategy	Type: <u>TwinCAT.Ads.SumCommand.SumCommandErrorStrategy</u> [▶ 1614] The error strategy.

Reference

[SumCreateHandles Class](#) [[▶ 1550](#)]









[SumCreateHandles Overload](#) [[▶ 1552](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.7.2 SumCreateHandles Methods

The [SumCreateHandles](#) [[▶ 1550](#)] type exposes the following members.

Methods

	Name	Description
	CreateHandles [▶ 1555]	Creates the ADS handles.
	CreateHandlesAsync [▶ 1556]	Create handles asynchronously.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[SumCreateHandles Class](#) [[▶ 1550](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.7.2.1 SumCreateHandles.CreateHandles Method

Creates the ADS handles.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint[] CreateHandles()
```

Return Value

Type: [.UInt32](#).
[System.UInt32\[\]](#).

Exceptions

Exception	Condition
AdsSumCommandException [▶ 735]	SumGetHandles failed!
AdsSumCommandException [▶ 735]	SumGetHandlesCommand failed!

Reference

[SumCreateHandles Class](#) [[▶ 1550](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

[SumCreateHandles.TryCreateHandles\(.String..., .UInt32..., .AdsErrorCode..\)](#)

[SumCreateHandles.CreateHandlesAsync\(CancellationTokentoken\)](#) [[▶ 1556](#)]

6.5.7.2.2 SumCreateHandles.CreateHandlesAsync Method

Create handles asynchronously.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultSumHandles> CreateHandlesAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationTokentoken The cancellation token.
--------	--

Return Value

Type: [Task.ResultSumHandles](#) [[▶ 1538](#)].

An asynchronous task that represents the 'ReadWriteRaw' operation and returns a [ResultSumHandles](#) [[▶ 1538](#)]. The overall error return code is contained in the [ErrorCode](#) [[▶ 1120](#)] property.

Reference

[SumCreateHandles Class](#) [[▶ 1550](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

[SumCreateHandles.CreateHandles.](#) [[▶ 1555](#)]

[SumCreateHandles.TryCreateHandles\(.String..., .UInt32..., .AdsErrorCode..\)](#)

6.5.8 SumHandleRead Class

Read (primitive, Any) values by Handle SumCommandBase.

Inheritance Hierarchy

```
System.Object
  SumCommandBase
    SumRead
      TwinCAT.Ads.SumCommand.SumHandleRead
```

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229










Syntax

C#

```
public class SumHandleRead : SumRead
```

The SumHandleRead type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Read [▶ 1559]	Reads the values.
	ReadAsync [▶ 1560]	Read the values asynchronously.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryRead [▶ 1560]	Tries to read the values of the .

Remarks

This is an ADS Sum Command to access values by handle information. It is always used in combination with `.` By design (and in contrast to the symbolic access in `SumRead`, [SumSymbolWrite](#) [[▶ 1582](#)]) this access method can act only with ADS ANY Type (Primitive) values (disadvantage). The Advantage is, that no symbolic information must be loaded before accessing the values, see samples:

Examples

Usage of Sum commands with handles (CreateHandles, Read, Write, ReleaseHandles)

```
/// <summary>
/// Defines the entry point of the application.
/// </summary>
/// <param name="args">The arguments.</param>
static async void Main(string[] args)
{
    Console.WriteLine("");
    Console.WriteLine("Press [Enter] for start:");
    Console.ReadLine();
}
```

```

// Parse the command-line arguments
AmsAddress address = ArgParser.Parse(args);

CancellationTokenSource cancelSource = new CancellationTokSource();
CancellationToken cancel = cancelSource.Token;

using (AdsClient client = new AdsClient())
{
// Connect the AdsClient to the device target.
client.Connect(address);

string[] instancePathList = {
    "GVL.bVar",
    "GVL.iCount",
    "TwinCAT_SystemInfoVarList._AppInfo.ProjectName"
};
SumCreateHandles createHandlesCommand = new SumCreateHandles(client,instancePathList);

var resultCreateHandles = await createHandlesCommand.CreateHandlesAsync(cancel);

if (resultCreateHandles.Succeeded)
{
    uint[] handles = resultCreateHandles.Handles;
    Type[] valueTypes = new Type[]
    {
        typeof(bool),
        typeof(short),
        typeof(string)
    };
    SumHandleRead readCommand = new SumHandleRead(client, handles, valueTypes);
    var resultRead = await readCommand.ReadAsync(cancel);

    if (resultRead.Succeeded)
    {
        object[] readValues = resultRead.Values;

        for (int i = 0; i < instancePathList.Length; i++)
        {
            Console.WriteLine("Symbol: {0} (Value: {1}, Type: {2})", instancePathList[i], readValues
[i].ToString(), valueTypes[i].Name);
        }

        // Sum Command Write
        SumHandleWrite writeCommand = new SumHandleWrite(client, handles, valueTypes);
        object[] writeValues = new object[] { true, (short)42, "MyNewProjectName" };

        await writeCommand.WriteAsync(writeValues, cancel);
    }

    SumReleaseHandles releaseCommand = new SumReleaseHandles(client, handles);
    await releaseCommand.ReleaseHandlesAsync(cancel);
}
}

Console.WriteLine("");
Console.WriteLine("Press [Enter] for leave:");
Console.ReadLine();
}

```

Reference

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

[TwinCAT.Ads.SumCommand.SumCreateHandles \[► 1550\]](#)

[TwinCAT.Ads.SumCommand.SumReleaseHandles \[► 1565\]](#)

[TwinCAT.Ads.SumCommand.ISumCommand \[► 1527\]](#)

[TwinCAT.Ads.SumCommand.SumReleaseHandles \[► 1565\]](#)

[TwinCAT.Ads.SumCommand.SumReleaseHandles \[► 1565\]](#)

[TwinCAT.Ads.SumCommand.SumHandleWrite \[► 1561\]](#)










Also see about this

 [SumSymbolRead Class \[▶ 1572\]](#)

6.5.8.1 SumHandleRead Methods

The [SumHandleRead \[▶ 1557\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Read [▶ 1559]	Reads the values.
	ReadAsync [▶ 1560]	Read the values asynchronously.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryRead [▶ 1560]	Tries to read the values of the .

Reference

[SumHandleRead Class \[▶ 1557\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▶ 1525\]](#)

6.5.8.1.1 SumHandleRead.Read Method

Reads the values.

Namespace: [TwinCAT.Ads.SumCommand \[▶ 1525\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object[] Read()
```

Return Value

Type: [.Object](#).
[System.Object\[\]](#).

Exceptions

Exception	Condition
AdsSumCommandException [▶ 735]	SumAnyReadByHandleCommand failed!

Reference[SumHandleRead Class \[► 1557\]](#)[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)[SumHandleRead.TryRead\(.Object..., .AdsErrorCode..\) \[► 1560\]](#)[SumHandleRead.ReadAsync\(CancellationTokens\) \[► 1560\]](#)**6.5.8.1.2 SumHandleRead.ReadAsync Method**

Read the values asynchronously.

Namespace: [TwinCAT.Ads.SumCommand \[► 1525\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public Task<ResultSumValues> ReadAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token
--------	--

Return ValueType: [Task.ResultSumValues \[► 1547\]](#).An asynchronous task that represents the 'ReadSymbols' operation and returns a [ResultSumValues \[► 1547\]](#). The overall error return code is contained in the [ErrorCode \[► 1120\]](#) property.**Reference**[SumHandleRead Class \[► 1557\]](#)[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)[SumHandleRead.Read. \[► 1559\]](#)[SumHandleRead.TryRead\(.Object..., .AdsErrorCode..\) \[► 1560\]](#)**6.5.8.1.3 SumHandleRead.TryRead Method**

Tries to read the values of the .

Namespace: [TwinCAT.Ads.SumCommand \[► 1525\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public AdsErrorCode TryRead(
    out Object[]? values,
    out AdsErrorCode[]?? returnCodes
)
```


Parameters

values	Type: .System.Object. The values.
returnCodes	Type: .TwinCAT.Ads.AdsErrorCode [▶ 664].. The return codes.

Return Value

Type: [AdsErrorCode](#) [▶ 664]
AdsErrorCode.

Reference

- [SumHandleRead Class](#) [▶ 1557]
- [TwinCAT.Ads.SumCommand Namespace](#) [▶ 1525]
- [TwinCAT.Ads.SumCommand.SumHandleRead](#) [▶ 1557]
- [SumHandleRead.Read.](#) [▶ 1559]
- [SumHandleRead.ReadAsync\(CancellationTokens\)](#) [▶ 1560]

6.5.9 SumHandleWrite Class

Write any (primitive) values by Handle SumCommandBase.

Inheritance Hierarchy

- [System.Object](#)
- SumCommandBase
- SumWrite
- TwinCAT.Ads.SumCommand.SumHandleWrite

Namespace: [TwinCAT.Ads.SumCommand](#) [▶ 1525]
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229







Syntax




C#

```
public class SumHandleWrite : SumWrite
```

The SumHandleWrite type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

	Name	Description
	TryWrite [► 1564]	Tries to write the values.
	Write [► 1564]	Writes the values to the Symbols.
	WriteAsync [► 1565]	Write the values asynchronously.

Remarks

This is an ADS Sum Command to access values by handle information. It is always used in combination with `and`. By design (and in contrast to the symbolic access in `SumRead`, [SumSymbolWrite \[► 1582\]](#)) this access method can act only with ADS ANY Type (Primitive) values (disadvantage). The Advantage is, that no symbolic information must be loaded before accessing the values, see samples:

Examples

Usage of Sum commands with handles (CreateHandles, Read, Write, ReleaseHandles)

```

/// <summary>
/// Defines the entry point of the application.
/// </summary>
/// <param name="args">The arguments.</param>
static async void Main(string[] args)
{
    Console.WriteLine("");
    Console.WriteLine("Press [Enter] for start:");
    Console.ReadLine();

    // Parse the command-line arguments
    AmsAddress address = ArgParser.Parse(args);

    CancellationTokensource cancelSource = new CancellationTokensource();
    CancellationToken cancel = cancelSource.Token;

    using (AdsClient client = new AdsClient())
    {
        // Connect the AdsClient to the device target.
        client.Connect(address);

        string[] instancePathList = {
            "GVL.bVar",
            "GVL.iCount",
            "TwinCAT_SystemInfoVarList._AppInfo.ProjectName"
        };
        SumCreateHandles createHandlesCommand = new SumCreateHandles(client, instancePathList);

        var resultCreateHandles = await createHandlesCommand.CreateHandlesAsync(cancel);

        if (resultCreateHandles.Succeeded)
        {
            uint[] handles = resultCreateHandles.Handles;
            Type[] valueTypes = new Type[]
            {
                typeof(bool),
                typeof(short),
                typeof(string)
            };
            SumHandleRead readCommand = new SumHandleRead(client, handles, valueTypes);
            var resultRead = await readCommand.ReadAsync(cancel);

            if (resultRead.Succeeded)
            {
                object[] readValues = resultRead.Values;

                for (int i = 0; i < instancePathList.Length; i++)
                {
                    Console.WriteLine("Symbol: {0} (Value: {1}, Type: {2})", instancePathList[i], readValues
[i].ToString(), valueTypes[i].Name);
                }

                // Sum Command Write
                SumHandleWrite writeCommand = new SumHandleWrite(client, handles, valueTypes);
                object[] writeValues = new object[] { true, (short)42, "MyNewProjectName" };

```

```

        await writeCommand.WriteAsync(writeValues, cancel);
    }

    SumReleaseHandles releaseCommand = new SumReleaseHandles(client, handles);
    await releaseCommand.ReleaseHandlesAsync(cancel);
}
}

Console.WriteLine("");
Console.WriteLine("Press [Enter] for leave.");
Console.ReadLine();
}

```

Reference

- [TwinCAT.Ads.SumCommand Namespace \[▶ 1525\]](#)
- [TwinCAT.Ads.SumCommand.SumCreateHandles \[▶ 1550\]](#)
- [TwinCAT.Ads.SumCommand.SumReleaseHandles \[▶ 1565\]](#)
- [TwinCAT.Ads.SumCommand.ISumCommand \[▶ 1527\]](#)
- [TwinCAT.Ads.SumCommand.SumReleaseHandles \[▶ 1565\]](#)
- [TwinCAT.Ads.SumCommand.SumReleaseHandles \[▶ 1565\]](#)
- [TwinCAT.Ads.SumCommand.SumHandleRead \[▶ 1557\]](#)










Also see about this

-  [SumSymbolRead Class \[▶ 1572\]](#)

6.5.9.1 SumHandleWrite Methods

The [SumHandleWrite \[▶ 1561\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryWrite [▶ 1564]	Tries to write the values.
	Write [▶ 1564]	Writes the values to the Symbols.
	WriteAsync [▶ 1565]	Write the values asynchronously.

Reference

- [SumHandleWrite Class \[▶ 1561\]](#)
- [TwinCAT.Ads.SumCommand Namespace \[▶ 1525\]](#)

6.5.9.1.1 SumHandleWrite.TryWrite Method

Tries to write the values.

Namespace: [TwinCAT.Ads.SumCommand](#) [► 1525]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWrite(
    Object[] values,
    out AdsErrorCode[]?? returnCodes
)
```

Parameters

values	Type: .System.Object . The values (ANY/Primitive types only).
returnCodes	Type: .TwinCAT.Ads.AdsErrorCode [► 664].. The return codes.

Return Value

Type: [AdsErrorCode](#) [► 664]

AdsErrorCode.

Reference

[SumHandleWrite Class](#) [► 1561]

[TwinCAT.Ads.SumCommand Namespace](#) [► 1525]

[SumHandleWrite.Write\(Object.\)](#) [► 1564]

[SumHandleWrite.WriteAsync\(Object., CancellationToken\)](#) [► 1565]

6.5.9.1.2 SumHandleWrite.Write Method

Writes the values to the Symbols.

Namespace: [TwinCAT.Ads.SumCommand](#) [► 1525]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Write(
    Object[] values
)
```

Parameters

values	Type: .System.Object . The Values (Any primitive types only):
--------	--

Exceptions

Exception	Condition
AdsSumCommandException [▶ 735]	SumAnyWriteByHandleCommand failed!

Reference

[SumHandleWrite Class](#) [[▶ 1561](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

[SumHandleWrite.TryWrite\(Object, AdsErrorCode\)](#) [[▶ 1564](#)]

[SumHandleWrite.WriteAsync\(Object, CancellationToken\)](#) [[▶ 1565](#)]

6.5.9.1.3 SumHandleWrite.WriteAsync Method

Write the values asynchronously.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultSumCommand> WriteAsync(
    Object[] values,
    CancellationToken cancel
)
```

Parameters

values	Type: System.Object . The values.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultSumCommand](#) [[▶ 1533](#)].

An asynchronous task that represents the 'ReadWriteRaw' operation and returns a [ResultSumCommand](#) [[▶ 1533](#)]. The overall error return code is contained in the [ErrorCode](#) [[▶ 1120](#)] property.

Reference

[SumHandleWrite Class](#) [[▶ 1561](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

[SumHandleWrite.Write\(Object\)](#) [[▶ 1564](#)]

[SumHandleWrite.TryWrite\(Object, AdsErrorCode\)](#) [[▶ 1564](#)]

6.5.10 SumReleaseHandles Class

Release Handles SumCommandBase.

Inheritance Hierarchy

[System.Object](#)

SumCommandWrapper.SumWrite.

TwinCAT.Ads.SumCommand.SumReleaseHandles

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229



Syntax

C#










```
public class SumReleaseHandles : SumCommandWrapper<SumWrite>
```

The SumReleaseHandles type exposes the following members.

Constructors

	Name	Description
	SumReleaseHandles (IAdsConnection , .UInt32 .) [▶ 1568]	Initializes a new instance of the SumReleaseHandles class.
	SumReleaseHandles (IAdsConnection , .UInt32 , SumCommandErrorStrategy) [▶ 1569]	Initializes a new instance of the SumReleaseHandles class.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ReleaseHandles [▶ 1570]	Releases the handles.
	ReleaseHandlesAsync [▶ 1570]	Releases the handles asynchronously.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryReleaseHandles [▶ 1571]	Tries to Release the Handles

Remarks

Releases the specified ADS handles. Usually used in conjunction with the [SumCreateHandles](#) [[▶ 1550](#)] and the [SumHandleRead](#) [[▶ 1557](#)] / [SumHandleWrite](#) [[▶ 1561](#)] commands.

Examples

Usage of Sum commands with handles (CreateHandles, Read, Write, ReleaseHandles)

```

/// <summary>
/// Defines the entry point of the application.
/// </summary>
/// <param name="args">The arguments.</param>
static async void Main(string[] args)
{
    Console.WriteLine("");
    Console.WriteLine("Press [Enter] for start:");
    Console.ReadLine();

    // Parse the command-line arguments
    AmsAddress address = ArgParser.Parse(args);

    CancellationTokenSource cancelSource = new CancellationTokenSource();
    CancellationToken cancel = cancelSource.Token;

    using (AdsClient client = new AdsClient())
    {
        // Connect the AdsClient to the device target.
        client.Connect(address);

        string[] instancePathList = {
            "GVL.bVar",
            "GVL.iCount",
            "TwinCAT_SystemInfoVarList._AppInfo.ProjectName"
        };
        SumCreateHandles createHandlesCommand = new SumCreateHandles(client, instancePathList);

        var resultCreateHandles = await createHandlesCommand.CreateHandlesAsync(cancel);

        if (resultCreateHandles.Succeeded)
        {
            uint[] handles = resultCreateHandles.Handles;
            Type[] valueTypes = new Type[]
            {
                typeof(bool),
                typeof(short),
                typeof(string)
            };
            SumHandleRead readCommand = new SumHandleRead(client, handles, valueTypes);
            var resultRead = await readCommand.ReadAsync(cancel);

            if (resultRead.Succeeded)
            {
                object[] readValues = resultRead.Values;

                for (int i = 0; i < instancePathList.Length; i++)
                {
                    Console.WriteLine("Symbol: {0} (Value: {1}, Type: {2})", instancePathList[i], readValues
[i].ToString(), valueTypes[i].Name);
                }

                // Sum Command Write
                SumHandleWrite writeCommand = new SumHandleWrite(client, handles, valueTypes);
                object[] writeValues = new object[] { true, (short)42, "MyNewProjectName" };

                await writeCommand.WriteAsync(writeValues, cancel);
            }

            SumReleaseHandles releaseCommand = new SumReleaseHandles(client, handles);
            await releaseCommand.ReleaseHandlesAsync(cancel);
        }
    }

    Console.WriteLine("");
    Console.WriteLine("Press [Enter] for leave:");
    Console.ReadLine();
}

```

Reference

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

[TwinCAT.Ads.SumCommand.ISumCommand \[► 1527\]](#)

[TwinCAT.Ads.SumCommand.SumCreateHandles](#) [[▶ 1550](#)]

[TwinCAT.Ads.SumCommand.SumHandleRead](#) [[▶ 1557](#)]

[TwinCAT.Ads.SumCommand.SumHandleWrite](#) [[▶ 1561](#)]

Also see about this

[SumReleaseHandles Constructor](#) [[▶ 1568](#)]

6.5.10.1 SumReleaseHandles Constructor

Overload List

	Name	Description
	SumReleaseHandles (IAdsConnection, .UInt32.) [▶ 1568]	Initializes a new instance of the SumReleaseHandles [▶ 1565] class.
	SumReleaseHandles (IAdsConnection, .UInt32., SumCommandErrorStrategy) [▶ 1569]	Initializes a new instance of the SumReleaseHandles [▶ 1565] class.

Reference

[SumReleaseHandles Class](#) [[▶ 1565](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

Also see about this

[SumReleaseHandles Class](#) [[▶ 1565](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.10.1.1 SumReleaseHandles Constructor (IAdsConnection, .UInt32.)

Initializes a new instance of the [SumReleaseHandles](#) [[▶ 1565](#)] class.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SumReleaseHandles(
    IAdsConnection connection,
    uint[] serverHandles
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
serverHandles	Type: .System.UInt32. The handles.

Remarks

Uses the [Strict](#) [[1614](#)].

Reference

[SumReleaseHandles Class](#) [[1565](#)]

[SumReleaseHandles Overload](#) [[1568](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[1525](#)]

6.5.10.1.2 SumReleaseHandles Constructor (IAdsConnection, .UInt32., SumCommandErrorStrategy)

Initializes a new instance of the [SumReleaseHandles](#) [[1565](#)] class.

Namespace: [TwinCAT.Ads.SumCommand](#) [[1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SumReleaseHandles(
    IAdsConnection connection,
    uint[] serverHandles,
    SumCommandErrorStrategy errorStrategy
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [876] The connection.
serverHandles	Type: .System.UInt32 . The handles.
errorStrategy	Type: TwinCAT.Ads.SumCommand.SumCommandErrorStrategy [1614] The error strategy.

Reference

[SumReleaseHandles Class](#) [[1565](#)]




[SumReleaseHandles Overload](#) [[1568](#)]







[TwinCAT.Ads.SumCommand Namespace](#) [[1525](#)]

6.5.10.2 SumReleaseHandles Methods

The [SumReleaseHandles](#) [[1565](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)

	Name	Description
	GetType	Gets the <u>Type</u> of the current instance. (Inherited from <u>Object</u> .)
	MemberwiseClone	Creates a shallow copy of the current <u>Object</u> . (Inherited from <u>Object</u> .)
	ReleaseHandles [▶ 1570]	Releases the handles.
	ReleaseHandlesAsync [▶ 1570]	Releases the handles asynchronously.
	ToString	Returns a string that represents the current object. (Inherited from <u>Object</u> .)
	TryReleaseHandles [▶ 1571]	Tries to Release the Handles

Reference

[SumReleaseHandles Class](#) [▶ [1565](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [▶ [1525](#)]

6.5.10.2.1 SumReleaseHandles.ReleaseHandles Method

Releases the handles.

Namespace: [TwinCAT.Ads.SumCommand](#) [▶ [1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void ReleaseHandles()
```

Exceptions

Exception	Condition
AdsSumCommandException [▶ 735]	SumReleaseHandlesCommand failed!

Reference

[SumReleaseHandles Class](#) [▶ [1565](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [▶ [1525](#)]

[SumReleaseHandles.TryReleaseHandles\(AdsErrorCode..\)](#) [▶ [1571](#)]

[SumReleaseHandles.ReleaseHandlesAsync\(CancellationTokens\)](#) [▶ [1570](#)]

6.5.10.2.2 SumReleaseHandles.ReleaseHandlesAsync Method

Releases the handles asynchronously.

Namespace: [TwinCAT.Ads.SumCommand](#) [▶ [1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultSumCommand> ReleaseHandlesAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultSumCommand](#) [[▶ 1533](#)].

An asynchronous task that represents the 'ReleaseHandles' operation and returns a [ResultSumCommand](#) [[▶ 1533](#)]. The overall error return code is contained in the [ErrorCode](#) [[▶ 1120](#)] property.

Reference

[SumReleaseHandles Class](#) [[▶ 1565](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.10.2.3 SumReleaseHandles.TryReleaseHandles Method

Tries to Release the Handles

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryReleaseHandles(
    out AdsErrorCode[]?? returnCodes
)
```

Parameters

returnCodes	Type: .TwinCAT.Ads.AdsErrorCode [▶ 664].. The return codes.
-------------	--

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]

AdsErrorCode.

Reference

[SumReleaseHandles Class](#) [[▶ 1565](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

[SumReleaseHandles.ReleaseHandles.](#) [[▶ 1570](#)]

[SumReleaseHandles.ReleaseHandlesAsync\(CancellationToken\)](#) [[▶ 1570](#)]

6.5.11 SumSymbolRead Class

Symbolic ADS Sum read access (automatic marshalling)

Inheritance Hierarchy

System.Object

SumCommandWrapper.SumReadWrite.

SumSymbolCommand.SumReadWrite.

SumSymbolReadBase

TwinCAT.Ads.SumCommand.SumSymbolRead

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229














Syntax

C#

```
public class SumSymbolRead : SumSymbolReadBase,
    ISumRead, ISumCommand, ISumRead2<ISymbol>
```

The SumSymbolRead type exposes the following members.

Methods

	Name	Description
	createResults [▶ 1579]	Creates the SumCommand results (abstract) (Overrides SumSymbolReadBase.createResults(IList.SumDataEntity., IList.ReadOnlyMemory.Byte..., .AdsErrorCode..))
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Read [▶ 1577]	Reads the combined (Sum) Symbols and returns them as value array.
	Read2 [▶ 1580]	Reads the Values.
	Read2Async [▶ 1581]	Read2 as an asynchronous operation.
	ReadAsResult [▶ 1581]	Reads the combined (Sum) Symbols and returns them as Result [▶ 1547] object.
	ReadAsync [▶ 1578]	Reads Symbol values as an asynchronous operation.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryRead [▶ 1579]	Reads the specified symbols.

Extension Methods

	Name	Description
	AllFailed [▶ 1592]	Gets a value indicating, that all SubCommands failed. (Defined by ISumCommandExtension [▶ 1591].)
	AllSucceeded [▶ 1593]	Gets a value indicating, that all SubCommands succeeded. (Defined by ISumCommandExtension [▶ 1591].)

Name	Description
FirstSubError [▶ 1594]	Gets the first SubError that is not NoError [▶ 664] (Defined by ISumCommandExtension [▶ 1591].)
OneFailed [▶ 1594]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Defined by ISumCommandExtension [▶ 1591].)
OneSucceeded [▶ 1595]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Defined by ISumCommandExtension [▶ 1591].)
OverallError [▶ 1595]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Defined by ISumCommandExtension [▶ 1591].)
PollValues(IObservable.Unit.) [▶ 1359]	Overloaded. Polls a series of symbols via a ISumRead [▶ 1596] command. The SumCommand will read all contained values with every trigger. (Defined by ValueSymbolExtensions [▶ 1333].)
PollValues(TimeSpan) [▶ 1361]	Overloaded. Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger in one roundtrip. (Defined by ValueSymbolExtensions [▶ 1333].)
PollValues2.ISymbol.(IObservable.Unit.) [▶ 1370]	Overloaded. Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger in one roundtrip. (Defined by ValueSymbolExtensions [▶ 1333].)
PollValues2.ISymbol.(TimeSpan) [▶ 1372]	Overloaded. Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger. (Defined by ValueSymbolExtensions [▶ 1333].)

Remarks

The SumSymbolRead implements symbolic read access with automatic (dynamic) value marshalling. The advantage of the symbolic access is (in contrast to the handle access classes [SumHandleRead](#) [[▶ 1557](#)], [SumHandleWrite](#) [[▶ 1561](#)]) that all type information is available when using this ADS Sum Command. The disadvantage is, that the Symbolic information must be loaded beforehand, see examples. The [SumSymbolWrite](#) [[▶ 1582](#)] defaults to an InstancePath/SymbolName read ([ValueByName](#) [[▶ 1608](#)]).

Examples

Usage of SumSymbolRead/SumSymbolWrite with AdsSession

```

/// <summary>
/// Defines the entry point of the application.
/// </summary>
/// <param name="args">The arguments.</param>
static async void Main(string[] args)
{
    Console.WriteLine("");
    Console.WriteLine("Press [Enter] for start:");
    Console.ReadLine();

    // Parse the command-line arguments
    AmsAddress address = ArgParser.Parse(args);

    CancellationTokenSource cancelSource = new CancellationTokenSource();
    CancellationToken cancel = cancelSource.Token;

    using (AdsSession session = new AdsSession(address))
    {
        // Connect to the device target.
        AdsConnection connection = (AdsConnection)session.Connect();

        // Load symbolic information
        var resultSymbols = await session.SymbolServer.GetSymbolsAsync(cancel);
        resultSymbols.ThrowOnError();
    }
}

```

```

ISymbolCollection<ISymbol> symbols = resultSymbols.Symbols;

ISymbol bVar1 = symbols["GVL.bVar1"];
ISymbol bVar2 = symbols["GVL.bVar2"];
ISymbol projectName = symbols["TwinCAT_SystemInfoVarList._AppInfo.ProjectName"];

SymbolCollection coll = new SymbolCollection() {bVar1, bVar2, projectName};

// Sum Command Read
SumSymbolRead readCommand = new SumSymbolRead(connection,coll);
var resultReadValues = await readCommand.ReadAsync(cancel);

if (resultReadValues.Succeeded)
{
    object[] values = resultReadValues.Values;

    for (int i = 0; i < coll.Count; i++)
    {
        Console.WriteLine("Symbol: {0} (Value: {1}, Type: {2})", coll[i].InstancePath, values[i].ToString(), values[i].GetType().Name);
    }
}
// Sum Command Write

SumSymbolWrite writeCommand = new SumSymbolWrite(connection,coll);
object[] writeValues = new object[] {true, (short) 42, "MyNewProjectName"};

var resultWrite = await writeCommand.WriteAsync(writeValues,cancel);
}

Console.WriteLine("");
Console.WriteLine("Press [Enter] for leave:");
Console.ReadLine();
}

```

Examples

Usage of SumSymbolRead/SumSymbolWrite with AdsClient

```

/// <summary>
/// Defines the entry point of the application.
/// </summary>
/// <param name="args">The arguments.</param>
static async void Main(string[] args)
{
    Console.WriteLine("");
    Console.WriteLine("Press [Enter] for start:");
    Console.ReadLine();

    CancellationTokenSource cancelSource = new CancellationTokenSource();
    CancellationToken cancel = cancelSource.Token;

    // Parse the command-line arguments
    AmsAddress address = ArgParser.Parse(args);

    using (AdsClient client = new AdsClient())
    {
        // Connect the AdsClient to the device target.
        client.Connect(address);

        // Load symbolic information
        ISymbolLoader loader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
        var resultReadSymbols = await loader.GetSymbolsAsync(cancel);

        resultReadSymbols.ThrowOnError();

        ISymbolCollection<ISymbol> allSymbols = resultReadSymbols.Symbols;

        ISymbol bVar1 = allSymbols["GVL.bVar1"];
        ISymbol bVar2 = allSymbols["GVL.iCount"];
        ISymbol projectName = allSymbols["TwinCAT_SystemInfoVarList._AppInfo.ProjectName"];

        SymbolCollection symbols = new SymbolCollection() {bVar1, bVar2, projectName};

        // Sum Command Read
        SumSymbolRead readCommand = new SumSymbolRead(client,symbols);
        var resultSumRead = await readCommand.ReadAsync(cancel);
    }
}

```

```
if (resultSumRead.Succeeded)
{
    object[] values = resultSumRead.Values;

    for (int i = 0; i < symbols.Count; i++)
    {
        Console.WriteLine("Symbol: {0} (Value: {1}, Type: {2})", symbols[i].InstancePath, values[i].
ToString(), values[i].GetType().Name);
    }

    // Sum Command Write
    SumSymbolWrite writeCommand = new SumSymbolWrite(client, symbols);
    object[] writeValues = new object[] { true, (short)42, "MyNewProjectName" };

    var resultSumWrite = await writeCommand.WriteAsync(writeValues, cancel);
}
}

Console.WriteLine("");
Console.WriteLine("Press [Enter] for leave:");
Console.ReadLine();
}
```

Reference

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

[TwinCAT.Ads.SumCommand.ISumCommand \[► 1527\]](#)

[TwinCAT.Ads.SumCommand.SumAnyTypeRead \[► 1608\]](#)

[TwinCAT.Ads.SumCommand.SumSymbolWrite \[► 1582\]](#)

Also see about this

- [ValueSymbolExtensions.PollValues Method \(ISumSymbolRead, IObservable.Unit.\) \[► 1352\]](#)
- [ISumSymbolRead Interface \[► 1587\]](#)
- [ValueSymbolExtensions.PollValues Method \(ISumSymbolRead, TimeSpan\) \[► 1353\]](#)

6.5.11.1 SumSymbolRead Constructor

Initializes a new instance of the [SumSymbolRead \[► 1572\]](#) class.

Namespace: [TwinCAT.Ads.SumCommand \[► 1525\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public SumSymbolRead(
    IAdsConnection connection,
    IList<ISymbol> symbols
)
```

Parameters

- | | |
|------------|--|
| connection | Type: TwinCAT.Ads.IAdsConnection [► 876]
The ADS Connection. |
| symbols | Type: System.Collections.Generic.IList.ISymbol [► 2691] .
The symbols to read |

Reference














[SumSymbolRead Class \[► 1572\]](#)

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

6.5.11.2 SumSymbolRead Methods

The [SumSymbolRead \[► 1572\]](#) type exposes the following members.

Methods

	Name	Description
	createResults [► 1579]	Creates the SumCommand results (abstract) (Overrides SumSymbolReadBase.createResults(IList.SumDataEntity., IList.ReadOnlyMemory.Byte..., .AdsErrorCode.) .)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Read [► 1577]	Reads the combined (Sum) Symbols and returns them as value array.
	Read2 [► 1580]	Reads the Values.
	Read2Async [► 1581]	Read2 as an asynchronous operation.
	ReadAsResult [► 1581]	Reads the combined (Sum) Symbols and returns them as Result [► 1547] object.
	ReadAsync [► 1578]	Reads Symbol values as an asynchronous operation.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryRead [► 1579]	Reads the specified symbols.

Extension Methods

	Name	Description
	AllFailed [► 1592]	Gets a value indicating, that all SubCommands failed. (Defined by ISumCommandExtension [► 1591] .)
	AllSucceeded [► 1593]	Gets a value indicating, that all SubCommands succeeded. (Defined by ISumCommandExtension [► 1591] .)
	FirstSubError [► 1594]	Gets the first SubError that is not NoError [► 664] (Defined by ISumCommandExtension [► 1591] .)
	OneFailed [► 1594]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Defined by ISumCommandExtension [► 1591] .)
	OneSucceeded [► 1595]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Defined by ISumCommandExtension [► 1591] .)
	OverallError [► 1595]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Defined by ISumCommandExtension [► 1591] .)
	PollValues(IObservable.Unit.) [► 1359]	Overloaded. Polls a series of symbols via a ISumRead [► 1596] command. The SumCommand will read all contained values with every trigger. (Defined by ValueSymbolExtensions [► 1333] .)

	Name	Description
	PollValues(TimeSpan) [▶ 1361]	Overloaded. Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger in one roundtrip. (Defined by ValueSymbolExtensions [▶ 1333] .)
	PollValues2.ISymbol.(IObservable.Unit.) [▶ 1370]	Overloaded. Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger in one roundtrip. (Defined by ValueSymbolExtensions [▶ 1333] .)
	PollValues2.ISymbol.(TimeSpan) [▶ 1372]	Overloaded. Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger. (Defined by ValueSymbolExtensions [▶ 1333] .)

Reference

[SumSymbolRead Class \[▶ 1572\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▶ 1525\]](#)

Also see about this

- ▣ [ValueSymbolExtensions.PollValues Method \(ISumSymbolRead, IObservable.Unit.\) \[▶ 1352\]](#)
- ▣ [ISumSymbolRead Interface \[▶ 1587\]](#)
- ▣ [ValueSymbolExtensions.PollValues Method \(ISumSymbolRead, TimeSpan\) \[▶ 1353\]](#)

6.5.11.2.1 SumSymbolRead.Read Method

Reads the combined (Sum) Symbols and returns them as value array.

Namespace: [TwinCAT.Ads.SumCommand \[▶ 1525\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object[] Read()
```

Return Value

Type: [.Object](#).
The value array.

Implements

[ISumRead.Read. \[▶ 1599\]](#)

Exceptions

Exception	Condition
AdsSumCommandException [▶ 735]	SumSymbolRead failed!

Remarks

The return values are automatically marshalled to their appropriate .NET types. This method throws Exceptions, when the Read access fails.

Reference

[SumSymbolRead Class](#) [► 1572]

[TwinCAT.Ads.SumCommand Namespace](#) [► 1525]

[ISumRead.TryRead\(.Object..., .AdsErrorCode..\)](#) [► 1601]

[ISumRead.ReadAsync\(CancellationTokens\)](#) [► 1600]

Also see about this

- [SumSymbolRead.TryRead Method](#) [► 1579]
- [SumSymbolRead.ReadAsync Method](#) [► 1578]
- [ISumSymbolRead.Read Method](#) [► 1589]

6.5.11.2 SumSymbolRead.ReadAsync Method

Reads Symbol values as an asynchronous operation.

Namespace: [TwinCAT.Ads.SumCommand](#) [► 1525]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultSumValues> ReadAsync(
    CancellationTokens cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationTokens The cancellation token.
--------	--

Return Value

Type: [Task.ResultSumValues](#) [► 1547].

An asynchronous task that represents the 'Read' operation and returns a [ResultSumValues](#) [► 1547]. The overall error return code is contained in the [ErrorCode](#) [► 1120] property.

Implements

[ISumRead.ReadAsync\(CancellationTokens\)](#) [► 1600]

Reference

[SumSymbolRead Class](#) [► 1572]

[TwinCAT.Ads.SumCommand Namespace](#) [► 1525]

[SumSymbolRead.Read.](#) [► 1577]

[SumSymbolRead.TryRead\(.Object..., .AdsErrorCode..\)](#) [► 1579]

Also see about this

[ISumSymbolRead.ReadAsync Method \[▸ 1590\]](#)

6.5.11.2.3 SumSymbolRead.TryRead Method

Reads the specified symbols.

Namespace: [TwinCAT.Ads.SumCommand \[▸ 1525\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryRead(
    out Object[]? values,
    out AdsErrorCode[]?? returnCodes
)
```

Parameters

values	Type: .System.Object. The values.
returnCodes	Type: .TwinCAT.Ads.AdsErrorCode [▸ 664]. The return codes.

Return Value

Type: [AdsErrorCode \[▸ 664\]](#)

AdsErrorCode.

Implements

[ISumRead.TryRead\(Object..., AdsErrorCode..\) \[▸ 1601\]](#)

Remarks

The returned values are automatically marshalled to their appropriate .NET types.

Reference

[SumSymbolRead Class \[▸ 1572\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▸ 1525\]](#)

[SumSymbolRead.Read. \[▸ 1577\]](#)

[SumSymbolRead.ReadAsync\(Cancellation token\) \[▸ 1578\]](#)

Also see about this

[ISumSymbolRead.TryRead Method \[▸ 1591\]](#)

6.5.11.2.4 SumSymbolRead.createResults Method

Creates the SumCommand results (abstract)

Namespace: [TwinCAT.Ads.SumCommand \[▸ 1525\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected override ResultValue2<ISymbol, Object>[] createResults(
    IList<SumDataEntity> infoList,
    IList<ReadOnlyMemory<byte>> readBlocks,
    AdsErrorCode[] subErrors
)
```

Parameters

infoList	Type: System.Collections.Generic.IList.SumDataEntity . The information list.
readBlocks	Type: System.Collections.Generic.IList.ReadOnlyMemory.Byte.. The read blocks.
subErrors	Type: .TwinCAT.Ads.AdsErrorCode [▶ 664]. The sub errors.

Return Value

Type: [.ResultValue2](#) [▶ 1234].[ISymbol](#) [▶ 2691], [Object..](#)
[ResultReadValueAccess2](#)[].

Reference

[SumSymbolRead Class](#) [▶ 1572]

[TwinCAT.Ads.SumCommand Namespace](#) [▶ 1525]

6.5.11.2.5 SumSymbolRead.Read2 Method

Reads the Values.

Namespace: [TwinCAT.Ads.SumCommand](#) [▶ 1525]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ResultSumValues2<ISymbol> Read2()
```

Return Value

Type: [ResultSumValues2](#) [▶ 1605].[ISymbol](#) [▶ 2691].
[System.Object](#)[].

Implements

[ISumRead2.S..Read2.](#) [▶ 1604]

Remarks

The return values are automatically marshalled to their appropriate .NET types.

Reference

[SumSymbolRead Class](#) [▶ 1572]

[TwinCAT.Ads.SumCommand Namespace](#) [▶ 1525]

[ISumRead2.S..Read2Async\(CancellationToken\) \[▸ 1605\]](#)

6.5.11.2.6 SumSymbolRead.Read2Async Method

Read2 as an asynchronous operation.

Namespace: [TwinCAT.Ads.SumCommand \[▸ 1525\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultSumValues2<ISymbol>> Read2Async (
    CancellationTokn cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationTokn The cancellation token.
--------	--

Return Value

Type: [Task.ResultSumValues2 \[▸ 1605\].ISymbol \[▸ 2691\]](#)..

A Task<ResultSumValues2`1> representing the asynchronous operation.

Implements

[ISumRead2.S..Read2Async\(CancellationToken\) \[▸ 1605\]](#)

Reference

[SumSymbolRead Class \[▸ 1572\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▸ 1525\]](#)

[ISumRead2.S..Read2. \[▸ 1604\]](#)

6.5.11.2.7 SumSymbolRead.ReadAsResult Method

Reads the combined (Sum) Symbols and returns them as [Result \[▸ 1547\]](#) object.

Namespace: [TwinCAT.Ads.SumCommand \[▸ 1525\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ResultSumValues ReadAsResult ()
```

Return Value

Type: [ResultSumValues \[▸ 1547\]](#)

The Result object.

Implements

[ISumRead.ReadAsResult. \[▸ 1600\]](#)

Reference

[SumSymbolRead Class \[► 1572\]](#)

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

6.5.12 SumSymbolWrite Class

Class for ADS Sum symbolic Write Access.

Inheritance Hierarchy

[System.Object](#)

SumCommandWrapper.SumWrite.

SumSymbolCommand.SumWrite.

TwinCAT.Ads.SumCommand.SumSymbolWrite

Namespace: [TwinCAT.Ads.SumCommand \[► 1525\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229










Syntax

C#

```
public class SumSymbolWrite : SumSymbolCommand<SumWrite>
```

The SumSymbolWrite type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryWrite [► 1585]	Writes the specified values.
	Write [► 1586]	Writes the specified values.
	WriteAsync [► 1587]	Reads the symbol values asynchronously.

Remarks

The SumSymbolWrite implements symbolic write access with automatic (dynamic) value marshalling. The advantage of the symbolic access is (in contrast to the handle access classes [SumHandleRead \[► 1557\]](#), [SumHandleWrite \[► 1561\]](#)) that all type information is available when using this ADS Sum Command. The disadvantage is, that the Symbolic information must be loaded beforehand, see examples. The SumSymbolWrite defaults to an IndexGroup/IndexOffset read ([IndexGroupIndexOffset \[► 1608\]](#)), because the the write access by InstancePath/SymbolName is not supported. The disadvantage of this mode is, that dereferencing References/Pointers don't work and their values cannot be resolved. An option is to use the [ValueByHandle \[► 1608\]](#), but that means to use more ADS roundtrips because the handles have to be requested first!

Examples

Usage of SumRead/SumSymbolWrite with AdsSession

```

/// <summary>
/// Defines the entry point of the application.
/// </summary>
/// <param name="args">The arguments.</param>
static async void Main(string[] args)
{
    Console.WriteLine("");
    Console.WriteLine("Press [Enter] for start:");
    Console.ReadLine();

    // Parse the command-line arguments
    AmsAddress address = ArgParser.Parse(args);

    CancellationTokenSource cancelSource = new CancellationTokenSource();
    CancellationToken cancel = cancelSource.Token;

    using (AdsSession session = new AdsSession(address))
    {
        // Connect to the device target.
        AdsConnection connection = (AdsConnection)session.Connect();

        // Load symbolic information
        var resultSymbols = await session.SymbolServer.GetSymbolsAsync(cancel);
        resultSymbols.ThrowOnError();

        ISymbolCollection<ISymbol> symbols = resultSymbols.Symbols;

        ISymbol bVar1 = symbols["GVL.bVar1"];
        ISymbol bVar2 = symbols["GVL.bVar2"];
        ISymbol projectName = symbols["TwinCAT_SystemInfoVarList._AppInfo.ProjectName"];

        SymbolCollection coll = new SymbolCollection() {bVar1, bVar2, projectName};

        // Sum Command Read
        SumSymbolRead readCommand = new SumSymbolRead(connection,coll);
        var resultReadValues = await readCommand.ReadAsync(cancel);

        if (resultReadValues.Succeeded)
        {
            object[] values = resultReadValues.Values;

            for (int i = 0; i < coll.Count; i++)
            {
                Console.WriteLine("Symbol: {0} (Value: {1}, Type: {2})", coll[i].InstancePath, values[i].ToString(), values[i].GetType().Name);
            }
        }
        // Sum Command Write

        SumSymbolWrite writeCommand = new SumSymbolWrite(connection,coll);
        object[] writeValues = new object[] {true, (short) 42, "MyNewProjectName"};

        var resultWrite = await writeCommand.WriteAsync(writeValues,cancel);
    }

    Console.WriteLine("");
    Console.WriteLine("Press [Enter] for leave:");
    Console.ReadLine();
}

```

Examples

Usage of SumRead/SumSymbolWrite with AdsClient

```

/// <summary>
/// Defines the entry point of the application.
/// </summary>
/// <param name="args">The arguments.</param>
static async void Main(string[] args)
{
    Console.WriteLine("");
    Console.WriteLine("Press [Enter] for start:");
    Console.ReadLine();
}

```

```

CancellationTokenSource cancelSource = new CancellationTok...
CancellationToken cancel = cancelSource.Token;

// Parse the command-line arguments
AmsAddress address = ArgParser.Parse(args);

using (AdsClient client = new AdsClient())
{
    // Connect the AdsClient to the device target.
    client.Connect(address);

    // Load symbolic information
    ISymbolLoader loader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    var resultReadSymbols = await loader.GetSymbolsAsync(cancel);

    resultReadSymbols.ThrowOnError();

    ISymbolCollection<ISymbol> allSymbols = resultReadSymbols.Symbols;

    ISymbol bVar1 = allSymbols["GVL.bVar1"];
    ISymbol bVar2 = allSymbols["GVL.iCount"];
    ISymbol projectName = allSymbols["TwinCAT_SystemInfoVarList._AppInfo.ProjectName"];

    SymbolCollection symbols = new SymbolCollection() {bVar1, bVar2, projectName};

    // Sum Command Read
    SumSymbolRead readCommand = new SumSymbolRead(client, symbols);
    var resultSumRead = await readCommand.ReadAsync(cancel);

    if (resultSumRead.Succeeded)
    {
        object[] values = resultSumRead.Values;

        for (int i = 0; i < symbols.Count; i++)
        {
            Console.WriteLine("Symbol: {0} (Value: {1}, Type: {2})", symbols[i].InstancePath, values[i].
ToString(), values[i].GetType().Name);
        }

        // Sum Command Write
        SumSymbolWrite writeCommand = new SumSymbolWrite(client, symbols);
        object[] writeValues = new object[] { true, (short)42, "MyNewProjectName" };

        var resultSumWrite = await writeCommand.WriteAsync(writeValues, cancel);
    }
}

Console.WriteLine("");
Console.WriteLine("Press [Enter] for leave:");
Console.ReadLine();
}

```

Reference

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

[TwinCAT.Ads.SumCommand.ISumCommand \[► 1527\]](#)

TwinCAT.Ads.SumCommand.SumSymbolWrite

6.5.12.1 SumSymbolWrite Constructor

Initializes a new instance of the [SumSymbolWrite \[► 1582\]](#) class.

Namespace: [TwinCAT.Ads.SumCommand \[► 1525\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public SumSymbolWrite(
    IAdsConnection connection,
    IList<ISymbol> symbols
)
```

Parameters

- connection Type: [TwinCAT.Ads.IAdsConnection](#) [▶ 876]
The ADS Connection.
- symbols Type: [System.Collections.Generic.IList.ISymbol](#) [▶ 2691].
The symbols to read










Reference

- [SumSymbolWrite Class](#) [▶ 1582]
- [TwinCAT.Ads.SumCommand Namespace](#) [▶ 1525]

6.5.12.2 SumSymbolWrite Methods

The [SumSymbolWrite](#) [▶ 1582] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryWrite [▶ 1585]	Writes the specified values.
	Write [▶ 1586]	Writes the specified values.
	WriteAsync [▶ 1587]	Reads the symbol values asynchronously.

Reference

- [SumSymbolWrite Class](#) [▶ 1582]
- [TwinCAT.Ads.SumCommand Namespace](#) [▶ 1525]

6.5.12.2.1 SumSymbolWrite.TryWrite Method

Writes the specified values.

Namespace: [TwinCAT.Ads.SumCommand](#) [▶ 1525]
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsErrorCode TryWrite(
    Object[] values,
    out AdsErrorCode[]?? returnCodes
)
```

Parameters

values	Type: .System.Object . The values.
returnCodes	Type: .TwinCAT.Ads.AdsErrorCode [▶ 664].. The return codes.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
AdsErrorCode.

Remarks

The written values will be marshalled automatically to their appropriate ADS types.

Reference

[SumSymbolWrite Class](#) [[▶ 1582](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

[SumSymbolWrite.Write\(Object.\)](#) [[▶ 1586](#)]

[SumSymbolWrite.WriteAsync\(Object., CancellationToken\)](#) [[▶ 1587](#)]

6.5.12.2.2 SumSymbolWrite.Write Method

Writes the specified values.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Write(
    Object[] values
)
```

Parameters

values	Type: .System.Object . The values.
--------	---

Exceptions

Exception	Condition
AdsSumCommandException [▶ 735]	SumSymbolWrite failed!

Remarks

The values will be marshalled automatically to their appropriate ADS types.

Reference

[SumSymbolWrite Class \[► 1582\]](#)

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

[SumSymbolWrite.TryWrite\(Object, AdsErrorCode..\) \[► 1585\]](#)

[SumSymbolWrite.WriteAsync\(Object, CancellationToken\) \[► 1587\]](#)

6.5.12.2.3 SumSymbolWrite.WriteAsync Method

Reads the symbol values asynchronously.

Namespace: [TwinCAT.Ads.SumCommand \[► 1525\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultSumCommand> WriteAsync (
    Object[] values,
    CancellationToken cancel
)
```

Parameters

values	Type: System.Object . The values.
cancel	Type: System.Threading.CancellationToken The cancellation token

Return Value

Type: [Task.ResultSumCommand \[► 1533\]](#).

An asynchronous task that represents the 'Write' operation and returns a [ResultSumCommand \[► 1533\]](#). The overall error return code is contained in the [ErrorCode \[► 1120\]](#) property.

Reference

[SumSymbolWrite Class \[► 1582\]](#)

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

[SumSymbolWrite.Write\(Object\) \[► 1586\]](#)

[SumSymbolWrite.TryWrite\(Object, AdsErrorCode..\) \[► 1585\]](#)

6.5.13 ISumSymbolRead Interface

Interface [ISumSymbolRead](#) Implements the [ISumCommand \[► 1527\]](#)

Namespace: [TwinCAT.Ads.SumCommand \[► 1525\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3






Syntax

C#




```
public interface ISumSymbolRead : ISumCommand
```

The ISumSymbolRead type exposes the following members.



Properties

	Name	Description
	Executed [▶ 1529]	Gets a value indicating whether this ISumCommand [▶ 1527] was already executed. (Inherited from ISumCommand [▶ 1527] .)
	Failed [▶ 1529]	Gets a value indicating whether this ISumCommand [▶ 1527] failed. (Inherited from ISumCommand [▶ 1527] .)
	Result [▶ 1530]	Gets the AdsErrorCode [▶ 664] of the main SumCommandBase ADS Request (Inherited from ISumCommand [▶ 1527] .)
	SubResults [▶ 1530]	Gets the sub results of the single Sub Requests. (Inherited from ISumCommand [▶ 1527] .)
	Succeeded [▶ 1531]	Gets a value indicating whether this ISumCommand [▶ 1527] is succeeded. (Inherited from ISumCommand [▶ 1527] .)

Methods

	Name	Description
	Read [▶ 1589]	Reads the Values.
	ReadAsync [▶ 1590]	Reads Symbol values as an asynchronous operation.
	TryRead [▶ 1591]	Reads the specified symbols.

Extension Methods

	Name	Description
	PollValues(IObservable.Unit.) [▶ 1352]	Overloaded. Polls a series of symbols via a ISumSymbolRead command. The SumCommand will read all contained values with every trigger. (Defined by ValueSymbolExtensions [▶ 1333] .)
	PollValues(TimeSpan) [▶ 1353]	Overloaded. Polls a series of symbols via a ISumSymbolRead command. The SumCommand will read all contained values with every trigger. (Defined by ValueSymbolExtensions [▶ 1333] .)

Reference






[TwinCAT.Ads.SumCommand Namespace \[▶ 1525\]](#)

[TwinCAT.Ads.SumCommand.ISumCommand \[▶ 1527\]](#)

6.5.13.1 ISumSymbolRead Properties

The [ISumSymbolRead \[▶ 1587\]](#) type exposes the following members.

Properties

	Name	Description
	Executed [▸ 1529]	Gets a value indicating whether this ISumCommand [▸ 1527] was already executed. (Inherited from ISumCommand [▸ 1527] .)
	Failed [▸ 1529]	Gets a value indicating whether this ISumCommand [▸ 1527] failed. (Inherited from ISumCommand [▸ 1527] .)
	Result [▸ 1530]	Gets the AdsErrorCode [▸ 664] of the main SumCommandBase ADS Request (Inherited from ISumCommand [▸ 1527] .)
	SubResults [▸ 1530]	Gets the sub results of the single Sub Requests. (Inherited from ISumCommand [▸ 1527] .)
	Succeeded [▸ 1531]	Gets a value indicating whether this ISumCommand [▸ 1527] is succeeded. (Inherited from ISumCommand [▸ 1527] .)

Reference




[ISumSymbolRead Interface \[▸ 1587\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▸ 1525\]](#)



6.5.13.2 ISumSymbolRead Methods

The [ISumSymbolRead \[▸ 1587\]](#) type exposes the following members.

Methods

	Name	Description
	Read [▸ 1589]	Reads the Values.
	ReadAsync [▸ 1590]	Reads Symbol values as an asynchronous operation.
	TryRead [▸ 1591]	Reads the specified symbols.

Extension Methods

	Name	Description
	PollValues(IObservableUnit.) [▸ 1352]	Overloaded. Polls a series of symbols via a ISumSymbolRead [▸ 1587] command. The SumCommand will read all contained values with every trigger. (Defined by ValueSymbolExtensions [▸ 1333] .)
	PollValues(TimeSpan) [▸ 1353]	Overloaded. Polls a series of symbols via a ISumSymbolRead [▸ 1587] command. The SumCommand will read all contained values with every trigger. (Defined by ValueSymbolExtensions [▸ 1333] .)

Reference

[ISumSymbolRead Interface \[▸ 1587\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▸ 1525\]](#)

6.5.13.2.1 ISumSymbolRead.Read Method

Reads the Values.

Namespace: [TwinCAT.Ads.SumCommand](#) [► 1525]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
Object[] Read()
```

Return Value

Type: [.Object](#).
[System.Object\[\]](#).

Remarks

The return values are automatically marshalled to their appropriate .NET types.

Reference

[ISumSymbolRead Interface](#) [► 1587]

[TwinCAT.Ads.SumCommand Namespace](#) [► 1525]

[ISumSymbolRead.TryRead\(Object..., AdsErrorCode.\)](#) [► 1591]

[ISumSymbolRead.ReadAsync\(CancellationToken\)](#) [► 1590]

6.5.13.2.2 ISumSymbolRead.ReadAsync Method

Reads Symbol values as an asynchronous operation.

Namespace: [TwinCAT.Ads.SumCommand](#) [► 1525]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
Task<ResultSumValues> ReadAsync(  
    CancellationToken cancel  
)
```

Parameters

cancel Type: [System.Threading.CancellationToken](#)
The cancellation token.

Return Value

Type: [Task.ResultSumValues](#) [► 1547].

An asynchronous task that represents the 'Read' operation and returns a [ResultSumValues](#) [► 1547]. The overall error return code is contained in the [ErrorCode](#) [► 1120] property.

Reference

[ISumSymbolRead Interface](#) [► 1587]

[TwinCAT.Ads.SumCommand Namespace](#) [► 1525]

[ISumSymbolRead.Read.](#) [► 1589]

[ISumSymbolRead.TryRead\(Object..., AdsErrorCode.\)](#) [► 1591]

6.5.13.2.3 ISumSymbolRead.TryRead Method

Reads the specified symbols.

Namespace: [TwinCAT.Ads.SumCommand](#) [► 1525]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
AdsErrorCode TryRead(  
    out Object[]? values,  
    out AdsErrorCode[]?? returnCodes  
)
```

Parameters

values	Type: .System.Object. The values.
returnCodes	Type: .TwinCAT.Ads.AdsErrorCode [► 664]. The return codes.

Return Value

Type: [AdsErrorCode](#) [► 664]
AdsErrorCode.

Remarks

The returned values are automatically marshalled to their appropriate .NET types.

Reference

[ISumSymbolRead Interface](#) [► 1587]

[TwinCAT.Ads.SumCommand Namespace](#) [► 1525]

[ISumSymbolRead.Read.](#) [► 1589]

[ISumSymbolRead.ReadAsync\(CancellationToken\)](#) [► 1590]

6.5.14 ISumCommandExtension Class

Extension class for [ISumCommand](#) [► 1527] derived classes.

Inheritance Hierarchy

[System.Object](#)
TwinCAT.Ads.SumCommand.ISumCommandExtension

Namespace: [TwinCAT.Ads.SumCommand](#) [► 1525]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static class ISumCommandExtension
```

The ISumCommandExtension type exposes the following members.

Methods

	Name	Description
	AllFailed [▶ 1592]	Gets a value indicating, that all SubCommands failed.
	AllSucceeded [▶ 1593]	Gets a value indicating, that all SubCommands succeeded.
	FirstSubError [▶ 1594]	Gets the first SubError that is not NoError [▶ 664]
	OneFailed [▶ 1594]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results).
	OneSucceeded [▶ 1595]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands)
	OverallError [▶ 1595]	Gets the Overall (combined error) from SumCommand AND SubCommands.

Reference

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.14.1 ISumCommandExtension Methods

The [ISumCommandExtension](#) [[▶ 1591](#)] type exposes the following members.

Methods

	Name	Description
	AllFailed [▶ 1592]	Gets a value indicating, that all SubCommands failed.
	AllSucceeded [▶ 1593]	Gets a value indicating, that all SubCommands succeeded.
	FirstSubError [▶ 1594]	Gets the first SubError that is not NoError [▶ 664]
	OneFailed [▶ 1594]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results).
	OneSucceeded [▶ 1595]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands)
	OverallError [▶ 1595]	Gets the Overall (combined error) from SumCommand AND SubCommands.

Reference

[ISumCommandExtension Class](#) [[▶ 1591](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.14.1.1 ISumCommandExtension.AllFailed Method

Gets a value indicating, that all SubCommands failed.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool AllFailed(  
    this ISumCommand sumCommand  
)
```

Parameters

sumCommand	Type: TwinCAT.Ads.SumCommand.ISumCommand [▸ 1527] The sum command.
------------	---

Return Value

Type: [Boolean](#)

true if the SumCommand and all SubCommands failed, false otherwise.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [ISumCommand \[▸ 1527\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[ISumCommandExtension Class \[▸ 1591\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▸ 1525\]](#)

6.5.14.1.2 ISumCommandExtension.AllSucceeded Method

Gets a value indicating, that all SubCommands succeeded.

Namespace: [TwinCAT.Ads.SumCommand \[▸ 1525\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool AllSucceeded(  
    this ISumCommand sumCommand  
)
```

Parameters

sumCommand	Type: TwinCAT.Ads.SumCommand.ISumCommand [▸ 1527] The sum command.
------------	---

Return Value

Type: [Boolean](#)

true if the SumCommand and all SubCommands succeeded, false otherwise.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [ISumCommand \[▸ 1527\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference[ISumCommandExtension Class \[► 1591\]](#)[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)**6.5.14.1.3 ISumCommandExtension.FirstSubError Method**Gets the first SubError that is not [NoError \[► 664\]](#)**Namespace:** [TwinCAT.Ads.SumCommand \[► 1525\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static AdsErrorCode FirstSubError(
    this ISumCommand sumCommand
)
```

Parameters

sumCommand	Type: TwinCAT.Ads.SumCommand.ISumCommand [► 1527]
------------	---

Field ValueType: [AdsErrorCode \[► 664\]](#)

The first subError.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [ISumCommand \[► 1527\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference[ISumCommandExtension Class \[► 1591\]](#)[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)**6.5.14.1.4 ISumCommandExtension.OneFailed Method**

Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results).

Namespace: [TwinCAT.Ads.SumCommand \[► 1525\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static bool OneFailed(
    this ISumCommand sumCommand
)
```

Parameters

sumCommand	Type: TwinCAT.Ads.SumCommand.ISumCommand [► 1527]
------------	---

Field Value

Type: [Boolean](#)
true if [overall failed]; otherwise, false.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [ISumCommand](#) [[▶ 1527](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[ISumCommandExtension Class](#) [[▶ 1591](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.14.1.5 ISumCommandExtension.OneSucceeded Method

Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands)

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool OneSucceeded(  
    this ISumCommand sumCommand  
)
```

Parameters

sumCommand	Type: TwinCAT.Ads.SumCommand.ISumCommand [▶ 1527]
------------	---

Field Value

Type: [Boolean](#)
true if [sub results succeeded]; otherwise, false.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [ISumCommand](#) [[▶ 1527](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[ISumCommandExtension Class](#) [[▶ 1591](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.14.1.6 ISumCommandExtension.OverallError Method

Gets the Overall (combined error) from SumCommand AND SubCommands.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AdsErrorCode OverallError(
    this ISumCommand sumCommand
)
```

Parameters

sumCommand	Type: TwinCAT.Ads.SumCommand.ISumCommand [► 1527]
------------	---

Field Value

Type: [AdsErrorCode](#) [► 664]

The combined error or [NoError](#) [► 664]

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [ISumCommand](#) [► 1527]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

This will return the ErrorCode of the Sumcommand (if failed) or the first failed subcommand.

Reference

[ISumCommandExtension Class](#) [► 1591]

[TwinCAT.Ads.SumCommand Namespace](#) [► 1525]

6.5.15 ISumRead Interface

Interface ISumRead Implements the [ISumCommand](#) [► 1527]

Namespace: [TwinCAT.Ads.SumCommand](#) [► 1525]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public interface ISumRead : ISumCommand
```

The ISumRead type exposes the following members.

Properties

	Name	Description
	Connection [► 1531]	The connection used for communication. (Inherited from ISumCommand [► 1527].)
	ErrorStrategy [► 1531]	Gets the error strategy of the ISumCommand [► 1527] (Inherited from ISumCommand [► 1527].)
	Executed [► 1529]	Gets a value indicating whether this ISumCommand [► 1527] was already executed. (Inherited from ISumCommand [► 1527].)
	Failed [► 1529]	Gets a value indicating whether this ISumCommand [► 1527] failed. (Inherited from ISumCommand [► 1527].)

	Name	Description
	FailedCount [▶ 1532]	Gets the count of failed subcommands. (Inherited from ISumCommand [▶ 1527].)
	Result [▶ 1530]	Gets the AdsErrorCode [▶ 664] of the main SumCommandBase ADS Request (Inherited from ISumCommand [▶ 1527].)
	SubResults [▶ 1530]	Gets the sub results of the single Sub Requests. (Inherited from ISumCommand [▶ 1527].)
	Succeeded [▶ 1531]	Gets a value indicating whether this ISumCommand [▶ 1527] is succeeded. (Inherited from ISumCommand [▶ 1527].)
	SucceededCount [▶ 1532]	Gets the count of succeeded subcommands. (Inherited from ISumCommand [▶ 1527].)

Methods

	Name	Description
	Read [▶ 1599]	Reads the combined (Sum) Symbols and returns them as value array.
	ReadAsResult [▶ 1600]	Reads the combined (Sum) Symbols and returns them as Result [▶ 1547] object.
	ReadAsync [▶ 1600]	Reads Symbol values as an asynchronous operation.
	TryRead [▶ 1601]	Reads the specified symbols.

Extension Methods

	Name	Description
	AllFailed [▶ 1592]	Gets a value indicating, that all SubCommands failed. (Defined by ISumCommandExtension [▶ 1591].)
	AllSucceeded [▶ 1593]	Gets a value indicating, that all SubCommands succeeded. (Defined by ISumCommandExtension [▶ 1591].)
	FirstSubError [▶ 1594]	Gets the first SubError that is not NoError [▶ 664] (Defined by ISumCommandExtension [▶ 1591].)
	OneFailed [▶ 1594]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Defined by ISumCommandExtension [▶ 1591].)
	OneSucceeded [▶ 1595]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Defined by ISumCommandExtension [▶ 1591].)
	OverallError [▶ 1595]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Defined by ISumCommandExtension [▶ 1591].)
	PollValues(IObservableUnit.) [▶ 1359]	Overloaded. Polls a series of symbols via a ISumRead command. The SumCommand will read all contained values with every trigger. (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues(TimeSpan) [▶ 1361]	Overloaded. Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger in one roundtrip. (Defined by ValueSymbolExtensions [▶ 1333].)

Reference

[TwinCAT.Ads.SumCommand Namespace](#) [▶ 1525]

[TwinCAT.Ads.SumCommand.ISumCommand](#) [▶ 1527]

[TwinCAT.Ads.SumCommand.ISumRead2.S.](#) [▶ 1601]

6.5.15.1 ISumRead Properties

The [ISumRead \[▸ 1596\]](#) type exposes the following members.

Properties

	Name	Description
	Connection [▸ 1531]	The connection used for communication. (Inherited from ISumCommand [▸ 1527] .)
	ErrorStrategy [▸ 1531]	Gets the error strategy of the ISumCommand [▸ 1527] (Inherited from ISumCommand [▸ 1527] .)
	Executed [▸ 1529]	Gets a value indicating whether this ISumCommand [▸ 1527] was already executed. (Inherited from ISumCommand [▸ 1527] .)
	Failed [▸ 1529]	Gets a value indicating whether this ISumCommand [▸ 1527] failed. (Inherited from ISumCommand [▸ 1527] .)
	FailedCount [▸ 1532]	Gets the count of failed subcommands. (Inherited from ISumCommand [▸ 1527] .)
	Result [▸ 1530]	Gets the AdsErrorCode [▸ 664] of the main SumCommandBase ADS Request (Inherited from ISumCommand [▸ 1527] .)
	SubResults [▸ 1530]	Gets the sub results of the single Sub Requests. (Inherited from ISumCommand [▸ 1527] .)
	Succeeded [▸ 1531]	Gets a value indicating whether this ISumCommand [▸ 1527] is succeeded. (Inherited from ISumCommand [▸ 1527] .)
	SucceededCount [▸ 1532]	Gets the count of succeeded subcommands. (Inherited from ISumCommand [▸ 1527] .)

Reference

[ISumRead Interface \[▸ 1596\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▸ 1525\]](#)

6.5.15.2 ISumRead Methods

The [ISumRead \[▸ 1596\]](#) type exposes the following members.

Methods

	Name	Description
	Read [▸ 1599]	Reads the combined (Sum) Symbols and returns them as value array.
	ReadAsResult [▸ 1600]	Reads the combined (Sum) Symbols and returns them as Result [▸ 1547] object.
	ReadAsync [▸ 1600]	Reads Symbol values as an asynchronous operation.
	TryRead [▸ 1601]	Reads the specified symbols.

Extension Methods

	Name	Description
	AllFailed [▸ 1592]	Gets a value indicating, that all SubCommands failed. (Defined by ISumCommandExtension [▸ 1591] .)
	AllSucceeded [▸ 1593]	Gets a value indicating, that all SubCommands succeeded. (Defined by ISumCommandExtension [▸ 1591] .)

	Name	Description
	FirstSubError [▶ 1594]	Gets the first SubError that is not NoError [▶ 664] (Defined by ISumCommandExtension [▶ 1591].)
	OneFailed [▶ 1594]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Defined by ISumCommandExtension [▶ 1591].)
	OneSucceeded [▶ 1595]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Defined by ISumCommandExtension [▶ 1591].)
	OverallError [▶ 1595]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Defined by ISumCommandExtension [▶ 1591].)
	PollValues(IObservable.Unit.) [▶ 1359]	Overloaded. Polls a series of symbols via a ISumRead [▶ 1596] command. The SumCommand will read all contained values with every trigger. (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues(TimeSpan) [▶ 1361]	Overloaded. Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger in one roundtrip. (Defined by ValueSymbolExtensions [▶ 1333].)

Reference

[ISumRead Interface](#) [▶ 1596]

[TwinCAT.Ads.SumCommand Namespace](#) [▶ 1525]

6.5.15.2.1 ISumRead.Read Method

Reads the combined (Sum) Symbols and returns them as value array.

Namespace: [TwinCAT.Ads.SumCommand](#) [▶ 1525]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object[] Read()
```

Return Value

Type: `.Object`.
The value array.

Remarks

The return values are automatically marshalled to their appropriate .NET types. This method throws Exceptions, when the Read access fails.

Reference

[ISumRead Interface](#) [▶ 1596]

[TwinCAT.Ads.SumCommand Namespace](#) [▶ 1525]

[ISumRead.TryRead\(Object., AdsErrorCode.\)](#) [▶ 1601]

[ISumRead.ReadAsync\(Cancellation token\)](#) [▶ 1600]

6.5.15.2.2 ISumRead.ReadAsResult Method

Reads the combined (Sum) Symbols and returns them as [Result \[▸ 1547\]](#) object.

Namespace: [TwinCAT.Ads.SumCommand \[▸ 1525\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ResultSumValues ReadAsResult ()
```

Return Value

Type: [ResultSumValues \[▸ 1547\]](#)

The Result object.

Reference

[ISumRead Interface \[▸ 1596\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▸ 1525\]](#)

6.5.15.2.3 ISumRead.ReadAsync Method

Reads Symbol values as an asynchronous operation.

Namespace: [TwinCAT.Ads.SumCommand \[▸ 1525\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultSumValues> ReadAsync (
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultSumValues \[▸ 1547\]](#).

An asynchronous task that represents the 'Read' operation and returns a [ResultSumValues \[▸ 1547\]](#). The overall error return code is contained in the [ErrorCode \[▸ 1120\]](#) property.

Reference

[ISumRead Interface \[▸ 1596\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▸ 1525\]](#)

[ISumRead.Read. \[▸ 1599\]](#)

[ISumRead.TryRead\(.Object., .AdsErrorCode..\) \[▸ 1601\]](#)

6.5.15.2.4 ISumRead.TryRead Method

Reads the specified symbols.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryRead(
    out Object[]? values,
    out AdsErrorCode[]?? returnCodes
)
```

Parameters

values	Type: .System.Object. The values.
returnCodes	Type: .TwinCAT.Ads.AdsErrorCode [▶ 664]. The return codes.

Return Value

Type: [AdsErrorCode](#) [[▶ 664](#)]
AdsErrorCode.

Remarks

The returned values are automatically marshalled to their appropriate .NET types.

Reference

[ISumRead Interface](#) [[▶ 1596](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

[ISumRead.Read.](#) [[▶ 1599](#)]

[ISumRead.ReadAsync\(CancellationTokens\)](#) [[▶ 1600](#)]

6.5.16 ISumRead2.S. Interface

Interface ISumRead2 (supports extended Sum SubResults) Implements the [ISumCommand](#) [[▶ 1527](#)]

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public interface ISumRead2<S> : ISumCommand
```

Type Parameters

S

The ISumRead2.S. type exposes the following members.

Properties

Name	Description
Connection [▶ 1531]	The connection used for communication. (Inherited from ISumCommand [▶ 1527].)
ErrorStrategy [▶ 1531]	Gets the error strategy of the ISumCommand [▶ 1527] (Inherited from ISumCommand [▶ 1527].)
Executed [▶ 1529]	Gets a value indicating whether this ISumCommand [▶ 1527] was already executed. (Inherited from ISumCommand [▶ 1527].)
Failed [▶ 1529]	Gets a value indicating whether this ISumCommand [▶ 1527] failed. (Inherited from ISumCommand [▶ 1527].)
FailedCount [▶ 1532]	Gets the count of failed subcommands. (Inherited from ISumCommand [▶ 1527].)
Result [▶ 1530]	Gets the AdsErrorCode [▶ 664] of the main SumCommandBase ADS Request (Inherited from ISumCommand [▶ 1527].)
SubResults [▶ 1530]	Gets the sub results of the single Sub Requests. (Inherited from ISumCommand [▶ 1527].)
Succeeded [▶ 1531]	Gets a value indicating whether this ISumCommand [▶ 1527] is succeeded. (Inherited from ISumCommand [▶ 1527].)
SucceededCount [▶ 1532]	Gets the count of succeeded subcommands. (Inherited from ISumCommand [▶ 1527].)

Methods

Name	Description
Read2 [▶ 1604]	Reads the Values.
Read2Async [▶ 1605]	Reads Symbol values as an asynchronous operation.

Extension Methods

Name	Description
AllFailed [▶ 1592]	Gets a value indicating, that all SubCommands failed. (Defined by ISumCommandExtension [▶ 1591].)
AllSucceeded [▶ 1593]	Gets a value indicating, that all SubCommands succeeded. (Defined by ISumCommandExtension [▶ 1591].)
FirstSubError [▶ 1594]	Gets the first SubError that is not NoError [▶ 664] (Defined by ISumCommandExtension [▶ 1591].)
OneFailed [▶ 1594]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Defined by ISumCommandExtension [▶ 1591].)
OneSucceeded [▶ 1595]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Defined by ISumCommandExtension [▶ 1591].)
OverallError [▶ 1595]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Defined by ISumCommandExtension [▶ 1591].)
PollValues2.S. (IObservable.Unit) [▶ 1370]	Overloaded. Polls a series of symbols via a ISumRead2.S. command. The SumCommand will read all contained values with every trigger in one roundtrip. (Defined by ValueSymbolExtensions [▶ 1333].)
PollValues2.S. (TimeSpan) [▶ 1372]	Overloaded. Polls a series of symbols via a ISumRead2.S. command. The SumCommand will read all contained values with every trigger. (Defined by ValueSymbolExtensions [▶ 1333].)

Remarks

Implementations of this interface return enriched SubResults on SumCommand Reads, that include the original sourceId in its result to backtrace.

Reference

[TwinCAT.Ads.SumCommand Namespace \[▶ 1525\]](#)

[TwinCAT.Ads.SumCommand.ISumCommand \[▶ 1527\]](#)

6.5.16.1 ISumRead2.S. Properties

The [ISumRead2.S. \[▶ 1601\]](#) generic type exposes the following members.

Properties

Name	Description
Connection [▶ 1531]	The connection used for communication. (Inherited from ISumCommand [▶ 1527].)
ErrorStrategy [▶ 1531]	Gets the error strategy of the ISumCommand [▶ 1527] (Inherited from ISumCommand [▶ 1527].)
Executed [▶ 1529]	Gets a value indicating whether this ISumCommand [▶ 1527] was already executed. (Inherited from ISumCommand [▶ 1527].)
Failed [▶ 1529]	Gets a value indicating whether this ISumCommand [▶ 1527] failed. (Inherited from ISumCommand [▶ 1527].)
FailedCount [▶ 1532]	Gets the count of failed subcommands. (Inherited from ISumCommand [▶ 1527].)
Result [▶ 1530]	Gets the AdsErrorCode [▶ 664] of the main SumCommandBase ADS Request (Inherited from ISumCommand [▶ 1527].)
SubResults [▶ 1530]	Gets the sub results of the single Sub Requests. (Inherited from ISumCommand [▶ 1527].)
Succeeded [▶ 1531]	Gets a value indicating whether this ISumCommand [▶ 1527] is succeeded. (Inherited from ISumCommand [▶ 1527].)
SucceededCount [▶ 1532]	Gets the count of succeeded subcommands. (Inherited from ISumCommand [▶ 1527].)

Reference

[ISumRead2.S. Interface \[▶ 1601\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▶ 1525\]](#)

6.5.16.2 ISumRead2.S. Methods

The [ISumRead2.S. \[▶ 1601\]](#) generic type exposes the following members.

Methods

Name	Description
Read2 [▶ 1604]	Reads the Values.
Read2Async [▶ 1605]	Reads Symbol values as an asynchronous operation.

Extension Methods

	Name	Description
	AllFailed [▶ 1592]	Gets a value indicating, that all SubCommands failed. (Defined by ISumCommandExtension [▶ 1591].)
	AllSucceeded [▶ 1593]	Gets a value indicating, that all SubCommands succeeded. (Defined by ISumCommandExtension [▶ 1591].)
	FirstSubError [▶ 1594]	Gets the first SubError that is not NoError [▶ 664] (Defined by ISumCommandExtension [▶ 1591].)
	OneFailed [▶ 1594]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Defined by ISumCommandExtension [▶ 1591].)
	OneSucceeded [▶ 1595]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Defined by ISumCommandExtension [▶ 1591].)
	OverallError [▶ 1595]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Defined by ISumCommandExtension [▶ 1591].)
	PollValues2.S. (IObservable.Unit.) [▶ 1370]	Overloaded. Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger in one roundtrip. (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues2.S. (TimeSpan) [▶ 1372]	Overloaded. Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger. (Defined by ValueSymbolExtensions [▶ 1333].)

Reference

[ISumRead2.S. Interface](#) [[▶ 1601](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.16.2.1 ISumRead2.S..Read2 Method

Reads the Values.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
ResultSumValues2<S> Read2 ()
```

Return Value

Type: [ResultSumValues2](#) [[▶ 1605](#)].[S](#) [[▶ 1601](#)].
System.Object[].

Remarks

The return values are automatically marshalled to their appropriate .NET types.

Reference

[ISumRead2.S. Interface](#) [[▶ 1601](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

[ISumRead2.S..Read2Async\(CancellationToken\) \[► 1605\]](#)

6.5.16.2.2 ISumRead2.S..Read2Async Method

Reads Symbol values as an asynchronous operation.

Namespace: [TwinCAT.Ads.SumCommand \[► 1525\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultSumValues2<S>> Read2Async (
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultSumValues2 \[► 1605\].S \[► 1601\]](#)..

An asynchronous task that represents the 'Read' operation and returns a [ResultSumValues \[► 1547\]](#). The overall error return code is contained in the [ErrorCode \[► 1120\]](#) property.

Reference

[ISumRead2.S. Interface \[► 1601\]](#)

[TwinCAT.Ads.SumCommand Namespace \[► 1525\]](#)

[ISumRead2.S..Read2. \[► 1604\]](#)

6.5.17 ResultSumValues2.S. Class

Extended result class for an asynchronous [ISumRead2.S. \[► 1601\]](#).

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.ResultAds \[► 1116\]](#)

[TwinCAT.Ads.SumCommand.ResultSumCommand \[► 1533\]](#)

[TwinCAT.Ads.SumCommand.ResultSumValues2.S.](#)

Namespace: [TwinCAT.Ads.SumCommand \[► 1525\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public class ResultSumValues2<S> : ResultSumCommand
```

Type Parameters

S

The ResultSumValues2.S. type exposes the following members.

Properties

Name	Description
ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
FirstSubError [▶ 1535]	Gets the first SubError that is not NoError [▶ 664] (Inherited from ResultSumCommand [▶ 1533].)
Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
OverallError [▶ 1536]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Inherited from ResultSumCommand [▶ 1533].)
OverallFailed [▶ 1536]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Inherited from ResultSumCommand [▶ 1533].)
OverallSucceeded [▶ 1537]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Inherited from ResultSumCommand [▶ 1533].)
SubErrors [▶ 1535]	Gets the Error codes [▶ 664] for the single SumCommands. (Inherited from ResultSumCommand [▶ 1533].)
Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)
ValueResults [▶ 1607]	Gets the values results (Values extended by the Symbol reference and error code).

Methods

Name	Description
Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
GetHashCode	Serves as the default hash function. (Inherited from Object .)
GetType	Gets the Type of the current instance. (Inherited from Object .)
MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
ToString	Returns a string that represents the current object. (Inherited from Object .)

Remarks

This result class contains additional mapped [ISymbol](#) [[▶ 2691](#)] information foreach subResult of the the SumCommand.

Reference

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

[TwinCAT.Ads.SumCommand.ResultSumValues](#) [[▶ 1547](#)]

6.5.17.1 ResultSumValues2.S. Properties

The [ResultSumValues2.S. \[▸ 1605\]](#) generic type exposes the following members.

Properties

	Name	Description
	ErrorCode [▸ 1120]	Gets the ADS Error code bound to this Result [▸ 1116] object. (Inherited from ResultAds [▸ 1116] .)
	Failed [▸ 1121]	Gets a value indicating whether the ResultAds [▸ 1116] state is failed. (Inherited from ResultAds [▸ 1116] .)
	FirstSubError [▸ 1535]	Gets the first SubError that is not NoError [▸ 664] (Inherited from ResultSumCommand [▸ 1533] .)
	Invokeld [▸ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▸ 1116] .)
	OverallError [▸ 1536]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Inherited from ResultSumCommand [▸ 1533] .)
	OverallFailed [▸ 1536]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Inherited from ResultSumCommand [▸ 1533] .)
	OverallSucceeded [▸ 1537]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Inherited from ResultSumCommand [▸ 1533] .)
	SubErrors [▸ 1535]	Gets the Error codes [▸ 664] for the single SumCommands. (Inherited from ResultSumCommand [▸ 1533] .)
	Succeeded [▸ 1121]	Gets a value indicating whether the ResultAds [▸ 1116] state is succeeded. (Inherited from ResultAds [▸ 1116] .)
	ValueResults [▸ 1607]	Gets the values results (Values extended by the Symbol reference and error code).

Reference

[ResultSumValues2.S. Class \[▸ 1605\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▸ 1525\]](#)

6.5.17.1.1 ResultSumValues2.S..ValueResults Property

Gets the values results (Values extended by the Symbol reference and error code).

Namespace: [TwinCAT.Ads.SumCommand \[▸ 1525\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ResultValue2<S, Object>[]? ValueResults { get; }
```

Property Value

Type: [.ResultValue2 \[▸ 1234\].S \[▸ 1605\]](#), [Object..](#)

The value results.

Reference

[ResultSumValues2.S. Class \[▸ 1605\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▸ 1525\]](#)

6.5.17.2 ResultSumValues2.S. Methods

The [ResultSumValues2.S. \[▸ 1605\]](#) generic type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▸ 1128]	Sets the error state of this ResultAds [▸ 1116] (Inherited from ResultAds [▸ 1116] .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ResultSumValues2.S. Class \[▸ 1605\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▸ 1525\]](#)

6.5.18 SumAccessMode Enumeration

Indicates how to handle each single sub-sum-command (IGIO, ByHandle, ByName)

Namespace: [TwinCAT.Ads.SumCommand \[▸ 1525\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum SumAccessMode
```

Members

	Member name	Value	Description
	IndexGroupIndexOffset	0	Access by IndexGroup / IndexOffset
	ValueByHandle	1	Accesses a value by handle
	ValueByName	2	Access a value by name
	AcquireHandleByName	3	Acquire handle by name
	ReleaseHandle	4	Release handle

Reference

[TwinCAT.Ads.SumCommand Namespace \[▸ 1525\]](#)

6.5.19 SumAnyTypeRead Class

Symbolic ADS Sum read access (ANY_TYPE marshalling) on already loaded symbols.

Inheritance Hierarchy

System.Object

SumCommandWrapper.SumReadWrite.

SumSymbolCommand.SumReadWrite.

SumSymbolReadBase

TwinCAT.Ads.SumCommand.SumAnyTypeRead

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public class SumAnyTypeRead : SumSymbolReadBase,
    ISumRead2<ISymbol>, ISumCommand
```

The SumAnyTypeRead type exposes the following members.

Constructors

	Name	Description
	SumAnyTypeRead [▶ 1611]	Initializes a new instance of the SumAnyTypeRead class.

Methods

	Name	Description
	createResults [▶ 1612]	Creates the SumCommand results (abstract) (Overrides SumSymbolReadBase.createResults(IList.SumDataEntity., IList.ReadOnlyMemory.Byte., .AdsErrorCode..))
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Read2 [▶ 1613]	Reads the Values.
	Read2Async [▶ 1614]	Read as an asynchronous operation.
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Extension Methods

	Name	Description
	AllFailed [▶ 1592]	Gets a value indicating, that all SubCommands failed. (Defined by ISumCommandExtension [▶ 1591].)
	AllSucceeded [▶ 1593]	Gets a value indicating, that all SubCommands succeeded. (Defined by ISumCommandExtension [▶ 1591].)
	FirstSubError [▶ 1594]	Gets the first SubError that is not NoError [▶ 664] (Defined by ISumCommandExtension [▶ 1591].)
	OneFailed [▶ 1594]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Defined by ISumCommandExtension [▶ 1591].)

	Name	Description
	OneSucceeded [▶ 1595]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Defined by ISumCommandExtension [▶ 1591].)
	OverallError [▶ 1595]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Defined by ISumCommandExtension [▶ 1591].)
	PollValues2.ISymbol .(IObservable.Unit.) [▶ 1370]	Overloaded. Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger in one roundtrip. (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues2.ISymbol .(TimeSpan) [▶ 1372]	Overloaded. Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger. (Defined by ValueSymbolExtensions [▶ 1333].)

Remarks

The `SumAnyTypeRead` implements symbolic read with predefined value marshalling. The requested value types are specified before calling the read operation and must fit to its symbol definition. The advantage of the symbolic access is (in contrast to the handle access classes [SumHandleRead](#) [▶ 1557], [SumHandleWrite](#) [▶ 1561]) or the access by instance path ([SumInstancePathAnyTypeRead](#) [▶ 1615]) that all type information is available when using this ADS Sum Command. The disadvantage is, that the Symbolic information must be loaded before using this SumCommand, see examples. The `SumAnyTypeRead` defaults to an `InstancePath/SymbolName` read ([ValueByName](#) [▶ 1231], [ValueByName](#) [▶ 1608]).

Examples

Usage of the SumAnyTypeRead SumCommand

```
AmsAddress address = new AmsAddress(AmsNetId.Local, AmsPort.PlcRuntime_851);

CancellationTokenSource cancelSource = new CancellationTokSource();
CancellationToken cancel = cancelSource.Token;

using (AdsClient client = new AdsClient())
{
    // Connect the AdsClient to the device target.
    client.Connect(address);

    // Load symbolic information
    ISymbolLoader loader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    var resultReadSymbols = await loader.GetSymbolsAsync(cancel);
    resultReadSymbols.ThrowOnError();

    ISymbolCollection<ISymbol> allSymbols = resultReadSymbols.Symbols;

    ISymbol var1Sym = allSymbols["GVL.bVar1"];
    ISymbol var2Sym = allSymbols["GVL.iCount"];
    ISymbol projectNameSym = allSymbols["TwinCAT_SystemInfoVarList._AppInfo.ProjectName"];

    // Define the ANY_TYPE specifiers
    List<(ISymbol, Type, int[] args)> instancePathList = new List<(ISymbol, Type, int[] args)>() {
        (var1Sym, typeof(bool), null),
        (var2Sym, typeof(short), null),
        (projectNameSym, typeof(string), new int[] {63}) // See ANY_TYPE specifications
    };
    SumAnyTypeRead sumRead = new SumAnyTypeRead(client, instancePathList);

    var resultRead = await sumRead.Read2Async(cancel);

    if (resultRead.Succeeded)
    {
        foreach (var subResult in resultRead.ValueResults)
        {
            Console.WriteLine($"Symbol: {subResult.Source}, Value: {subResult.Value}, ErrorCode: {(AdsError.ErrorCode) subResult.ErrorCode}");
        }
    }
}
```

```
var var1Value = (bool)resultRead.ValueResults[0].Value;
var var2Value = (short)resultRead.ValueResults[1].Value;
var projectName = (string)resultRead.ValueResults[2].Value;
}
```

Reference

[TwinCAT.Ads.SumCommand Namespace \[▶ 1525\]](#)

[TwinCAT.Ads.SumCommand.ISumCommand \[▶ 1527\]](#)

[TwinCAT.Ads.SumCommand.ISumRead2.S. \[▶ 1601\]](#)

SumRead

[TwinCAT.Ads.SumCommand.SumSymbolWrite \[▶ 1582\]](#)

[TwinCAT.Ads.SumCommand.SumInstancePathAnyTypeRead \[▶ 1615\]](#)

6.5.19.1 SumAnyTypeRead Constructor

Initializes a new instance of the [SumAnyTypeRead \[▶ 1608\]](#) class.

Namespace: [TwinCAT.Ads.SumCommand \[▶ 1525\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SumAnyTypeRead(
    IAdsConnection connection,
    IList<ISymbol symbol, Type type, int[] > symbolSpec
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The ADS Connection.
symbolSpec	Type: System.Collections.Generic.IList.ValueTuple.ISymbol [▶ 2691] , Type , .Int32... The symbols specifier (ANY_TYPE)

Reference

[SumAnyTypeRead Class \[▶ 1608\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▶ 1525\]](#)

6.5.19.2 SumAnyTypeRead Methods

The [SumAnyTypeRead \[▶ 1608\]](#) type exposes the following members.

Methods

	Name	Description
	createResults [▶ 1612]	Creates the SumCommand results (abstract) (Overrides SumSymbolReadBase.createResults(IList.SumDataEntity., IList.ReadOnlyMemory.Byte..., AdsErrorCode.)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)

	Name	Description
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Read2 [▶ 1613]	Reads the Values.
	Read2Async [▶ 1614]	Read as an asynchronous operation.
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Extension Methods

	Name	Description
	AllFailed [▶ 1592]	Gets a value indicating, that all SubCommands failed. (Defined by ISumCommandExtension [▶ 1591].)
	AllSucceeded [▶ 1593]	Gets a value indicating, that all SubCommands succeeded. (Defined by ISumCommandExtension [▶ 1591].)
	FirstSubError [▶ 1594]	Gets the first SubError that is not NoError [▶ 664] (Defined by ISumCommandExtension [▶ 1591].)
	OneFailed [▶ 1594]	Gets a value indicating, whether the overall Sumcommand failed (checking all subcommand results). (Defined by ISumCommandExtension [▶ 1591].)
	OneSucceeded [▶ 1595]	Gets a value indicating, whether the overall SumCommand succeeded (including all subcommands) (Defined by ISumCommandExtension [▶ 1591].)
	OverallError [▶ 1595]	Gets the Overall (combined error) from SumCommand AND SubCommands. (Defined by ISumCommandExtension [▶ 1591].)
	PollValues2.ISymbol.(IObservable.Unit.) [▶ 1370]	Overloaded. Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger in one roundtrip. (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues2.ISymbol.(TimeSpan) [▶ 1372]	Overloaded. Polls a series of symbols via a ISumRead2.S. [▶ 1601] command. The SumCommand will read all contained values with every trigger. (Defined by ValueSymbolExtensions [▶ 1333].)

Reference

[SumAnyTypeRead Class](#) [[▶ 1608](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.19.2.1 SumAnyTypeRead.createResults Method

Creates the SumCommand results (abstract)

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected override ResultValue2<ISymbol, Object>[] createResults(
    IList<SumDataEntity> infoList,
    IList<ReadOnlyMemory<byte>> readBlocks,
    AdsErrorCode[] subErrors
)
```

Parameters

infoList	Type: System.Collections.Generic.IList.SumDataEntity . The information list.
readBlocks	Type: System.Collections.Generic.IList.ReadOnlyMemory.Byte.. The read blocks.
subErrors	Type: .TwinCAT.Ads.AdsErrorCode [▶ 664]. The sub errors.

Return Value

Type: [.ResultValue2](#) [▶ 1234].[ISymbol](#) [▶ 2691], [Object..](#)
[ResultReadValueAccess2](#)[].

Reference

[SumAnyTypeRead Class](#) [▶ 1608]

[TwinCAT.Ads.SumCommand Namespace](#) [▶ 1525]

6.5.19.2.2 SumAnyTypeRead.Read2 Method

Reads the Values.

Namespace: [TwinCAT.Ads.SumCommand](#) [▶ 1525]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ResultSumValues2<ISymbol> Read2()
```

Return Value

Type: [ResultSumValues2](#) [▶ 1605].[ISymbol](#) [▶ 2691].
[System.Object](#)[].

Implements

[ISumRead2.S..Read2.](#) [▶ 1604]

Remarks

The return values are automatically marshaled to their appropriate .NET types.

Reference

[SumAnyTypeRead Class](#) [▶ 1608]

[TwinCAT.Ads.SumCommand Namespace](#) [▶ 1525]

[!:TryRead(out object[], out AdsErrorCode[])]

ISumRead2.ReadAsync(CancellationTokentoken)

6.5.19.2.3 SumAnyTypeRead.Read2Async Method

Read as an asynchronous operation.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultSumValues2<ISymbol>> Read2Async (
    CancellationTokentoken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationTokentoken The cancellation token.
--------	--

Return Value

Type: [Task.ResultSumValues2](#) [[▶ 1605](#)].[ISymbol](#) [[▶ 2691](#)].

A Task<ResultSumValues2> representing the asynchronous operation.

Implements

[ISumRead2.S..Read2Async\(CancellationTokentoken\)](#) [[▶ 1605](#)]

Reference

[SumAnyTypeRead Class](#) [[▶ 1608](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

ISumRead2.Read.

[!:TryRead(out object[], out AdsErrorCode[])]

6.5.20 SumCommandErrorStrategy Enumeration

The Error strategy for [ISumCommand](#) [[▶ 1527](#)]

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum SumCommandErrorStrategy
```

Members

	Member name	Value	Description
	None	0	Errors are ignored

	Member name	Value	Description
	Relaxed	1	Whole SumCommand fails if the Main Request fails or all subCommands indicate errors.
	Strict	2	The SumCommand succeeds only if all Sub-Requests succeeded.

Reference

[TwinCAT.Ads.SumCommand Namespace \[▶ 1525\]](#)

6.5.21 SumCommandMode Enumeration

Indicates, which (extended) SumCommand to use.

Namespace: [TwinCAT.Ads.SumCommand \[▶ 1525\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum SumCommandMode
```

Members

	Member name	Value	Description
	Read	0	Sum Read command
	Write	1	Sum Write command
	ReadWrite	2	Sum ReadWrite command

Reference

[TwinCAT.Ads.SumCommand Namespace \[▶ 1525\]](#)

6.5.22 SumInstancePathAnyTypeRead Class

SumRead Command that uses the instancePath as symbol ID / Address and returns objects as Values (ANY Type marshaling)

Inheritance Hierarchy

[System.Object](#)

SumCommandWrapper.SumReadWrite.

SumInstancePathCommand.SumReadWrite, [String](#).

TwinCAT.Ads.SumCommand.SumInstancePathAnyTypeRead

Namespace: [TwinCAT.Ads.SumCommand \[▶ 1525\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public class SumInstancePathAnyTypeRead : SumInstancePathCommand<SumReadWrite, string>
```

The SumInstancePathAnyTypeRead type exposes the following members.

Constructors

	Name	Description
	SumInstancePathAnyTypeRead [▶ 1617]	Initializes a new instance of the SumRead class.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Read [▶ 1618]	Reads all the values in one ADS roundtrip and returns an result object containing all the SumCommand SubResults.
	ReadAsync [▶ 1618]	Reads all the values as an asynchronous operation in one ADS roundtrip and returns an result object containing all the SumCommand SubResults.
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Remarks

The return value results will be produced by the AnyTypeMarshaler and must be predefined following the ANY_TYPE concept. Therefore a list of specifiers must be specified for the SumInstancePathAnyTypeRead constructor. The internal single value reads are done by [ValueByName](#) [[▶ 1231](#)] / [ValueByName](#) [[▶ 1608](#)].

Examples

Usage of the SumInstancePathAnyTypeRead SumCommand

```
AmsAddress address = new AmsAddress(AmsNetId.Local, AmsPort.PlcRuntime_851);

CancellationTokenSource cancelSource = new CancellationTokSource();
CancellationToken cancel = cancelSource.Token;

using (AdsClient client = new AdsClient())
{
    // Connect the AdsClient to the device target.
    client.Connect(address);

    // Load symbolic information
    ISymbolLoader loader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);
    var resultReadSymbols = await loader.GetSymbolsAsync(cancel);
    resultReadSymbols.ThrowOnError();

    ISymbolCollection<ISymbol> allSymbols = resultReadSymbols.Symbols;

    ISymbol var1Sym = allSymbols["GVL.bVar1"];
    ISymbol var2Sym = allSymbols["GVL.iCount"];
    ISymbol projectNameSym = allSymbols["TwinCAT_SystemInfoVarList._AppInfo.ProjectName"];

    // Define the ANY_TYPE specifiers
    List<(ISymbol,Type,int[] args)> instancePathList = new List<(ISymbol, Type, int[] args)>() {
        (var1Sym, typeof(bool), null),
        (var2Sym, typeof(short), null),
        (projectNameSym,typeof(string),new int[] {63}) // See ANY_TYPE specifications
    };
    SumAnyTypeRead sumRead = new SumAnyTypeRead(client, instancePathList);

    var resultRead = await sumRead.Read2Async(cancel);

    if (resultRead.Succeeded)
```



```

{
    foreach (var subResult in resultRead.ValueResults)
    {
        Console.WriteLine($"Symbol: {subResult.Source}, Value: {subResult.Value}, ErrorCode: {(AdsEr
rorCode) subResult.ErrorCode}");
    }

    var var1Value = (bool)resultRead.ValueResults[0].Value;
    var var2Value = (short)resultRead.ValueResults[1].Value;
    var projectName = (string)resultRead.ValueResults[2].Value;
}

```

Reference

[TwinCAT.Ads.SumCommand Namespace \[▶ 1525\]](#)

6.5.22.1 SumInstancePathAnyTypeRead Constructor

Initializes a new instance of the SumRead class.

Namespace: [TwinCAT.Ads.SumCommand \[▶ 1525\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public SumInstancePathAnyTypeRead(
    IAdsConnection connection,
    IList<(string instancePath, Type type, int[] )> typeSpecs
)

```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The ADS Connection.
typeSpecs	Type: System.Collections.Generic.IList.ValueTuple.String, Type, .Int32... List of InstancePath / ANY_TYPE specifiers to specify the types of the return values.

Reference

[SumInstancePathAnyTypeRead Class \[▶ 1615\]](#)

[TwinCAT.Ads.SumCommand Namespace \[▶ 1525\]](#)

6.5.22.2 SumInstancePathAnyTypeRead Methods

The [SumInstancePathAnyTypeRead \[▶ 1615\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)

	Name	Description
	Read [▶ 1618]	Reads all the values in one ADS roundtrip and returns an result object containing all the SumCommand SubResults.
	ReadAsync [▶ 1618]	Reads all the values as an asynchronous operation in one ADS roundtrip and returns an result object containing all the SumCommand SubResults.
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[SumInstancePathAnyTypeRead Class](#) [[▶ 1615](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

6.5.22.2.1 SumInstancePathAnyTypeRead.Read Method

Reads all the values in one ADS roundtrip and returns an result object containing all the SumCommand SubResults.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ResultSumTypedValues<string, Object> Read()
```

Return Value

Type: [ResultSumTypedValues](#).[String](#), [Object](#).
[System.Object](#)[].

Remarks

The return values are automatically marshalled to their appropriate .NET types.

Reference

[SumInstancePathAnyTypeRead Class](#) [[▶ 1615](#)]

[TwinCAT.Ads.SumCommand Namespace](#) [[▶ 1525](#)]

[I!.TryRead\(out object\[\], out AdsErrorCode\[\]\)](#)

[ISumRead2.ReadAsync\(CancellationTokentoken\)](#)

6.5.22.2.2 SumInstancePathAnyTypeRead.ReadAsync Method

Reads all the values as an asynchronous operation in one ADS roundtrip and returns an result object containing all the SumCommand SubResults.

Namespace: [TwinCAT.Ads.SumCommand](#) [[▶ 1525](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultSumTypedValues<string, Object>> ReadAsync(
    CancellationTokentoken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultSumTypedValues.String, Object..](#)
A [Task<ResultSumValues2>](#) representing the asynchronous operation.

Reference

[SumInstancePathAnyTypeRead Class \[▶ 1615\]](#)










[TwinCAT.Ads.SumCommand Namespace \[▶ 1525\]](#)


[ISumRead2.Read.](#)

[\[!:TryRead\(out object\[\], out AdsErrorCode\[\]\)\]](#)


6.6 TwinCAT.Ads.TcpRouter Namespace

Classes


	Class	Description
	AmsTcpIpRouter [▶ 1620]	ADS Router class
	Route [▶ 1654]	Class Route specifies a an AMS/ADS Route object.
	RouteCollection [▶ 1664]	Collection of routes.
	RouterException [▶ 1678]	An RouterException [▶ 1678] is thrown on communication errors in the AmsTcpIpRouter [▶ 1620] class.
	RouterNotInitializedException [▶ 1682]	Class RouterNotInitializedException. Implements the RouterException [▶ 1678]
	RouterNotStartedException [▶ 1684]	Class RouterNotStartedException. Implements the RouterException [▶ 1678]
	RouterStatusChangedEventArgs [▶ 1688]	Event Arguments sent when the RouterStatusChanged [▶ 1654]. Implements the EventArgs
	StaticRoutesXmlConfigurationBuilderExtension [▶ 1690]	Extension class adding StaticRoutes.xml file reading to the IConfigurationBuilder .
	StaticRoutesXmlConfigurationProvider [▶ 1691]	Class StaticRoutesXmlConfigurationProvider. Implements the ConfigurationProvider

	Class	Description
	StaticRoutesXmlConfigurationSource [▶ 1696]	StaticRoutes Configuration Sources Implements the IConfigurationSource

Interfaces

	Interface	Description
	IAmsRouter [▶ 1646]	Interface IAmsRouter

Enumerations

	Enumeration	Description
	RouterStatus [▶ 1687]	Enum RouterStatus

6.6.1 AmsTcpIpRouter Class

ADS Router class

Inheritance Hierarchy

[System.Object](#)

TwinCAT.Ads.TcpRouter.AmsTcpIpRouter

Namespace: [TwinCAT.Ads.TcpRouter](#) [▶ 1619]

Assembly: [TwinCAT.Ads.TcpRouter](#) (in [TwinCAT.Ads.TcpRouter.dll](#)) Version: 6.0.328+39e3229





Syntax



C#

```
public class AmsTcpIpRouter : IAmsRouter
```













The AmsTcpIpRouter type exposes the following members.

Constructors




	Name	Description
	AmsTcpIpRouter(AmsNetId) [▶ 1627]	Initializes a new instance of the AmsTcpIpRouter class.
	AmsTcpIpRouter(IConfiguration) [▶ 1627]	Initializes a new instance of the AmsTcpIpRouter class.
	AmsTcpIpRouter(AmsNetId, ILogger) [▶ 1629]	Initializes a new instance of the AmsTcpIpRouter class.
	AmsTcpIpRouter(ILogger, IConfiguration) [▶ 1628]	Initializes a new instance of the AmsTcpIpRouter class.












	Name	Description
	AmsTcpIpRouter(AmsNetId, Int32, IPAddress, Int32, .IPAddress, ILogger) [► 1630]	Initializes a new instance of the AmsTcpIpRouter class.
	AmsTcpIpRouter(AmsNetId, Int32, IPAddress, Int32, IPNetwork, ILogger) [► 1631]	Initializes a new instance of the AmsTcpIpRouter class.

Properties


	Name	Description
	AcceptedLoopbackIps [► 1636]	Gets the allowed loopback sources.
	AcceptedLoopbackNetwork [► 1637]	Gets the Allowed loopback network.
	IsActive [► 1633]	Gets a value indicating whether the IAmsRouter [► 1646] is active (Running or in Starting / Stopping state).
	IsRunning [► 1634]	Gets a value indicating whether the IAmsRouter [► 1646] is running (Start phase completely finished).
	LocalNetId [► 1634]	Gets the local AmsNetId [► 767]
	Logger [► 1637]	Gets the logger interface
	Loopback [► 1635]	Gets the loopback alias
	LoopbackPort [► 1638]	Gets the loopback port for internal Loopback communication (default is 0xBF02)
	NetId [► 1635]	The Local AmsNetId [► 767] of this router.
	RouterStatus [► 1636]	Gets the router status.
	Routes [► 1638]	Gets a read-only list of the routes
	TcpIpPort [► 1638]	Gets the used TCP IP Port for External communication.

Methods



	Name	Description
	AddRoute [► 1640]	Adds a dynamic Route [► 1654]
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)

	Name	Description
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	IsRegisteredServer [▶ 1640]	Determines whether the Address specifies an locally registered AmsServer
	LoadRouteInfo [▶ 1644]	Loads the route information.
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	RemoveRoute(AmsNetId) [▶ 1642]	Removes a dynamic Route [▶ 1654]
	RemoveRoute(String) [▶ 1641]	Removes a dynamic Route [▶ 1654]
	StartAsync [▶ 1642]	Starts the AmsTcpIpRouter asynchronously.
	Stop [▶ 1643]	Stops the AmsTcpIpRouter.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryAddRoute [▶ 1643]	Adds a dynamic Route [▶ 1654]

Events

	Name	Description
	RouterStatusChanged [▶ 1645]	Occurs when the router status changes.

Fields

	Name	Description
	DEFAULT_TCP_PORT [▶ 1645]	The default TCP port (0xBF02, 48898)
	TCP_TIMEOUT [▶ 1646]	The TCP timeout

Remarks

The AmsTcpIpRouter implements a simple ADS/AMS Router for systems, where no TwinCAT Router installation is available.

The Router opens two different type of communication channels

- One for the machine internal communication between AdsClients and AdsServers. This is called the ****ADS Loopback**** port and has by default the IPAddress.Loopback (172.0.0.1) and Port 0xBF02 ([DEFAULT_TCP_PORT](#) [▶ 1645])
- The other that routes the machine external communication to other TwinCAT ADS routers. This ****ADS External Channel**** is opened by default on every available IPAddress connected to a Network again on default Port 0xBF02. A ****StaticRoutes**** table (e.g. the StaticRoutes.xml of every TwinCAT System) is used to ensure security and maintain the list of remote target systems.

With this infrastructure, the AmsTcpIpRouter is able to process ADS/AMS communication in form of ADS Requests and Responses not only isolated on the local system, but also between the local system and remote systems as long they are properly configured.

For sophisticated scenarios, the default configurations can be changed to also support more than one router on one system and/or split Client/Server/Router applications in virtual environments like docker.

Property Name	Description	Default
Name	The name of the local system.	No default, must be set.
LocalNetId [▶ 1634]	The AmsNetId of the local system.	No default, must be set.
RemoteConnections	List of the registered remote Routes / Systems that can be reached remotely.	None/Empty
TcpIpPort [▶ 1638]	The TCP Port to be used for external ADS communication (**ADS External channel**)	0xBF02 (48898)
Loopback [▶ 1635]	The LoopbackIP of Device internal communication. This setting should only be changed if ADS Server and Router application should run on different (virtual) devices. On the AdsServer side an appropriate system configuration via RouterEndPoint [▶ 2006] is necessary if the LoopbackIP is changed.	Loopback 127.0.0.1
LoopbackPort	The Loopback Port of Device internal ADS communication. This setting should only be changed when different TcpPorts are necessary to separate internal and external router communication (e.g. in WSL2 docker containers) where the LoopbackIP 127.0.0.1 isn't appropriate. On the AdsServer side an appropriate system configuration via RouterEndPoint [▶ 2006] is necessary if the loopback port is changed.	0xBF02 (48898)
AcceptedLoopbackIPs [▶ 1636]	The TCP Port to be used for external ADS communication (**ADS External channel**)	0xBF02 (48898)
AcceptedLoopbackNetwork [▶ 1637]	The TCP Port to be used for external ADS communication (**ADS External channel**)	0xBF02 (48898)

The router settings can be configured by the .NET Configuration Builder [IConfigurationBuilder](#) or simply by calling one of the AmsTcpIpRouter constructor overloads.

Xml Configuration (StaticRoutes.xml)	AddStaticRoutesXmlConfiguration(IConfigurationBuilder) [▶ 1691]
Json Configuration (appSettings.json)	Microsoft.Extensions.Configuration.JsonConfigurationExtensions.AddJsonFile(string)
Environment variables	Microsoft.Extensions.Configuration.EnvironmentVariablesExtensions.AddEnvironmentVariables()
ConsoleArguments	Microsoft.Extensions.Configuration.CommandLineConfigurationExtensions.AddCommandLine(string[])

Examples

The following sample shows how to use the `AmsTcpIpRouter` class within an own Console application. This console application can also be accessed as binary from the Nuget.org package repository. The ID of the package is 'Beckhoff.TwinCAT.Ads.AdsRouterConsole'.

Ads Router WorkerService

```

class Program
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    public static void Main(string[] args)
    {
        try
        {
            CreateHostBuilder(args).Build().Run();
        }
        catch (OperationCanceledException /*cex*/)
        {
            Console.WriteLine("Router cancelled!");
        }
        catch (Exception ex)
        {
            Console.WriteLine($"Router failed with '{ex.Message}");
        }
    }

    /// <summary>
    /// Creates the host builder.
    /// </summary>
    /// <param name="args">The arguments.</param>
    /// <returns>IHostBuilder.</returns>
    public static IHostBuilder CreateHostBuilder(string[] args)
    {
        var ret = Host.CreateDefaultBuilder(args);

        ret.ConfigureServices((hostContext, services) =>
        {
            services.AddHostedService<RouterService>();
        })
        .ConfigureAppConfiguration((hostingContext, config) =>
        {
            // Add further AppConfigurationProvider here.
            //config.Sources.Clear(); // Clear all default config sources
            config.AddEnvironmentVariables("ENV_"); // Use Environment variables
            //config.AddCommandLine(args); // Use Command Line
            //config.AddJsonFile("appSettings.Development.json"); // Use Appsettings
            //config.AddStaticRoutesXmlConfiguration(); // Overriding settings with StaticRoutes.Xml
        })
        .ConfigureLogging(logging =>
        {
            //logging.ClearProviders();
            // Adding console logging here.
            //logging.AddConsole();
        });
        return ret;
    }
}

/// <summary>
/// The RouterService instance represents a long running (hosted) service that implements an <see cref="AmsTcpIpRouter"/>.
/// Implements the <see cref="BackgroundService" />
/// </summary>
/// </remarks>
/// Long running Background task that runs a <see cref="AmsTcpIpRouter."/>.
/// The service is stopped via the <see cref="CancellationToken"/>
/// given to the <see cref="ExecuteAsync(CancellationTokent)"/> method.
/// </remarks>
/// <seealso cref="BackgroundService" />
public class RouterService : BackgroundService
{
    /// <summary>
    /// Logger
    /// </summary>
    private readonly ILogger<RouterService> _logger;

```



```

/// <summary>
/// Configuration
/// </summary>
private readonly IConfiguration _configuration;

/// <summary>
/// Initializes a new instance of the <see cref="RouterService"/> class.
/// </summary>
/// <param name="logger">The logger.</param>
/// <param name="configuration">The configuration.</param>
public RouterService(ILogger<RouterService> logger, IConfiguration configuration)
{
    _logger = logger;
    _configuration = configuration;
    string? value = _configuration.GetValue("ASPNETCORE_ENVIRONMENT", "Production");
}

/// <summary>
/// Execute the Router asynchronously as <see cref="BackgroundService"/>.
/// </summary>
/// <param name="cancel">The cancellation token.</param>
protected override async Task ExecuteAsync(CancellationToken cancel)
{
    AmsTcpIpRouter router;

using (_logger.BeginScope("Starting"))
{
    StringBuilder appCommon = new StringBuilder();

    appCommon.AppendLine($"ApplicationPath: {Environment.GetCommandLineArgs()[0]}");
    appCommon.AppendLine($"BaseDirectory: {AppContext.BaseDirectory}");
    appCommon.AppendLine($"CurrentDirectory: {Directory.GetCurrentDirectory()}");
    // _logger.LogInformation(sB.ToString());

    StringBuilder config = new StringBuilder();
    string value = _configuration.GetValue("ASPNETCORE_ENVIRONMENT", "Production");
    config.AppendLine($"ASPNETCORE_ENVIRONMENT: {value}");

    Console.WriteLine("Application Directories");
    Console.WriteLine("=====");
    Console.WriteLine(appCommon);
    Console.WriteLine("");
    Console.WriteLine("Configuration");
    Console.WriteLine("=====");
    Console.WriteLine(config);
    Console.WriteLine("");

    Console.WriteLine("Press Ctrl + C to shutdown!");

    router = new AmsTcpIpRouter(_logger, _configuration);
    router.RouterStatusChanged += Router_RouterStatusChanged;

    // Use this overload to instantiate a Router without support of IHost/
    IConfigurationProvider support and parametrize by code
    // AmsTcpIpRouter router = new AmsTcpIpRouter(new AmsNetId("1.2.3.4.5.6"), AmsTcpIpRouter.DE
    FAULT_TCP_PORT, IPAddress.Loopback, AmsTcpIpRouter.DEFAULT_TCP_PORT, _logger);
    // router.AddRoute(...);

    _logger.LogInformation(appCommon.ToString());
    _logger.LogInformation(config.ToString());
}

Task routerTask = router.StartAsync(cancel); // Start the router

//#if ADSSERVER
// AdsServer could be started here:
SystemServiceServer systemService = new SystemServiceServer(router, _logger);
AdsRouterServer adsRouterService = new AdsRouterServer(router, _logger);

//TestAdsServer testServer = new TestAdsServer(_logger);
// _logger.LogInformation($"Adding Test Server on port '{systemService.ServerAddress}'");
Task systemServiceTask = systemService.ConnectServerAndWaitAsync(cancel);
Task routerServerTask = adsRouterService.ConnectServerAndWaitAsync(cancel);

await Task.WhenAll(routerTask, systemServiceTask, routerServerTask);

//await routerTask;
//#endif
}

```

```

/// <summary>
/// Handles the RouterStatusChanged event of the <see cref="AmsTcpIpRouter"/>
/// </summary>
/// <param name="sender">The source of the event.</param>
/// <param name="e">The <see cref="RouterStatusChangedEventArgs"/>
> instance containing the event data.</param>
private void Router_RouterStatusChanged(object? sender, RouterStatusChangedEventArgs e)
{
    if (e.RouterStatus == RouterStatus.Started)
    {
        // From here on, the Router is available to receive Data.
    }
}

```

Reference

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

[RouteConfig](#)

[TwinCAT.Ams.AmsConfiguration](#) [► 2004]

Also see about this

- 📖 [AmsTcpIpRouter Constructor \(IConfigurationRoot\)](#) [► 1631]
- 📖 [AmsTcpIpRouter Constructor \(ILogger, IConfigurationRoot\)](#) [► 1632]

6.6.1.1 AmsTcpIpRouter Constructor

Overload List

	Name	Description
📖	AmsTcpIpRouter(AmsNetId) [► 1627]	Initializes a new instance of the AmsTcpIpRouter [► 1620] class.
📖	AmsTcpIpRouter(IConfiguration) [► 1627]	Initializes a new instance of the AmsTcpIpRouter [► 1620] class.
📖	AmsTcpIpRouter(AmsNetId, ILogger) [► 1629]	Initializes a new instance of the AmsTcpIpRouter [► 1620] class.
📖	AmsTcpIpRouter(ILogger, IConfiguration) [► 1628]	Initializes a new instance of the AmsTcpIpRouter [► 1620] class.
📖	AmsTcpIpRouter(AmsNetId, Int32, IPAddress, Int32, IPAddress, ILogger) [► 1630]	Initializes a new instance of the AmsTcpIpRouter [► 1620] class.
📖	AmsTcpIpRouter(AmsNetId, Int32,	Initializes a new instance of the AmsTcpIpRouter [► 1620] class.

	Name	Description
	IPAddress, Int32, IPNetwork, ILogger [▶ 1631]	

Reference

[AmsTcpIpRouter Class \[▶ 1620\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

Also see about this

- [AmsTcpIpRouter Constructor \(IConfigurationRoot\) \[▶ 1631\]](#)
- [AmsTcpIpRouter Constructor \(ILogger, IConfigurationRoot\) \[▶ 1632\]](#)

6.6.1.1.1 AmsTcpIpRouter Constructor (IConfiguration)

Initializes a new instance of the [AmsTcpIpRouter \[▶ 1620\]](#) class.

Namespace: [TwinCAT.Ads.TcpRouter \[▶ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsTcpIpRouter(
    IConfiguration configuration
)
```

Parameters

configuration	Type: Microsoft.Extensions.Configuration.IConfiguration
---------------	---

Remarks

The [AmsTcpIpRouter \[▶ 1620\]](#) will be connected to the Default port [DEFAULT_TCP_PORT \[▶ 1645\]](#).

Reference

[AmsTcpIpRouter Class \[▶ 1620\]](#)

[AmsTcpIpRouter Overload \[▶ 1626\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.1.1.2 AmsTcpIpRouter Constructor (AmsNetId)

Initializes a new instance of the [AmsTcpIpRouter \[▶ 1620\]](#) class.

Namespace: [TwinCAT.Ads.TcpRouter \[▶ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsTcpIpRouter(
    AmsNetId local
)
```

Parameters

local	Type: TwinCAT.Ads.AmsNetId [▸ 767] The local.
-------	--

Remarks

The [AmsTcpIpRouter \[▸ 1620\]](#) will be connected to the Default port [DEFAULT_TCP_PORT \[▸ 1645\]](#).

Reference

[AmsTcpIpRouter Class \[▸ 1620\]](#)

[AmsTcpIpRouter Overload \[▸ 1626\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▸ 1619\]](#)

6.6.1.1.3 AmsTcpIpRouter Constructor (ILogger, IConfiguration)

Initializes a new instance of the [AmsTcpIpRouter \[▸ 1620\]](#) class.

Namespace: [TwinCAT.Ads.TcpRouter \[▸ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public AmsTcpIpRouter(
    ILogger? logger,
    IConfiguration configuration
)
```

Parameters

logger	Type: Microsoft.Extensions.Logging.ILogger The logger.
configuration	Type: Microsoft.Extensions.Configuration.IConfiguration The configuration.

Exceptions

Exception	Condition
Exception	No IPv4 address for server

Remarks

The [AmsTcpIpRouter \[▸ 1620\]](#) will be connected to the Default port [DEFAULT_TCP_PORT \[▸ 1645\]](#).

Reference

[AmsTcpIpRouter Class \[▸ 1620\]](#)

[AmsTcpIpRouter Overload \[▸ 1626\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▸ 1619\]](#)

6.6.1.1.4 AmsTcpIpRouter Constructor (AmsNetId, Int32, IPAddress, Int32, ILogger)

Initializes a new instance of the [AmsTcpIpRouter \[▸ 1620\]](#) class.

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public AmsTcpIpRouter(
    AmsNetId local,
    int externalPort,
    IPAddress loopbackIP,
    int loopbackPort,
    ILogger logger
)
```

Parameters

- local Type: [TwinCAT.Ads.AmsNetId](#) [[▶ 767](#)]
The (enforced) local NET Id.
- externalPort Type: [System.Int32](#)
The TCP/IP port the [AmsTcplpRouter](#) [[▶ 1620](#)] is using.
- loopbackIP Type: [System.Net.IPAddress](#)
The loopback ip.
- loopbackPort Type: [System.Int32](#)
The loopback port.
- logger Type: [ILogger](#)
The logger.

Remarks

This constructor doesn't read the 'StaticRoutes.xml'. Instead, it is expected to force the local [AmsNetId](#) [[▶ 767](#)], the TCP/IP port and add the routes as Dynamic routes.

Reference

- [AmsTcplpRouter Class](#) [[▶ 1620](#)]
- [AmsTcplpRouter Overload](#) [[▶ 1626](#)]
- [TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.1.1.5 AmsTcplpRouter Constructor (AmsNetId, ILogger)

Initializes a new instance of the [AmsTcplpRouter](#) [[▶ 1620](#)] class.

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsTcpIpRouter(
    AmsNetId local,
    ILogger logger
)
```

Parameters

local	Type: TwinCAT.Ads.AmsNetId [▶ 767] The local.
-------	--

logger	Type: Microsoft.Extensions.Logging.ILogger The logger.
--------	---

Remarks

The [AmsTcpIpRouter](#) [▶ 1620] will be connected to the Default port [DEFAULT_TCP_PORT](#) [▶ 1645].

Reference

[AmsTcpIpRouter Class](#) [▶ 1620]

[AmsTcpIpRouter Overload](#) [▶ 1626]

[TwinCAT.Ads.TcpRouter Namespace](#) [▶ 1619]

6.6.1.1.6 AmsTcpIpRouter Constructor (AmsNetId, Int32, IPAddress, Int32, .IPAddress., ILogger)

Initializes a new instance of the [AmsTcpIpRouter](#) [▶ 1620] class.

Namespace: [TwinCAT.Ads.TcpRouter](#) [▶ 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsTcpIpRouter(
    AmsNetId local,
    int externalPort,
    IPAddress? loopbackIP,
    int loopbackPort,
    IPAddress[]? loopbackCommunicationIPs,
    ILogger? logger
)
```

Parameters

local	Type: TwinCAT.Ads.AmsNetId [▶ 767] The (enforced) local NET Id.
externalPort	Type: System.Int32 The TCP/IP port the AmsTcpIpRouter [▶ 1620] is using.
loopbackIP	Type: System.Net.IPAddress The loopback ip.
loopbackPort	Type: System.Int32 The loopback port.
loopbackCommunicationIPs	Type: .System.Net.IPAddress . The loopback communication sources.
logger	Type: Microsoft.Extensions.Logging.ILogger The logger.

Remarks

This constructor doesn't read the 'StaticRoutes.xml'. Instead, it is expected to force the local [AmsNetId](#) [▶ 767], the TCP/IP port and add the routes as Dynamic routes.

Reference

[AmsTcpIpRouter Class](#) [▶ 1620]

[AmsTcpIpRouter Overload \[► 1626\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[► 1619\]](#)

6.6.1.1.7 AmsTcpIpRouter Constructor (AmsNetId, Int32, IPAddress, Int32, IPNetwork, ILogger)

Initializes a new instance of the [AmsTcpIpRouter \[► 1620\]](#) class.

Namespace: [TwinCAT.Ads.TcpRouter \[► 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsTcpIpRouter(
    AmsNetId local,
    int externalPort,
    IPAddress? loopbackIP,
    int loopbackPort,
    IPNetwork? loopbackCommunicationNetwork,
    ILogger? logger
)
```

Parameters

local	Type: TwinCAT.Ads.AmsNetId [► 767] The (enforced) local NET Id.
externalPort	Type: System.Int32 The TCP/IP port the AmsTcpIpRouter [► 1620] is using.
loopbackIP	Type: System.Net.IPAddress The loopback ip.
loopbackPort	Type: System.Int32 The loopback port.
loopbackCommunicationNetwork	Type: IPNetwork The loopback communication sources.
logger	Type: Microsoft.Extensions.Logging.ILogger The logger.

Remarks

This constructor doesn't read the 'StaticRoutes.xml'. Instead, it is expected to force the local [AmsNetId \[► 767\]](#), the TCP/IP port and add the routes as Dynamic routes.

Reference

[AmsTcpIpRouter Class \[► 1620\]](#)

[AmsTcpIpRouter Overload \[► 1626\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[► 1619\]](#)

6.6.1.1.8 AmsTcpIpRouter Constructor (IConfigurationRoot)

Initializes a new instance of the [AmsTcpIpRouter \[► 1620\]](#) class.

Namespace: [TwinCAT.Ads.TcpRouter \[► 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public AmsTcpIpRouter(
    IConfigurationRoot configuration
)
```

Parameters

configuration Type: [Microsoft.Extensions.Configuration.IConfigurationRoot](#)

Remarks

The [AmsTcpIpRouter](#) [▸ 1620] will be connected to the Default port [DEFAULT_TCP_PORT](#) [▸ 1645].

Reference

[AmsTcpIpRouter Class](#) [▸ 1620]

[AmsTcpIpRouter Overload](#) [▸ 1626]

[TwinCAT.Ads.TcpRouter Namespace](#) [▸ 1619]

6.6.1.1.9 AmsTcpIpRouter Constructor (ILogger, IConfigurationRoot)

Initializes a new instance of the [AmsTcpIpRouter](#) [▸ 1620] class.

Namespace: [TwinCAT.Ads.TcpRouter](#) [▸ 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public AmsTcpIpRouter(
    ILogger? logger,
    IConfigurationRoot configuration
)
```

Parameters

logger Type: [Microsoft.Extensions.Logging.ILogger](#)
The logger.

configuration Type: [Microsoft.Extensions.Configuration.IConfigurationRoot](#)
The configuration.

Exceptions

Exception	Condition
Exception	No IPv4 address for server

Remarks

The [AmsTcpIpRouter](#) [▸ 1620] will be connected to the Default port [DEFAULT_TCP_PORT](#) [▸ 1645].

Reference

[AmsTcpIpRouter Class](#) [▸ 1620]













[AmsTcpIpRouter Overload](#) [▸ 1626]

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.1.2 AmsTcpIpRouter Properties

The [AmsTcpIpRouter \[▶ 1620\]](#) type exposes the following members.

Properties

	Name	Description
	AcceptedLoopbackPs [▶ 1636]	Gets the allowed loopback sources.
	AcceptedLoopbackNetwork [▶ 1637]	Gets the Allowed loopback network.
	IsActive [▶ 1633]	Gets a value indicating whether the IAmsRouter [▶ 1646] is active (Running or in Starting / Stopping state).
	IsRunning [▶ 1634]	Gets a value indicating whether the IAmsRouter [▶ 1646] is running (Start phase completely finished).
	LocalNetId [▶ 1634]	Gets the local AmsNetId [▶ 767]
	Logger [▶ 1637]	Gets the logger interface
	Loopback [▶ 1635]	Gets the loopback alias
	LoopbackPort [▶ 1638]	Gets the loopback port for internal Loopback communication (default is 0xBF02)
	NetId [▶ 1635]	The Local AmsNetId [▶ 767] of this router.
	RouterStatus [▶ 1636]	Gets the router status.
	Routes [▶ 1638]	Gets a read-only list of the routes
	TcpIpPort [▶ 1638]	Gets the used TCP IP Port for External communication.

Reference

[AmsTcpIpRouter Class \[▶ 1620\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.1.2.1 AmsTcpIpRouter.IsActive Property

Gets a value indicating whether the [IAmsRouter \[▶ 1646\]](#) is active (Running or in Starting / Stopping state).

Namespace: [TwinCAT.Ads.TcpRouter \[▶ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsActive { get; }
```

Property Value

Type: [Boolean](#)

true if this the router is active; otherwise, false.

Implements

[IAmsRouter.IsActive](#) [[▶ 1648](#)]

Reference

[AmsTcpIpRouter Class](#) [[▶ 1620](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.1.2.2 AmsTcpIpRouter.IsRunning Property

Gets a value indicating whether the [IAmsRouter](#) [[▶ 1646](#)] is running (Start phase completely finished).

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsRunning { get; }
```

Property Value

Type: [Boolean](#)

true if this the router is running; otherwise, false.

Implements

[IAmsRouter.IsRunning](#) [[▶ 1648](#)]

Reference

[AmsTcpIpRouter Class](#) [[▶ 1620](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.1.2.3 AmsTcpIpRouter.LocalNetId Property

Gets the local [AmsNetId](#) [[▶ 767](#)]

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AmsNetId? LocalNetId { get; }
```

Property Value

Type: [AmsNetId](#) [[▶ 767](#)]

The local net identifier.

Remarks

This is initialized during the execution of the [AmsTcpIpRouter \[▸ 1620\]](#) constructor.

Reference

[AmsTcpIpRouter Class \[▸ 1620\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▸ 1619\]](#)

6.6.1.2.4 AmsTcpIpRouter.Loopback Property

Gets the loopback alias

Namespace: [TwinCAT.Ads.TcpRouter \[▸ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IPAddress Loopback { get; }
```

Property Value

Type: [IPAddress](#)

The loopback.

Reference

[AmsTcpIpRouter Class \[▸ 1620\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▸ 1619\]](#)

6.6.1.2.5 AmsTcpIpRouter.NetId Property

The Local [AmsNetId \[▸ 767\]](#) of this router.

Namespace: [TwinCAT.Ads.TcpRouter \[▸ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsNetId? NetId { get; }
```

Property Value

Type: [AmsNetId \[▸ 767\]](#)

The net identifier.

Implements

[IAmsRouter.NetId \[▸ 1648\]](#)

Reference

[AmsTcpIpRouter Class \[▸ 1620\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▸ 1619\]](#)

6.6.1.2.6 AmsTcpIpRouter.RouterStatus Property

Gets the router status.

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RouterStatus RouterStatus { get; }
```

Property Value

Type: [RouterStatus](#) [[▶ 1687](#)]

The router status.

Implements

[IAmsRouter.RouterStatus](#) [[▶ 1649](#)]

Reference

[AmsTcpIpRouter Class](#) [[▶ 1620](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

[AmsTcpIpRouter.RouterStatusChanged](#) [[▶ 1645](#)]

6.6.1.2.7 AmsTcpIpRouter.AcceptedLoopbackIPs Property

Gets the allowed loopback sources.

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IPAddress[]? AcceptedLoopbackIPs { get; }
```

Property Value

Type: [.IPAddress](#).

The allowed loopback sources.

Remarks

These are the Loopback sources that can use the configured Loopback for communication

Remarks

This array specifies the range of valid (external) IPAddresses that are allowed to use the now 'virtual' internal Loopback TCP socket. If not set, the [AmsTcpIpRouter](#) [[▶ 1620](#)] allows on the internal Loopback socket only connections from the same machine. These connections are used for the 'internal' bidirectional communication between Router and AdsClient/AdsServer. To enable sophisticated scenarios (e.g. virtualization of AdsClient or AdsServer applications running in a decoupled environment/container independant of the Router) the accepted Network can be set. The specified network then forms a virtual ADS Device using the Device internal loopback port. This property can be used as an alternative to AcceptedLoopbackIPs.

Reference

[AmsTcpIpRouter Class](#) [► 1620]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

[AmsTcpIpRouter.AcceptedLoopbackNetwork](#) [► 1637]

6.6.1.2.8 AmsTcpIpRouter.AcceptedLoopbackNetwork Property

Gets the Allowed loopback network.

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IPNetwork? AcceptedLoopbackNetwork { get; }
```

Property Value

Type: [IPNetwork](#)

The allowed loopback network in CIDR notation

Remarks

This subnet specifies the range of valid (external) IPAddresses that are allowed to use the now 'virtual' internal Loopback TCP socket. If not set, the [AmsTcpIpRouter](#) [► 1620] allows on the internal Loopback socket only connections from the same machine. These connections are used for the 'internal' bidirectional communication between Router and AdsClient/AdsServer. To enable sophisticated scenarios (e.g. virtualization of AdsClient or AdsServer applications running in a decoupled environment/container independant of the Router) the accepted Network can be set. The specified network then forms a virtual ADS Device using the Device internal loopback port. This property can be used as an alternative to [AcceptedLoopbackIPs](#) [► 1636].

Reference

[AmsTcpIpRouter Class](#) [► 1620]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

[AmsTcpIpRouter.AcceptedLoopbackIPs](#) [► 1636]

6.6.1.2.9 AmsTcpIpRouter.Logger Property

Gets the logger interface

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ILogger? Logger { get; }
```

Property Value

Type: [ILogger](#)

The logger.

Reference

[AmsTcpIpRouter Class](#) [► 1620]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.1.2.10 AmsTcpIpRouter.LoopbackPort Property

Gets the loopback port for internal Loopback communication (default is 0xBF02)

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int LoopbackPort { get; }
```

Property Value

Type: [Int32](#)

The loopback port.

Reference

[AmsTcpIpRouter Class](#) [► 1620]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.1.2.11 AmsTcpIpRouter.Routes Property

Gets a read-only list of the routes

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IReadOnlyList<Route> Routes { get; }
```

Property Value

Type: [IReadOnlyList.Route](#) [► 1654].

The routes.

Implements

[IAmsRouter.Routes](#) [► 1649]

Reference

[AmsTcpIpRouter Class](#) [► 1620]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.1.2.12 AmsTcpIpRouter.TcpIpPort Property

Gets the used TCP IP Port for External communication.

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int TcpIpPort { get; }
```

Property Value

Type: [Int32](#)
 The TCP ip port.

Reference















[AmsTcpIpRouter Class](#) [[▶ 1620](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.1.3 AmsTcpIpRouter Methods

The [AmsTcpIpRouter](#) [[▶ 1620](#)] type exposes the following members.

Methods

	Name	Description
	AddRoute [▶ 1640]	Adds a dynamic Route [▶ 1654]
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	IsRegisteredServer [▶ 1640]	Determines whether the Address specifies an locally registered AmsServer
	LoadRouteInfo [▶ 1644]	Loads the route information.
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	RemoveRoute(AmsNetId) [▶ 1642]	Removes a dynamic Route [▶ 1654]
	RemoveRoute(String) [▶ 1641]	Removes a dynamic Route [▶ 1654]
	StartAsync [▶ 1642]	Starts the AmsTcpIpRouter [▶ 1620] asynchronously.
	Stop [▶ 1643]	Stops the AmsTcpIpRouter [▶ 1620].
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryAddRoute [▶ 1643]	Adds a dynamic Route [▶ 1654]

Reference[AmsTcpIpRouter Class](#) [► 1620][TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]**6.6.1.3.1 AmsTcpIpRouter.AddRoute Method**Adds a dynamic [Route](#) [► 1654]**Namespace:** [TwinCAT.Ads.TcpRouter](#) [► 1619]**Assembly:** TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public void AddRoute(
    Route route
)
```

Parameters

route	Type: TwinCAT.Ads.TcpRouter.Route [► 1654] The route.
-------	--

Implements[IAmsRouter.AddRoute\(Route\)](#) [► 1650]**Reference**[AmsTcpIpRouter Class](#) [► 1620][TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]**6.6.1.3.2 AmsTcpIpRouter.IsRegisteredServer Method**

Determines whether the Address specifies an locally registered AmsServer

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]**Assembly:** TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public bool IsRegisteredServer(
    AmsAddress serverAddress
)
```

Parameters

serverAddress	Type: TwinCAT.Ads.AmsAddress [► 752] The server address.
---------------	---

Return ValueType: [Boolean](#)

true if [is server registered] [the specified server address]; otherwise, false.



Reference

[AmsTcpIpRouter Class \[▸ 1620\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▸ 1619\]](#)

6.6.1.3.3 AmsTcpIpRouter.RemoveRoute Method

Overload List

	Name	Description
	RemoveRoute(AmsNetId) [▸ 1642]	Removes a dynamic Route [▸ 1654]
	RemoveRoute(String) [▸ 1641]	Removes a dynamic Route [▸ 1654]

Reference

[AmsTcpIpRouter Class \[▸ 1620\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▸ 1619\]](#)

6.6.1.3.3.1 AmsTcpIpRouter.RemoveRoute Method (String)

Removes a dynamic [Route \[▸ 1654\]](#)

Namespace: [TwinCAT.Ads.TcpRouter \[▸ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool RemoveRoute(
    string route
)
```

Parameters

route	Type: System.String The route.
-------	---

Return Value

Type: [Boolean](#)
true if removed, false otherwise.

Implements

[IAmsRouter.RemoveRoute\(String\) \[▸ 1651\]](#)

Reference

[AmsTcpIpRouter Class \[▸ 1620\]](#)

[RemoveRoute Overload \[▸ 1641\]](#)

[TwinCAT.Ads.TcpRouter Namespace](#) [▶ 1619]

6.6.1.3.3.2 AmsTcpIpRouter.RemoveRoute Method (AmsNetId)

Removes a dynamic [Route](#) [▶ 1654]

Namespace: [TwinCAT.Ads.TcpRouter](#) [▶ 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool RemoveRoute(  
    AmsNetId netId  
)
```

Parameters

netId	Type: TwinCAT.Ads.AmsNetId [▶ 767] The NetId of the route.
-------	---

Return Value

Type: [Boolean](#)

true if removed, false otherwise.

Implements

[IAmsRouter.RemoveRoute\(AmsNetId\)](#) [▶ 1651]

Reference

[AmsTcpIpRouter Class](#) [▶ 1620]

[RemoveRoute Overload](#) [▶ 1641]

[TwinCAT.Ads.TcpRouter Namespace](#) [▶ 1619]

6.6.1.3.4 AmsTcpIpRouter.StartAsync Method

Starts the [AmsTcpIpRouter](#) [▶ 1620] asynchronously.

Namespace: [TwinCAT.Ads.TcpRouter](#) [▶ 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task StartAsync(  
    CancellationToken cancel  
)
```

Parameters

cancel	Type: System.Threading.CancellationToken
--------	--

Return Value

Type: [Task](#)
Task.

Implements

[IAmsRouter.StartAsync\(CancellationToken\)](#) [[▶ 1652](#)]

Remarks

The asynchronous task runs as long until [Stop](#). [[▶ 1643](#)] is called.

Reference

[AmsTcpIpRouter Class](#) [[▶ 1620](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.1.3.5 AmsTcpIpRouter.Stop Method

Stops the [AmsTcpIpRouter](#) [[▶ 1620](#)].

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Stop()
```

Implements

[IAmsRouter.Stop](#). [[▶ 1652](#)]

Reference

[AmsTcpIpRouter Class](#) [[▶ 1620](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.1.3.6 AmsTcpIpRouter.TryAddRoute Method

Adds a dynamic [Route](#) [[▶ 1654](#)].

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryAddRoute(  
    Route route  
)
```

Parameters

route	Type: TwinCAT.Ads.TcpRouter.Route [▸ 1654] The route.
-------	--

Return Value

Type: [Boolean](#)
true if added, false otherwise.

Implements

[IAmsRouter.TryAddRoute\(Route\)](#) [[▸ 1653](#)]

Reference

[AmsTcpIpRouter Class](#) [[▸ 1620](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▸ 1619](#)]

6.6.1.3.7 AmsTcpIpRouter.LoadRouteInfo Method

Loads the route information.

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▸ 1619](#)]

Assembly: [TwinCAT.Ads.TcpRouter](#) (in [TwinCAT.Ads.TcpRouter.dll](#)) Version: 6.0.328+39e3229

Syntax**C#**

```
public AmsRegisterTpPortIn[] LoadRouteInfo()
```

Return Value

Type: [.AmsRegisterTpPortIn](#).
[AmsRegisterTpPortIn\[\]](#).

Reference


[AmsTcpIpRouter Class](#) [[▸ 1620](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▸ 1619](#)]

6.6.1.4 AmsTcpIpRouter Events

The [AmsTcpIpRouter](#) [[▸ 1620](#)] type exposes the following members.

Events

	Name	Description
	RouterStatusChanged [▸ 1645]	Occurs when the router status changes.

Reference

[AmsTcpIpRouter Class](#) [[▸ 1620](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▸ 1619](#)]

6.6.1.4.1 AmsTcpIpRouter.RouterStatusChanged Event

Occurs when the router status changes.

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<RouterStatusChangedEventArgs> RouterStatusChanged
```

Value

Type: [System.EventHandler.RouterStatusChangedEventArgs](#) [[▶ 1688](#)].

Implements

[IAmsRouter.RouterStatusChanged](#) [[▶ 1654](#)]

Reference

[AmsTcpIpRouter Class](#) [[▶ 1620](#)]





[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

[AmsTcpIpRouter.RouterStatus](#) [[▶ 1636](#)]

6.6.1.5 AmsTcpIpRouter Fields

The [AmsTcpIpRouter](#) [[▶ 1620](#)] type exposes the following members.

Fields

	Name	Description
 	DEFAULT_TCP_PORT [▶ 1645]	The default TCP port (0xBF02, 48898)
 	TCP_TIMEOUT [▶ 1646]	The TCP timeout

Reference

[AmsTcpIpRouter Class](#) [[▶ 1620](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.1.5.1 AmsTcpIpRouter.DEFAULT_TCP_PORT Field

The default TCP port (0xBF02, 48898)

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public const int DEFAULT_TCP_PORT = 48898
```

Field Value

Type: [Int32](#)

Reference

[AmsTcpIpRouter Class](#) [[▶ 1620](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.1.5.2 AmsTcpIpRouter.TCP_TIMEOUT Field

The TCP timeout

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static int TCP_TIMEOUT
```

Field Value

Type: [Int32](#)

Reference

[AmsTcpIpRouter Class](#) [[▶ 1620](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.2 IAmsRouter Interface

Interface IAmsRouter

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229



Syntax




C#

```
public interface IAmsRouter
```







The IAmsRouter type exposes the following members.

Properties


	Name	Description
	IsActive [▶ 1648]	Gets a value indicating whether the IAmsRouter is active (Running or in Starting / Stopping state).
	IsRunning [▶ 1648]	Gets a value indicating whether the IAmsRouter is running (Start phase completely finished).

	Name	Description
	NetId [▶ 1648]	The Local AmsNetId [▶ 767] of the IAmsRouter
	RouterStatus [▶ 1649]	Gets the router status.
	Routes [▶ 1649]	Gets a read-only list of the routes

Methods

	Name	Description
	AddRoute [▶ 1650]	Adds a dynamic Route [▶ 1654]
	RemoveRoute(AmsNetId) [▶ 1651]	Removes a dynamic Route [▶ 1654]
	RemoveRoute(String) [▶ 1651]	Removes a dynamic Route [▶ 1654]
	StartAsync [▶ 1652]	Starts the IAmsRouter asynchronously.
	Stop [▶ 1652]	Stops the IAmsRouter .
	TryAddRoute [▶ 1653]	Adds a dynamic Route [▶ 1654]

Events

	Name	Description
	RouterStatusChanged [▶ 1654]	Occurs when the router status changes.






Reference

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.2.1 IAmsRouter Properties

The [IAmsRouter \[▶ 1646\]](#) type exposes the following members.

Properties

	Name	Description
	IsActive [▶ 1648]	Gets a value indicating whether the IAmsRouter [▶ 1646] is active (Running or in Starting / Stopping state).
	IsRunning [▶ 1648]	Gets a value indicating whether the IAmsRouter [▶ 1646] is running (Start phase completely finished).
	NetId [▶ 1648]	The Local AmsNetId [▶ 767] of the IAmsRouter [▶ 1646]
	RouterStatus [▶ 1649]	Gets the router status.
	Routes [▶ 1649]	Gets a read-only list of the routes

Reference

[IAmsRouter Interface \[▶ 1646\]](#)

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.2.1.1 IAmRouter.IsActive Property

Gets a value indicating whether the [IAmRouter](#) [► 1646] is active (Running or in Starting / Stopping state).

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsActive { get; }
```

Property Value

Type: [Boolean](#)

true if this the router is active; otherwise, false.

Reference

[IAmRouter Interface](#) [► 1646]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.2.1.2 IAmRouter.IsRunning Property

Gets a value indicating whether the [IAmRouter](#) [► 1646] is running (Start phase completely finished).

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsRunning { get; }
```

Property Value

Type: [Boolean](#)

true if this the router is running; otherwise, false.

Reference

[IAmRouter Interface](#) [► 1646]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.2.1.3 IAmRouter.NetId Property

The Local [AmsNetId](#) [► 767] of the [IAmRouter](#) [► 1646]

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AmsNetId? NetId { get; }
```


Property Value

Type: [AmsNetId](#) [[▶ 767](#)]

The net identifier.

Reference

[IAmsRouter Interface](#) [[▶ 1646](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.2.1.4 IAmsRouter.RouterStatus Property

Gets the router status.

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
RouterStatus RouterStatus { get; }
```

Property Value

Type: [RouterStatus](#) [[▶ 1687](#)]

The router status.

Reference

[IAmsRouter Interface](#) [[▶ 1646](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.2.1.5 IAmsRouter.Routes Property

Gets a read-only list of the routes

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IReadOnlyList<Route> Routes { get; }
```

Property Value

Type: [IReadOnlyList.Route](#) [[▶ 1654](#)].

The routes.

Reference







[IAmsRouter Interface](#) [[▶ 1646](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.2.2 IAmsRouter Methods

The [IAmsRouter](#) [[▶ 1646](#)] type exposes the following members.

Methods

	Name	Description
	AddRoute [▶ 1650]	Adds a dynamic Route [▶ 1654]
	RemoveRoute(AmsNetId) [▶ 1651]	Removes a dynamic Route [▶ 1654]
	RemoveRoute(String) [▶ 1651]	Removes a dynamic Route [▶ 1654]
	StartAsync [▶ 1652]	Starts the IAmsRouter [▶ 1646] asynchronously.
	Stop [▶ 1652]	Stops the IAmsRouter [▶ 1646].
	TryAddRoute [▶ 1653]	Adds a dynamic Route [▶ 1654]

Reference

[IAmsRouter Interface](#) [[▶ 1646](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.2.2.1 IAmsRouter.AddRoute Method

Adds a dynamic [Route](#) [[▶ 1654](#)]

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: [TwinCAT.Ads.TcpRouter](#) (in [TwinCAT.Ads.TcpRouter.dll](#)) Version: 6.0.328+39e3229

Syntax**C#**

```
void AddRoute (
    Route route
)
```

Parameters


route	Type: TwinCAT.Ads.TcpRouter.Route [▶ 1654] The route.
-------	--


Reference

[IAmsRouter Interface](#) [[▶ 1646](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.2.2.2 IAmsRouter.RemoveRoute Method**Overload List**

	Name	Description
	RemoveRoute(AmsNetId) [▶ 1651]	Removes a dynamic Route [▶ 1654]

	Name	Description
	RemoveRoute(String) [▸ 1651]	Removes a dynamic Route [▸ 1654]

Reference

[IAmsRouter Interface](#) [▸ 1646]

[TwinCAT.Ads.TcpRouter Namespace](#) [▸ 1619]

6.6.2.2.1 IAmsRouter.RemoveRoute Method (String)

Removes a dynamic [Route](#) [▸ 1654]

Namespace: [TwinCAT.Ads.TcpRouter](#) [▸ 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool RemoveRoute(
    string route
)
```

Parameters

route	Type: System.String The route.
-------	---

Return Value

Type: [Boolean](#)

true if removed, false otherwise.

Reference

[IAmsRouter Interface](#) [▸ 1646]

[RemoveRoute Overload](#) [▸ 1650]

[TwinCAT.Ads.TcpRouter Namespace](#) [▸ 1619]

6.6.2.2.2 IAmsRouter.RemoveRoute Method (AmsNetId)

Removes a dynamic [Route](#) [▸ 1654]

Namespace: [TwinCAT.Ads.TcpRouter](#) [▸ 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool RemoveRoute(
    AmsNetId netId
)
```

Parameters

netId	Type: TwinCAT.Ads.AmsNetId [▶ 767] The NetId of the route.
-------	---

Return Value

Type: [Boolean](#)
true if removed, false otherwise.

Reference

[IAmsRouter Interface](#) [▶ 1646]

[RemoveRoute Overload](#) [▶ 1650]

[TwinCAT.Ads.TcpRouter Namespace](#) [▶ 1619]

6.6.2.2.3 IAmsRouter.StartAsync Method

Starts the [IAmsRouter](#) [▶ 1646] asynchronously.

Namespace: [TwinCAT.Ads.TcpRouter](#) [▶ 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
Task StartAsync (
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken
--------	--

Return Value

Type: [Task](#)
Task.

Remarks

The asynchronous task runs as long until [Stop.](#) [▶ 1652] is called.

Reference

[IAmsRouter Interface](#) [▶ 1646]

[TwinCAT.Ads.TcpRouter Namespace](#) [▶ 1619]

6.6.2.2.4 IAmsRouter.Stop Method

Stops the [IAmsRouter](#) [▶ 1646].

Namespace: [TwinCAT.Ads.TcpRouter](#) [▶ 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void Stop()
```

Reference

[IAmsRouter Interface \[▶ 1646\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.2.2.5 IAmsRouter.TryAddRoute Method

Adds a dynamic [Route \[▶ 1654\]](#)

Namespace: [TwinCAT.Ads.TcpRouter \[▶ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool TryAddRoute (
    Route route
)
```

Parameters

route	Type: TwinCAT.Ads.TcpRouter.Route [▶ 1654] The route.
-------	--

Return Value

Type: [Boolean](#)
true if added, false otherwise.

Reference


[IAmsRouter Interface \[▶ 1646\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.2.3 IAmsRouter Events

The [IAmsRouter \[▶ 1646\]](#) type exposes the following members.

Events

	Name	Description
	RouterStatusChanged [▶ 1654]	Occurs when the router status changes.

Reference

[IAmsRouter Interface \[▶ 1646\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.2.3.1 IAmRouter.RouterStatusChanged Event

Occurs when the router status changes.

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
event EventHandler<RouterStatusChangedEventArgs> RouterStatusChanged
```

Value

Type: [System.EventHandler.RouterStatusChangedEventArgs](#) [[▶ 1688](#)].

Reference

[IAmRouter Interface](#) [[▶ 1646](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.3 Route Class

Class Route specifies a an AMS/ADS Route object.

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.TcpRouter.Route](#)

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229



Syntax

C#



```
public class Route
```







The Route type exposes the following members.

Constructors







	Name	Description
	Route(String, AmsNetId, IList<IPAddress>) [▶ 1656]	Initializes a new instance of the Route class.
	Route(String, AmsNetId, String) [▶ 1656]	Initializes a new instance of the Route class.

Properties



	Name	Description
	Address [▶ 1657]	Gets the address of the Route.
	HostName [▶ 1658]	Gets the Host name of the route.

	Name	Description
	IPAddresses [▶ 1658]	Gets the resolved IP Addresses belonging to this Route.
	IsHostNameAddress ed [▶ 1659]	Gets a value indicating whether this instance is an HostName Route.
	IsIPAddressed [▶ 1659]	Gets a value indicating whether the Route address specifies an IPAddress
	IsResolved [▶ 1659]	Gets a value indicating whether the IPAddresses are resolved.
	Name [▶ 1660]	Gets the name of the Route.
	NetId [▶ 1660]	Gets the AmsNetId [▶ 767] of the route.

Methods

	Name	Description
	Equals [▶ 1661]	Determines whether the specified Object is equal to this instance. (Overrides Object.Equals(Object) .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode [▶ 1662]	Returns a hash code for this instance. (Overrides Object.GetHashCode() .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString [▶ 1662]	Returns a String that represents this instance. (Overrides Object.ToString() .)

Operators


	Name	Description
	Equality [▶ 1663]	Implements the == operator.
	Inequality [▶ 1663]	Implements the != operator.


Reference

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.3.1 Route Constructor

Overload List

	Name	Description
	Route (String , AmsNetId , IList.IPAddress) [▶ 1656]	Initializes a new instance of the Route [▶ 1654] class.

	Name	Description
	Route(String, AmsNetId, String) [► 1656]	Initializes a new instance of the Route [► 1654] class.

Reference

[Route Class \[► 1654\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[► 1619\]](#)

6.6.3.1.1 Route Constructor (String, AmsNetId, IList.IPAddress.)

Initializes a new instance of the [Route \[► 1654\]](#) class.

Namespace: [TwinCAT.Ads.TcpRouter \[► 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Route(
    string name,
    AmsNetId netId,
    IList<IPAddress> ips
)
```

Parameters

name	Type: System.String The name of the route.
netId	Type: TwinCAT.Ads.AmsNetId [► 767] The net identifier.
ips	Type: System.Collections.Generic.IList.IPAddress. The ips.

Reference

[Route Class \[► 1654\]](#)

[Route Overload \[► 1655\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[► 1619\]](#)

6.6.3.1.2 Route Constructor (String, AmsNetId, String)

Initializes a new instance of the [Route \[► 1654\]](#) class.

Namespace: [TwinCAT.Ads.TcpRouter \[► 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Route(
    string name,
    AmsNetId netId,
    string address
)
```


Parameters

name	Type: System.String The name.
netId	Type: TwinCAT.Ads.AmsNetId [▶ 767] The AmsNetId of the route.
address	Type: System.String The address (HostName or IP)

Reference

[Route Class](#) [[▶ 1654](#)]









[Route Overload](#) [[▶ 1655](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.3.2 Route Properties

The [Route](#) [[▶ 1654](#)] type exposes the following members.

Properties

	Name	Description
	Address [▶ 1657]	Gets the address of the Route [▶ 1654].
	HostName [▶ 1658]	Gets the Host name of the route.
	IPAddresses [▶ 1658]	Gets the resolved IP Addresses belonging to this Route [▶ 1654].
	IsHostNameAddress [▶ 1659]	Gets a value indicating whether this instance is an HostName Route [▶ 1654].
	IsIPAddressed [▶ 1659]	Gets a value indicating whether the Route address specifies an IP Address
	IsResolved [▶ 1659]	Gets a value indicating whether the IPAddresses are resolved.
	Name [▶ 1660]	Gets the name of the Route [▶ 1654].
	NetId [▶ 1660]	Gets the AmsNetId [▶ 767] of the route.

Reference

[Route Class](#) [[▶ 1654](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.3.2.1 Route.Address Property

Gets the address of the [Route](#) [[▶ 1654](#)].

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: [TwinCAT.Ads.TcpRouter](#) (in [TwinCAT.Ads.TcpRouter.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public string Address { get; }
```

Property Value

Type: [String](#)
The address.

Remarks

The Address can be the IPAddress / or the host name of the target system.

Reference

[Route Class](#) [► 1654]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.3.2.2 Route.HostName Property

Gets the Host name of the route.

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string HostName { get; }
```

Property Value

Type: [String](#)
The name of the host.

Reference

[Route Class](#) [► 1654]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.3.2.3 Route.IPAddresses Property

Gets the resolved IP Addresses belonging to this [Route](#) [► 1654].

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IPAddress[]? IPAddresses { get; }
```

Property Value

Type: [.IPAddress](#).
The ip addresses.

Reference

[Route Class](#) [► 1654]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.3.2.4 Route.IsHostNameAddressed Property

Gets a value indicating whether this instance is an [HostName](#) [Route](#) [► 1654].

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsHostNameAddressed { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is an [HostName](#) Addressed route; otherwise, false.

Reference

[Route Class](#) [► 1654]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.3.2.5 Route.IsIPAddressed Property

Gets a value indicating whether the [Route](#) address specifies an [IPAddress](#)

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsIPAddressed { get; }
```

Property Value

Type: [Boolean](#)

true if the [Address](#) is an [IPAddress](#); otherwise when its specified as [HostAddress](#) false.

Reference

[Route Class](#) [► 1654]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.3.2.6 Route.IsResolved Property

Gets a value indicating whether the [IPAddresses](#) are resolved.

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsResolved { get; }
```

Property Value

Type: [Boolean](#)

true if the IP Addresses are resolved; otherwise, false.

Reference

[Route Class](#) [► 1654]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.3.2.7 Route.Name Property

Gets the name of the [Route](#) [► 1654].

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Name { get; }
```

Property Value

Type: [String](#)

The name.

Reference

[Route Class](#) [► 1654]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.3.2.8 Route.NetId Property

Gets the [AmsNetId](#) [► 767] of the route.

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsNetId NetId { get; }
```

Property Value

Type: [AmsNetId](#) [► 767]

The AmsNetId.

Remarks

The [AmsNetId](#) [► 767] is the unique identifier used for the route.

Reference







[Route Class \[▶ 1654\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.3.3 Route Methods

The [Route \[▶ 1654\]](#) type exposes the following members.

Methods

	Name	Description
	Equals [▶ 1661]	Determines whether the specified <u>Object</u> is equal to this instance. (Overrides <u>Object.Equals(Object).</u>)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from <u>Object.</u>)
	GetHashCode [▶ 1662]	Returns a hash code for this instance. (Overrides <u>Object.GetHashCode.</u>)
	GetType	Gets the <u>Type</u> of the current instance. (Inherited from <u>Object.</u>)
	MemberwiseClone	Creates a shallow copy of the current <u>Object</u> . (Inherited from <u>Object.</u>)
	ToString [▶ 1662]	Returns a <u>String</u> that represents this instance. (Overrides <u>Object.ToString.</u>)

Reference

[Route Class \[▶ 1654\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.3.3.1 Route.Equals Method

Determines whether the specified Object is equal to this instance.

Namespace: [TwinCAT.Ads.TcpRouter \[▶ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool Equals(
    Object? obj
)
```

Parameters

obj	Type: System.Object The object to compare with the current object.
-----	---

Return Value

Type: [Boolean](#)

true if the specified Object is equal to this instance; otherwise, false.

Reference

[Route Class](#) [► 1654]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.3.3.2 Route.GetHashCode Method

Returns a hash code for this instance.

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override int GetHashCode()
```

Return Value

Type: [Int32](#)

A hash code for this instance, suitable for use in hashing algorithms and data structures like a hash table.

Reference

[Route Class](#) [► 1654]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.3.3.3 Route.ToString Method

Returns a [String](#) that represents this instance.

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```

Return Value

Type: [String](#)

A [String](#) that represents this instance.

Reference





[Route Class](#) [► 1654]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.3.4 Route Operators

The [Route](#) [► 1654] type exposes the following members.

Operators

	Name	Description
 	Equality [▶ 1663]	Implements the == operator.
 	Inequality [▶ 1663]	Implements the != operator.

Reference

[Route Class \[▶ 1654\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.3.4.1 Route.Equality Operator

Implements the == operator.

Namespace: [TwinCAT.Ads.TcpRouter \[▶ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool operator ==(
    Route? o1,
    Route? o2
)
```

Parameters

o1	Type: TwinCAT.Ads.TcpRouter.Route [▶ 1654] The o1.
o2	Type: TwinCAT.Ads.TcpRouter.Route [▶ 1654] The o2.

Return Value

Type: [Boolean](#)
The result of the operator.

Reference

[Route Class \[▶ 1654\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.3.4.2 Route.Inequality Operator

Implements the != operator.

Namespace: [TwinCAT.Ads.TcpRouter \[▶ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool operator !=(
    Route? o1,
    Route? o2
)
```

Parameters

o1	Type: TwinCAT.Ads.TcpRouter.Route [► 1654] The o1.
o2	Type: TwinCAT.Ads.TcpRouter.Route [► 1654] The o2.

Return Value

Type: [Boolean](#)
The result of the operator.

Reference

[Route Class](#) [► 1654]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.4 RouteCollection Class

Collection of routes.

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.TcpRouter.RouteCollection](#)

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]

Assembly: [TwinCAT.Ads.TcpRouter](#) (in [TwinCAT.Ads.TcpRouter.dll](#)) Version: 6.0.328+39e3229



Syntax

C#




```
public class RouteCollection : IList<Route>,
    ICollection<Route>, IEnumerable<Route>, IEnumerable
```

The [RouteCollection](#) type exposes the following members.






















Constructors

	Name	Description
	RouteCollection. [► 1666]	Initializes a new instance of the RouteCollection class.
	RouteCollection(IEnumerable.Route.) [► 1666]	Initializes a new instance of the RouteCollection class (copy constructor)

Properties

	Name	Description
	Count [▶ 1667]	Gets the number of elements contained in the ICollection.T.
	IsReadOnly [▶ 1667]	Gets a value indicating whether the ICollection.T. is read-only.
	Item [▶ 1668]	Gets or sets the Route [▶ 1654] at the specified index.

Methods



	Name	Description
	Add [▶ 1670]	Adds an item to the ICollection.T.
	AddRange [▶ 1670]	Adds a range of routes.
	AsReadOnly [▶ 1677]	Ases the read only.
	Clear [▶ 1671]	Removes all items from the ICollection.T.
	Clone [▶ 1671]	Clones this instance.
	Contains [▶ 1671]	Determines whether the ICollection.T. contains a specific value.
	CopyTo [▶ 1672]	Copies the elements of the ICollection.T. to an Array , starting at a particular Array index.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 1673]	Returns an enumerator that iterates through the collection.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 1673]	Determines the index of a specific item in the IList.T.
	Insert [▶ 1674]	Inserts an item to the IList.T. at the specified index.
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove(AmsNetId) [▶ 1675]	Removes the specified AmsNetId [▶ 767] from the RouteCollection.
	Remove(Route) [▶ 1676]	Removes the first occurrence of a specific object from the ICollection.T.
	Remove(String) [▶ 1675]	Removes the specified route from the RouteCollection.
	RemoveAt [▶ 1676]	Removes the IList.T. item at the specified index.
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryAdd [▶ 1677]	Tries to add the route to the RouteCollection.

Reference

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.4.1 RouteCollection Constructor

Overload List

	Name	Description
	RouteCollection. [► 1666]	Initializes a new instance of the RouteCollection [► 1664] class.
	RouteCollection(IEn umerable.Route.) [► 1666]	Initializes a new instance of the RouteCollection [► 1664] class (copy constructor)

Reference

[RouteCollection Class \[► 1664\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[► 1619\]](#)

6.6.4.1.1 RouteCollection Constructor

Initializes a new instance of the [RouteCollection \[► 1664\]](#) class.

Namespace: [TwinCAT.Ads.TcpRouter \[► 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RouteCollection()
```

Reference

[RouteCollection Class \[► 1664\]](#)

[RouteCollection Overload \[► 1666\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[► 1619\]](#)

6.6.4.1.2 RouteCollection Constructor (IEnumerable.Route.)

Initializes a new instance of the [RouteCollection \[► 1664\]](#) class (copy constructor)

Namespace: [TwinCAT.Ads.TcpRouter \[► 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RouteCollection(
    IEnumerable<Route> routes
)
```

Parameters

routes	Type: System.Collections.Generic.IEnumerable.Route [► 1654] . The routes.
--------	--

Reference

[RouteCollection Class \[▸ 1664\]](#)




[RouteCollection Overload \[▸ 1666\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▸ 1619\]](#)

6.6.4.2 RouteCollection Properties

The [RouteCollection \[▸ 1664\]](#) type exposes the following members.

Properties

	Name	Description
	Count [▸ 1667]	Gets the number of elements contained in the ICollection.T.
	IsReadOnly [▸ 1667]	Gets a value indicating whether the ICollection.T. is read-only.
	Item [▸ 1668]	Gets or sets the Route [▸ 1654] at the specified index.

Reference

[RouteCollection Class \[▸ 1664\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▸ 1619\]](#)

6.6.4.2.1 RouteCollection.Count Property

Gets the number of elements contained in the [ICollection.T.](#)

Namespace: [TwinCAT.Ads.TcpRouter \[▸ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Count { get; }
```

Property Value

Type: [Int32](#)

The count.

Implements

[ICollection.T..Count](#)

Reference

[RouteCollection Class \[▸ 1664\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▸ 1619\]](#)

6.6.4.2.2 RouteCollection.IsReadOnly Property

Gets a value indicating whether the [ICollection.T.](#) is read-only.

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsReadOnly { get; }
```

Property Value

Type: Boolean

true if this instance is read only; otherwise, false.

Implements

ICollection.T..IsReadOnly

Reference

[RouteCollection Class](#) [[▶ 1664](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.4.2.3 RouteCollection.Item Property

Gets or sets the [Route](#) [[▶ 1654](#)] at the specified index.

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Route this[
    int index
] { get; set; }
```

Parameters

index	Type: System.Int32 The index.
-------	----------------------------------

Return Value

Type: [Route](#) [[▶ 1654](#)]

Route.

Implements

ICollection.T..Item.Int32.

Exceptions

Exception	Condition
NotImplementedException	

Reference

[RouteCollection Class \[▶ 1664\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.4.3 RouteCollection Methods

The [RouteCollection \[▶ 1664\]](#) type exposes the following members.

Methods

	Name	Description
	Add [▶ 1670]	Adds an item to the ICollection.T.
	AddRange [▶ 1670]	Adds a range of routes.
	AsReadOnly [▶ 1677]	Ases the read only.
	Clear [▶ 1671]	Removes all items from the ICollection.T.
	Clone [▶ 1671]	Clones this instance.
	Contains [▶ 1671]	Determines whether the ICollection.T. contains a specific value.
	CopyTo [▶ 1672]	Copies the elements of the ICollection.T. to an Array , starting at a particular Array index.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 1673]	Returns an enumerator that iterates through the collection.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 1673]	Determines the index of a specific item in the IList.T.
	Insert [▶ 1674]	Inserts an item to the IList.T. at the specified index.
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove(AmsNetId) [▶ 1675]	Removes the specified AmsNetId [▶ 767] from the RouteCollection [▶ 1664] .
	Remove(Route) [▶ 1676]	Removes the first occurrence of a specific object from the ICollection.T.
	Remove(String) [▶ 1675]	Removes the specified route from the RouteCollection [▶ 1664] .
	RemoveAt [▶ 1676]	Removes the IList.T. item at the specified index.
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryAdd [▶ 1677]	Tries to add the route to the RouteCollection [▶ 1664] .

Reference

[RouteCollection Class \[▶ 1664\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.4.3.1 RouteCollection.Add Method

Adds an item to the [ICollection.T.](#).

Namespace: [TwinCAT.Ads.TcpRouter \[▶ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Add(
    Route item
)
```

Parameters

item	Type: TwinCAT.Ads.TcpRouter.Route [▶ 1654] The object to add to the ICollection.T. .
------	---

Implements

[ICollection.T.Add\(T\)](#)

Exceptions

Exception	Condition
ArgumentException	Cannot add item

Reference

[RouteCollection Class \[▶ 1664\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.4.3.2 RouteCollection.AddRange Method

Adds a range of routes.

Namespace: [TwinCAT.Ads.TcpRouter \[▶ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void AddRange(
    IEnumerable<Route> routes
)
```

Parameters

routes	Type: System.Collections.Generic.IEnumerable.Route [▶ 1654] . The routes.
--------	--

Reference

[RouteCollection Class \[▶ 1664\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▸ 1619\]](#)

6.6.4.3.3 RouteCollection.Clear Method

Removes all items from the [ICollection.T.](#)

Namespace: [TwinCAT.Ads.TcpRouter \[▸ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Clear()
```

Implements

[ICollection.T..Clear.](#)

Reference

[RouteCollection Class \[▸ 1664\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▸ 1619\]](#)

6.6.4.3.4 RouteCollection.Clone Method

Clones this instance.

Namespace: [TwinCAT.Ads.TcpRouter \[▸ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RouteCollection Clone()
```

Return Value

Type: [RouteCollection \[▸ 1664\]](#)

RouteCollection.

Reference

[RouteCollection Class \[▸ 1664\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▸ 1619\]](#)

6.6.4.3.5 RouteCollection.Contains Method

Determines whether the [ICollection.T.](#) contains a specific value.

Namespace: [TwinCAT.Ads.TcpRouter \[▸ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(
    Route item
)
```

Parameters

item	Type: TwinCAT.Ads.TcpRouter.Route [▶ 1654] The object to locate in the ICollection.T.
------	--

Return Value

Type: [Boolean](#)

. if item is found in the [ICollection.T.](#); otherwise, ..

Implements

[ICollection.T.Contains\(T\)](#)

Reference

[RouteCollection Class](#) [[▶ 1664](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.4.3.6 RouteCollection.CopyTo Method

Copies the elements of the [ICollection.T.](#) to an [Array](#), starting at a particular [Array](#) index.

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void CopyTo(
    Route[] array,
    int arrayIndex
)
```

Parameters

array	Type: .TwinCAT.Ads.TcpRouter.Route [▶ 1654]. The one-dimensional Array that is the destination of the elements copied from ICollection.T. . The Array must have zero-based indexing.
arrayIndex	Type: System.Int32 The zero-based index in array at which copying begins.

Implements

[ICollection.T.CopyTo\(T, Int32\)](#)

Reference

[RouteCollection Class](#) [[▶ 1664](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [▶ 1619]

6.6.4.3.7 RouteCollection.GetEnumerator Method

Returns an enumerator that iterates through the collection.

Namespace: [TwinCAT.Ads.TcpRouter](#) [▶ 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IEnumerator<Route> GetEnumerator()
```

Return Value

Type: [IEnumerator.Route](#) [▶ 1654].

An enumerator that can be used to iterate through the collection.

Implements

[IEnumerable.T..GetEnumerator.](#)

Reference

[RouteCollection Class](#) [▶ 1664]

[TwinCAT.Ads.TcpRouter Namespace](#) [▶ 1619]

6.6.4.3.8 RouteCollection.IndexOf Method

Determines the index of a specific item in the [IList.T.](#).

Namespace: [TwinCAT.Ads.TcpRouter](#) [▶ 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int IndexOf(  
    Route item  
)
```

Parameters

item	Type: TwinCAT.Ads.TcpRouter.Route [▶ 1654] The object to locate in the IList.T. .
------	--

Return Value

Type: [Int32](#)

The index of item if found in the list; otherwise, -1.

Implements

[IList.T..IndexOf\(T\)](#)

Reference[RouteCollection Class \[▶ 1664\]](#)[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)**6.6.4.3.9 RouteCollection.Insert Method**

Inserts an item to the [IList.T.](#) at the specified index.

Namespace: [TwinCAT.Ads.TcpRouter \[▶ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public void Insert(
    int index,
    Route item
)
```

Parameters

index	Type: System.Int32 The zero-based index at which item should be inserted.
item	Type: TwinCAT.Ads.TcpRouter.Route [▶ 1654] The object to insert into the IList.T.




Implements

[IList.T..Insert\(Int32, T\)](#)

Exceptions

Exception	Condition
ArgumentException	Cannot add item

Reference[RouteCollection Class \[▶ 1664\]](#)[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)**6.6.4.3.10 RouteCollection.Remove Method****Overload List**

	Name	Description
	Remove(AmsNetId) [▶ 1675]	Removes the specified AmsNetId [▶ 767] from the RouteCollection [▶ 1664] .
	Remove(Route) [▶ 1676]	Removes the first occurrence of a specific object from the ICollection.T.
	Remove(String) [▶ 1675]	Removes the specified route from the RouteCollection [▶ 1664] .

Reference

[RouteCollection Class \[► 1664\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[► 1619\]](#)

6.6.4.3.10.1 RouteCollection.Remove Method (String)

Removes the specified route from the [RouteCollection \[► 1664\]](#).

Namespace: [TwinCAT.Ads.TcpRouter \[► 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Remove(  
    string routeName  
)
```

Parameters

routeName	Type: System.String Name of the route.
-----------	---

Return Value

Type: Boolean
true if removed, false otherwise.

Reference

[RouteCollection Class \[► 1664\]](#)

[Remove Overload \[► 1674\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[► 1619\]](#)

6.6.4.3.10.2 RouteCollection.Remove Method (AmsNetId)

Removes the specified [AmsNetId \[► 767\]](#) from the [RouteCollection \[► 1664\]](#).

Namespace: [TwinCAT.Ads.TcpRouter \[► 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Remove(  
    AmsNetId netId  
)
```

Parameters

netId	Type: TwinCAT.Ads.AmsNetId [► 767] The Ams Net ID..
-------	--

Return Value

Type: Boolean
true if removed, false otherwise.

Reference

[RouteCollection Class \[► 1664\]](#)

[Remove Overload \[► 1674\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[► 1619\]](#)

6.6.4.3.10.3 RouteCollection.Remove Method (Route)

Removes the first occurrence of a specific object from the ICollection.T..

Namespace: [TwinCAT.Ads.TcpRouter \[► 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Remove(  
    Route item  
)
```

Parameters

item	Type: TwinCAT.Ads.TcpRouter.Route [► 1654] The object to remove from the ICollection.T..
------	---

Return Value

Type: Boolean

. if item was successfully removed from the ICollection.T.; otherwise, .. This method also returns . if item is not found in the original ICollection.T..

Implements

ICollection.T..Remove(T)

Reference

[RouteCollection Class \[► 1664\]](#)

[Remove Overload \[► 1674\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[► 1619\]](#)

6.6.4.3.11 RouteCollection.RemoveAt Method

Removes the IList.T. item at the specified index.

Namespace: [TwinCAT.Ads.TcpRouter \[► 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void RemoveAt(  
    int index  
)
```

Parameters

index	Type: System.Int32 The zero-based index of the item to remove.
-------	---

Implements

IList.T..RemoveAt(Int32)

Reference

[RouteCollection Class \[► 1664\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[► 1619\]](#)

6.6.4.3.12 RouteCollection.TryAdd Method

Tries to add the route to the [RouteCollection \[► 1664\]](#).

Namespace: [TwinCAT.Ads.TcpRouter \[► 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryAdd(  
    Route route  
)
```

Parameters

route	Type: TwinCAT.Ads.TcpRouter.Route [► 1654] The route.
-------	--

Return Value

Type: Boolean
true if the route is added, false otherwise.

Reference

[RouteCollection Class \[► 1664\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[► 1619\]](#)

6.6.4.3.13 RouteCollection.AsReadOnly Method

Ases the read only.

Namespace: [TwinCAT.Ads.TcpRouter \[► 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IReadOnlyList<Route> AsReadOnly()
```

Return Value

Type: [IReadOnlyList.Route](#) [[▶ 1654](#)].
[ReadOnlyCollection<Route>](#).

Reference

[RouteCollection Class](#) [[▶ 1664](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.5 RouterException Class

An RouterException is thrown on communication errors in the [AmsTcpIpRouter](#) [[▶ 1620](#)] class.

Inheritance Hierarchy

System.Object
 System.Exception
 TwinCAT.Ads.TcpRouter.RouterException
 [TwinCAT.Ads.TcpRouter.RouterNotInitializedException](#) [[▶ 1682](#)]
 [TwinCAT.Ads.TcpRouter.RouterNotStartedException](#) [[▶ 1684](#)]

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229


Syntax

C#









```
[SerializableAttribute]
public class RouterException : Exception
```

The RouterException type exposes the following members.









Constructors

	Name	Description
	RouterException [▶ 1679]	Initializes a new instance of the RouterException class.


Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	<u>GetObjectData</u> [▶ 1681]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Overrides Exception.GetObjectData(SerializationInfo, StreamingContext).)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)

Reference

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.5.1 RouterException Constructor

Initializes a new instance of the [RouterException \[▶ 1678\]](#) class.

Namespace: [TwinCAT.Ads.TcpRouter \[▶ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected RouterException(
    SerializationInfo serializationInfo,
    StreamingContext streamingContext
)
```

Parameters

serializationInfo	Type: System.Runtime.Serialization.SerializationInfo The serialization information.
streamingContext	Type: System.Runtime.Serialization.StreamingContext The streaming context.

Reference









[RouterException Class \[▶ 1678\]](#)

[TwinCAT.Ads.TcpRouter Namespace](#) [▶ 1619]

6.6.5.2 RouterException Properties

The [RouterException](#) [▶ 1678] type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Reference









[RouterException Class](#) [▶ 1678]

[TwinCAT.Ads.TcpRouter Namespace](#) [▶ 1619]

6.6.5.3 RouterException Methods

The [RouterException](#) [▶ 1678] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData [▶ 1681]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Overrides Exception.GetObjectData(SerializationInfo, StreamingContext).)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Reference

[RouterException Class \[► 1678\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[► 1619\]](#)

6.6.5.3.1 RouterException.GetObjectData Method

When overridden in a derived class, sets the `SerializationInfo` with information about the exception.

Namespace: [TwinCAT.Ads.TcpRouter \[► 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override void GetObjectData(
    SerializationInfo info,
    StreamingContext context
)
```

Parameters

info	Type: System.Runtime.Serialization.SerializationInfo The <code>SerializationInfo</code> that holds the serialized object data about the exception being thrown.
context	Type: System.Runtime.Serialization.StreamingContext The <code>StreamingContext</code> that contains contextual information about the source or destination.

Implements

`ISerializable.GetObjectData(SerializationInfo, StreamingContext)`

Exceptions

Exception	Condition
<code>ArgumentNullException</code>	info

Reference


[RouterException Class \[► 1678\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[► 1619\]](#)

6.6.5.4 RouterException Events

The [RouterException \[► 1678\]](#) type exposes the following members.

Events

	Name	Description
	<code>SerializeObjectState</code>	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from <code>Exception</code> .)

Reference

[RouterException Class](#) [► 1678]

[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.6 RouterNotInitializedException Class

Class RouterNotInitializedException. Implements the [RouterException](#) [► 1678]

Inheritance Hierarchy

System.Object
 System.Exception
[TwinCAT.Ads.TcpRouter.RouterException](#) [► 1678]
 TwinCAT.Ads.TcpRouter.RouterNotInitializedException

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229









Syntax

C#





```
public class RouterNotInitializedException : RouterException
```





The RouterNotInitializedException type exposes the following members.

Properties


	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)

	Name	Description
	GetObjectData [▶ 1681]	When overridden in a derived class, sets the <code>SerializationInfo</code> with information about the exception. (Inherited from RouterException [▶ 1678].)
	GetType	Gets the runtime type of the current instance. (Inherited from <code>Exception</code> .)
	MemberwiseClone	Creates a shallow copy of the current <code>Object</code> . (Inherited from <code>Object</code> .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from <code>Exception</code> .)

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from <code>Exception</code> .)

Reference









[TwinCAT.Ads.TcpRouter Namespace](#) [▶ 1619]

[TwinCAT.Ads.TcpRouter.RouterException](#) [▶ 1678]

6.6.6.1 RouterNotInitializedException Properties

The [RouterNotInitializedException](#) [▶ 1682] type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from <code>Exception</code> .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from <code>Exception</code> .)
	HResult	Gets or sets <code>HRESULT</code> , a coded numerical value that is assigned to a specific exception. (Inherited from <code>Exception</code> .)
	InnerException	Gets the <code>Exception</code> instance that caused the current exception. (Inherited from <code>Exception</code> .)
	Message	Gets a message that describes the current exception. (Inherited from <code>Exception</code> .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from <code>Exception</code> .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from <code>Exception</code> .)
	TargetSite	Gets the method that throws the current exception. (Inherited from <code>Exception</code> .)

Reference









[RouterNotInitializedException Class](#) [▶ 1682]

[TwinCAT.Ads.TcpRouter Namespace](#) [▶ 1619]

6.6.6.2 RouterNotInitializedException Methods

The [RouterNotInitializedException](#) [▶ 1682] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	<u>GetObjectData</u> [▶ 1681]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from <u>RouterException</u> [▶ 1678].)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Reference


[RouterNotInitializedException Class \[▶ 1682\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.6.3 RouterNotInitializedException Events

The [RouterNotInitializedException](#) [▶ 1682] type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)

Reference

[RouterNotInitializedException Class \[▶ 1682\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.7 RouterNotStartedException Class

Class RouterNotStartedException. Implements the [RouterException](#) [▶ 1678]

Inheritance Hierarchy

System.Object
 System.Exception
[TwinCAT.Ads.TcpRouter.RouterException](#) [▶ 1678]
 TwinCAT.Ads.TcpRouter.RouterNotStartedException

Namespace: [TwinCAT.Ads.TcpRouter](#) [▶ 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229









Syntax

C#









```
public class RouterNotStartedException : RouterException
```

The RouterNotStartedException type exposes the following members.


Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData [▶ 1681]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from RouterException [▶ 1678].)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)

Reference









[TwinCAT.Ads.TcpRouter Namespace \[► 1619\]](#)

[TwinCAT.Ads.TcpRouter.RouterException \[► 1678\]](#)

6.6.7.1 RouterNotStartedException Properties

The [RouterNotStartedException \[► 1684\]](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Reference








[RouterNotStartedException Class \[► 1684\]](#)


[TwinCAT.Ads.TcpRouter Namespace \[► 1619\]](#)

6.6.7.2 RouterNotStartedException Methods

The [RouterNotStartedException \[► 1684\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData [► 1681]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from RouterException [► 1678] .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)

	Name	Description
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Reference


[RouterNotStartedException Class \[▶ 1684\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.7.3 RouterNotStartedException Events

The [RouterNotStartedException \[▶ 1684\]](#) type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)

Reference

[RouterNotStartedException Class \[▶ 1684\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.8 RouterStatus Enumeration

Enum RouterStatus

Namespace: [TwinCAT.Ads.TcpRouter \[▶ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum RouterStatus
```

Members

	Member name	Value	Description
	None	0	None / Uninitialized
	Initializing	1	Initializing
	Starting	2	Router Starting
	Started	3	Router Started / Executing
	Stopping	4	Router Stopping.
	Stopped	5	Router stopped.

Remarks

Indicates the status of the router.

Reference

[TwinCAT.Ads.TcpRouter Namespace \[▶ 1619\]](#)

6.6.9 RouterStatusChangedEventArgs Class

Event Arguments sent when the [RouterStatusChanged](#) [[▶ 1654](#)]. Implements the EventArgs

Inheritance Hierarchy

System.Object
 System.EventArgs
 TwinCAT.Ads.TcpRouter.RouterStatusChangedEventArgs

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229


Syntax

C#







```
public class RouterStatusChangedEventArgs : EventArgs
```

The RouterStatusChangedEventArgs type exposes the following members.


Constructors

	Name	Description
	RouterStatusChangedEventArgs [▶ 1688]	Initializes a new instance of the RouterStatusChangedEventArgs class.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Fields

	Name	Description
	RouterStatus [▶ 1690]	The router status

Reference

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

System.EventArgs

[IAmsRouter.RouterStatusChanged](#) [[▶ 1654](#)]

6.6.9.1 RouterStatusChangedEventArgs Constructor

Initializes a new instance of the [RouterStatusChangedEventArgs](#) [[▶ 1688](#)] class.

Namespace: [TwinCAT.Ads.TcpRouter](#) [▶ 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RouterStatusChangedEventArgs (
    RouterStatus status
)
```

Parameters

status	Type: TwinCAT.Ads.TcpRouter.RouterStatus [▶ 1687] The status.
--------	--

Reference







[RouterStatusChangedEventArgs Class](#) [▶ 1688]

[TwinCAT.Ads.TcpRouter Namespace](#) [▶ 1619]

6.6.9.2 RouterStatusChangedEventArgs Methods

The [RouterStatusChangedEventArgs](#) [▶ 1688] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference


[RouterStatusChangedEventArgs Class](#) [▶ 1688]

[TwinCAT.Ads.TcpRouter Namespace](#) [▶ 1619]

6.6.9.3 RouterStatusChangedEventArgs Fields

The [RouterStatusChangedEventArgs](#) [▶ 1688] type exposes the following members.

Fields

	Name	Description
	RouterStatus [▶ 1690]	The router status

Reference[RouterStatusChangedEventArgs Class](#) [► 1688][TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]**6.6.9.3.1 RouterStatusChangedEventArgs.RouterStatus Field**

The router status

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]**Assembly:** TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public readonly RouterStatus RouterStatus
```

Field ValueType: [RouterStatus](#) [► 1687]**Reference**[RouterStatusChangedEventArgs Class](#) [► 1688][TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]**6.6.10 StaticRoutesXmlConfigurationBuilderExtension Class**

Extension class adding StaticRoutes.xml file reading to the IConfigurationBuilder.

Inheritance Hierarchy

System.Object


TwinCAT.Ads.TcpRouter.StaticRoutesXmlConfigurationBuilderExtension

Namespace: [TwinCAT.Ads.TcpRouter](#) [► 1619]**Assembly:** TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static class StaticRoutesXmlConfigurationBuilderExtension
```

The StaticRoutesXmlConfigurationBuilderExtension type exposes the following members.

Methods


	Name	Description
	AddStaticRoutesXmlConfiguration [► 1691]	Adds the static routes XML configuration.

Reference[TwinCAT.Ads.TcpRouter Namespace](#) [► 1619]

6.6.10.1 StaticRoutesXmlConfigurationBuilderExtension Methods

The [StaticRoutesXmlConfigurationBuilderExtension](#) [▸ 1690] type exposes the following members.

Methods

	Name	Description
	AddStaticRoutesXmlConfiguration [▸ 1691]	Adds the static routes XML configuration.

Reference

[StaticRoutesXmlConfigurationBuilderExtension Class](#) [▸ 1690]

[TwinCAT.Ads.TcpRouter Namespace](#) [▸ 1619]

6.6.10.1.1 StaticRoutesXmlConfigurationBuilderExtension.AddStaticRoutesXmlConfiguration Method

Adds the static routes XML configuration.

Namespace: [TwinCAT.Ads.TcpRouter](#) [▸ 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IConfigurationBuilder AddStaticRoutesXmlConfiguration(
    this IConfigurationBuilder builder
)
```

Parameters

builder	Type: Microsoft.Extensions.Configuration.IConfigurationBuilder The builder.
---------	--

Return Value

Type: IConfigurationBuilder
IConfigurationBuilder.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type IConfigurationBuilder. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[StaticRoutesXmlConfigurationBuilderExtension Class](#) [▸ 1690]

[TwinCAT.Ads.TcpRouter Namespace](#) [▸ 1619]

6.6.11 StaticRoutesXmlConfigurationProvider Class

Class StaticRoutesXmlConfigurationProvider. Implements the ConfigurationProvider

Inheritance Hierarchy

System.Object

Microsoft.Extensions.Configuration.ConfigurationProvider

TwinCAT.Ads.TcpRouter.StaticRoutesXmlConfigurationProvider

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229


Syntax

C#


```
public class StaticRoutesXmlConfigurationProvider : ConfigurationProvider
```

The StaticRoutesXmlConfigurationProvider type exposes the following members.













Constructors

	Name	Description
	StaticRoutesXmlConfigurationProvider [▶ 1693]	Initializes a new instance of the StaticRoutesXmlConfigurationProvider class.

Properties

	Name	Description
	Data	The configuration key value pairs for this provider. (Inherited from ConfigurationProvider.)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetChildKeys [▶ 1694]	Returns the list of keys that this provider has. (Overrides ConfigurationProvider.GetChildKeys(IEnumerable<String>, String).)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetReloadToken	Returns a IChangeToken that can be used to listen when this provider is reloaded. (Inherited from ConfigurationProvider.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	Load [▶ 1695]	Loads (or reloads) the data for this provider. (Overrides ConfigurationProvider.Load..)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	OnReload	Triggers the reload change token and creates a new one. (Inherited from ConfigurationProvider.)
	Set	Sets a value for a given key. (Inherited from ConfigurationProvider.)
	ToString	Generates a string representing this provider name and relevant details. (Inherited from ConfigurationProvider.)
	TryGet [▶ 1695]	Attempts to find a value with the given key, returns true if one is found, false otherwise. (Overrides ConfigurationProvider.TryGet(String, String).)

Reference

[TwinCAT.Ads.TcpRouter Namespace |> 1619](#)

Microsoft.Extensions.Configuration.ConfigurationProvider

6.6.11.1 StaticRoutesXmlConfigurationProvider Constructor

Initializes a new instance of the [StaticRoutesXmlConfigurationProvider |> 1691](#) class.

Namespace: [TwinCAT.Ads.TcpRouter |> 1619](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public StaticRoutesXmlConfigurationProvider()
```

Reference


[StaticRoutesXmlConfigurationProvider Class |> 1691](#)

[TwinCAT.Ads.TcpRouter Namespace |> 1619](#)

6.6.11.2 StaticRoutesXmlConfigurationProvider Properties

The [StaticRoutesXmlConfigurationProvider |> 1691](#) type exposes the following members.

Properties

	Name	Description
	Data	The configuration key value pairs for this provider. (Inherited from ConfigurationProvider.)

Reference






[StaticRoutesXmlConfigurationProvider Class |> 1691](#)








[TwinCAT.Ads.TcpRouter Namespace |> 1619](#)

6.6.11.3 StaticRoutesXmlConfigurationProvider Methods

The [StaticRoutesXmlConfigurationProvider |> 1691](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetChildKeys > 1694	Returns the list of keys that this provider has. (Overrides ConfigurationProvider.GetChildKeys(IEnumerable.String., String.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetReloadToken	Returns a IChangeToken that can be used to listen when this provider is reloaded. (Inherited from ConfigurationProvider.)

	Name	Description
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	Load [► 1695]	Loads (or reloads) the data for this provider. (Overrides ConfigurationProvider.Load..)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	OnReload	Triggers the reload change token and creates a new one. (Inherited from ConfigurationProvider.)
	Set	Sets a value for a given key. (Inherited from ConfigurationProvider.)
	ToString	Generates a string representing this provider name and relevant details. (Inherited from ConfigurationProvider.)
	TryGet [► 1695]	Attempts to find a value with the given key, returns true if one is found, false otherwise. (Overrides ConfigurationProvider.TryGet(String, String..))

Reference

[StaticRoutesXmlConfigurationProvider Class \[► 1691\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[► 1619\]](#)

6.6.11.3.1 StaticRoutesXmlConfigurationProvider.GetChildKeys Method

Returns the list of keys that this provider has.

Namespace: [TwinCAT.Ads.TcpRouter \[► 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override IEnumerable<string> GetChildKeys(
    IEnumerable<string> earlierKeys,
    string? parentPath
)
```

Parameters

earlierKeys	Type: System.Collections.Generic.IEnumerable.String. The earlier keys that other providers contain.
parentPath	Type: System.String The path for the parent IConfiguration.

Return Value

Type: IEnumerable.String.
The list of keys for this provider.

Implements

IConfigurationProvider.GetChildKeys(IEnumerable.String., String)

Reference

[StaticRoutesXmlConfigurationProvider Class \[► 1691\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[► 1619\]](#)

6.6.11.3.2 StaticRoutesXmlConfigurationProvider.Load Method

Loads (or reloads) the data for this provider.

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override void Load()
```

Implements

IConfigurationProvider.Load.

Reference

[StaticRoutesXmlConfigurationProvider Class](#) [[▶ 1691](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.11.3.3 StaticRoutesXmlConfigurationProvider.TryGet Method

Attempts to find a value with the given key, returns true if one is found, false otherwise.

Namespace: [TwinCAT.Ads.TcpRouter](#) [[▶ 1619](#)]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TryGet(  
    string key,  
    out string?? value  
)
```

Parameters

key	Type: System.String The key to lookup.
value	Type: System.String. The value found at key if one is found.

Return Value

Type: Boolean

True if key has a value, false otherwise.

Implements

IConfigurationProvider.TryGet(String, String.)

Reference

[StaticRoutesXmlConfigurationProvider Class](#) [[▶ 1691](#)]

[TwinCAT.Ads.TcpRouter Namespace](#) [[▶ 1619](#)]

6.6.12 StaticRoutesXmlConfigurationSource Class

StaticRoutes Configuration Sources Implements the IConfigurationSource

Inheritance Hierarchy

System.Object

TwinCAT.Ads.TcpRouter.StaticRoutesXmlConfigurationSource

Namespace: [TwinCAT.Ads.TcpRouter](#) [▶ 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229


Syntax

C#








```
public class StaticRoutesXmlConfigurationSource : IConfigurationSource
```

The StaticRoutesXmlConfigurationSource type exposes the following members.

Constructors

	Name	Description
	StaticRoutesXmlConfigurationSource [▶ 1696]	

Methods

	Name	Description
	Build [▶ 1697]	Builds the IConfigurationProvider for this source.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[TwinCAT.Ads.TcpRouter Namespace](#) [▶ 1619]

Microsoft.Extensions.Configuration.IConfigurationSource

6.6.12.1 StaticRoutesXmlConfigurationSource Constructor

Namespace: [TwinCAT.Ads.TcpRouter](#) [▶ 1619]

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public StaticRoutesXmlConfigurationSource()
```


Reference








[StaticRoutesXmlConfigurationSource Class \[▸ 1696\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▸ 1619\]](#)

6.6.12.2 StaticRoutesXmlConfigurationSource Methods

The [StaticRoutesXmlConfigurationSource \[▸ 1696\]](#) type exposes the following members.

Methods

	Name	Description
	Build [▸ 1697]	Builds the IConfigurationProvider for this source.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[StaticRoutesXmlConfigurationSource Class \[▸ 1696\]](#)

[TwinCAT.Ads.TcpRouter Namespace \[▸ 1619\]](#)

6.6.12.2.1 StaticRoutesXmlConfigurationSource.Build Method

Builds the IConfigurationProvider for this source.

Namespace: [TwinCAT.Ads.TcpRouter \[▸ 1619\]](#)

Assembly: TwinCAT.Ads.TcpRouter (in TwinCAT.Ads.TcpRouter.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IConfigurationProvider Build(
    IConfigurationBuilder builder
)
```

Parameters

builder	Type: Microsoft.Extensions.Configuration.IConfigurationBuilder The IConfigurationBuilder.
---------	--

Return Value

Type: IConfigurationProvider
An IConfigurationProvider

Implements

`IConfigurationSource.Build(IConfigurationBuilder)`

Reference



















[StaticRoutesXmlConfigurationSource Class \[▸ 1696\]](#)












[TwinCAT.Ads.TcpRouter Namespace \[▸ 1619\]](#)

6.7 TwinCAT.Ads.TypeSystem Namespace




Root namespace for the ADS type system.

Classes

	Class	Description
	AliasType [▸ 1699]	Alias DataType
	AlignmentCalculator [▸ 1944]	Class calculating alignments.
	ArrayType [▸ 1707]	Represents an Array DataType
	BitMappingType [▸ 1718]	Helper Data Type to implement Bit mapping types.
	DataType [▸ 1721]	DataType class
	EnumType.T. [▸ 1738]	Enum DataType [▸ 1721] .
	Field [▸ 1750]	Represents a field of an Struct/Alias/Union
	FluentRpcMethodExtension [▸ 1947]	Class FluentRpcMethodExtension .
	FluentStructTypeExtension [▸ 1955]	Class FluentStructTypeExtension .
	Instance [▸ 1765]	Instance implementation
	InterfaceType [▸ 1957]	Represents an interface type
	Member [▸ 1781]	Represents a member of an StructType [▸ 1846]
	PCCHType [▸ 1787]	Class PCCHType . This class cannot be inherited. Implements the PointerType [▸ 1790]
	PointerType [▸ 1790]	Represents a pointer type.
	PrimitiveType [▸ 1796]	Class PrimitiveType .
	PVoidType [▸ 1801]	Class PVoidType . This class cannot be inherited. Implements the PointerType [▸ 1790]
	ReferenceType [▸ 1804]	Represents a reference type
	RpcMethod [▸ 1813]	RPC Method Description

	Class	Description
	RpcMethodParameter [▶ 1824]	Class RpcMethodParameter.
	StringType [▶ 1840]	String DataType
	StructType [▶ 1846]	Represents a struct type
	SubRangeType.T. [▶ 1856]	Represents a SubRangType
	Symbol [▶ 1863]	Symbol class
 	SymbolIterator [▶ 1916]	Iterator class for enumerations of Symbols [▶ 2691].
	SymbolLoaderFactory [▶ 1923]	The class SymbolLoaderFactory [▶ 1923] is used to create a new instance of the AdsSymbolLoader initialized to the parametrized mode (SymbolBrowser V2 , new Version)
	UnionInstance [▶ 1992]	Class representing a Union Instance
	UnionType [▶ 1933]	Represents a union type
	WStringType [▶ 1938]	Represents an Unicode string (Wide string)

Interfaces

	Interface	Description
	IAdsSymbol [▶ 1755]	Interface IAdsSymbol
	IAdsSymbolLoader [▶ 1760]	Symbol Loader interface
	IContextMaskProvider [▶ 1764]	Interface IContextMaskProvider

6.7.1 AliasType Class

Alias DataType

Inheritance Hierarchy

System.Object

[TwinCAT.Ads.TypeSystem.DataType](#) [[▶ 1721](#)]

[TwinCAT.Ads.TypeSystem.AliasType](#)

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#





```
public sealed class AliasType : DataType,
    IAliasType, IDatatype, IBitSize
```

The AliasType type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BaseType [▶ 1703]	Gets the Base Type
	BaseTypeName [▶ 1703]	Gets the BaseType name
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the DataType (Inherited from DataType [▶ 1721].)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DataType [▶ 1721].)
	ManagedType [▶ 1705]	Gets the corresponding .NET Type if attached. (Overrides DataType.ManagedType [▶ 1733].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString [▶ 1706]	Returns a String that represents this instance. (Overrides DataType.ToString [▶ 1738].)

Extension Methods

	Name	Description
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)

	Name	Description
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021] .)
	IsContainer. [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021] .)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021] .)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021] .)
	IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021] .)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021] .)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021] .)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021] .)
	ResolvableAsPrimitive. [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021] .)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021] .)
	Resolve. [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021] .)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021] .)

Reference

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

Also see about this

- ▣ [AliasType.IsContainer Property \[▶ 1703\]](#)
- ▣ [DataType.IsContainer Property \[▶ 1731\]](#)
- ▣ [DataType.IsPointer Property \[▶ 1732\]](#)
- ▣ [AliasType.IsPrimitive Property \[▶ 1704\]](#)
- ▣ [DataType.IsPrimitive Property \[▶ 1732\]](#)
- ▣ [DataType.IsReference Property \[▶ 1733\]](#)

6.7.1.1 AliasType Properties

The [AliasType \[▸ 1699\]](#) type exposes the following members.

Properties







	Name	Description
	Attributes [▸ 1727]	Gets the attributes of the IDataType [▸ 2475] (Inherited from DataType [▸ 1721].)
	BaseType [▸ 1703]	Gets the Base Type
	BaseTypeName [▸ 1703]	Gets the BaseType name
	BitSize [▸ 1727]	Gets the size of the DataType [▸ 1721] in bits. (Inherited from DataType [▸ 1721].)
	ByteSize [▸ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▸ 1721].)
	Category [▸ 1728]	Gets the Data Type category (Inherited from DataType [▸ 1721].)
	Comment [▸ 1729]	Gets the comment. (Inherited from DataType [▸ 1721].)
	FullName [▸ 1729]	Gets the full name of the IDataType [▸ 2475] (Namespace + Name) (Inherited from DataType [▸ 1721].)
	Id [▸ 1729]	Gets the ID of the DataType (Inherited from DataType [▸ 1721].)
	IsBitType [▸ 1730]	Gets a value indicating whether this IDataType [▸ 2475] is a bit mapping Type (Inherited from DataType [▸ 1721].)
	IsByteAligned [▸ 1730]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DataType [▸ 1721].)
	ManagedType [▸ 1705]	Gets the corresponding .NET Type if attached. (Overrides DataType.ManagedType [▸ 1733].)
	Name [▸ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▸ 1721].)
	Namespace [▸ 1734]	Gets the namespace string within the IDataType [▸ 2475] exists. (Inherited from DataType [▸ 1721].)
	Size [▸ 1735]	Gets the Size of the DataType [▸ 1721] in Bytes or bits. (Inherited from DataType [▸ 1721].)
	TypeGuid [▸ 1735]	Gets the Guid of the DataType [▸ 1721] (optional) (Inherited from DataType [▸ 1721].)

Reference

[AliasType Class \[▸ 1699\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

Also see about this

-  [AliasType.IsContainer Property \[▸ 1703\]](#)
-  [DataType.IsContainer Property \[▸ 1731\]](#)
-  [DataType.IsPointer Property \[▸ 1732\]](#)
-  [AliasType.IsPrimitive Property \[▸ 1704\]](#)
-  [DataType.IsPrimitive Property \[▸ 1732\]](#)
-  [DataType.IsReference Property \[▸ 1733\]](#)

6.7.1.1.1 AliasType.BaseType Property

Gets the Base Type

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataType? BaseType { get; }
```

Property Value

Type: [IDataType](#) [[▶ 2475](#)]

Implements

[IAliasType.BaseType](#) [[▶ 2439](#)]

Reference

[AliasType Class](#) [[▶ 1699](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.1.1.2 AliasType.BaseTypeName Property

Gets the BaseType name

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string BaseTypeName { get; }
```

Property Value

Type: String

Implements

[IAliasType.BaseTypeName](#) [[▶ 2440](#)]

Reference

[AliasType Class](#) [[▶ 1699](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.1.1.3 AliasType.IsContainer Property

Gets a value indicating whether this [IDataType](#) [[▶ 2475](#)] is a container type

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public override bool IsContainer { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is container type; otherwise, false.

Implements

[IDataType.IsContainer](#) [[▶ 2480](#)]

[IDataType.IsContainer](#) [[▶ 2480](#)]

Remarks

Container Types are all types that contain SubElements like

- [Array](#) [[▶ 2111](#)]
- [Pointer](#) [[▶ 2111](#)]
- [Union](#) [[▶ 2111](#)]
- [Struct](#) [[▶ 2111](#)]
- [Function](#) [[▶ 2111](#)]
- [FunctionBlock](#) [[▶ 2111](#)]
- [Program](#) [[▶ 2111](#)]

And the [Alias](#) [[▶ 2111](#)] types, if they have a container type as base type.

Reference

[AliasType Class](#) [[▶ 1699](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

[IDataType.Category](#) [[▶ 2478](#)]

6.7.1.1.4 AliasType.IsPrimitive Property

Gets a value indicating whether this [IDataType](#) [[▶ 2475](#)] is primitive

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public override bool IsPrimitive { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is primitive; otherwise, false.

Implements

[IDataType.IsPrimitive \[▸ 2481\]](#)

[IDataType.IsPrimitive \[▸ 2481\]](#)

Reference

[AliasType Class \[▸ 1699\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.1.1.5 AliasType.ManagedType Property

Gets the corresponding .NET Type if attached.

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override Type? ManagedType { get; }
```

Property Value

Type: Type
Dot net type.

Reference





[AliasType Class \[▸ 1699\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.1.2 AliasType Methods

The [AliasType \[▸ 1699\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString [▸ 1706]	Returns a String that represents this instance. (Overrides IDataType.ToString [▸ 1738] .)

Extension Methods

	Name	Description
	IsArrayOfPrimitives [▸ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by IDataTypeExtension [▸ 3021] .)

	Name	Description
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021] .)
	IsContainer. [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021] .)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021] .)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021] .)
	IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021] .)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021] .)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021] .)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021] .)
	ResolvableAsPrimitive. [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021] .)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021] .)
	Resolve. [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021] .)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021] .)

Reference

[AliasType Class \[▶ 1699\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.1.2.1 AliasType.ToString Method

Returns a String that represents this instance.

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```

Return Value

Type: String
A String that represents this instance.

Reference

[AliasType Class \[▸ 1699\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.2 ArrayType Class

Represents an Array DataType

DataType class

Inheritance Hierarchy

System.Object
[TwinCAT.Ads.TypeSystem.DataType \[▸ 1721\]](#)
[TwinCAT.Ads.TypeSystem.ArrayType](#)

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229









Syntax
















C#

```
public class ArrayType : DataType, IArrayType,
    IDataType, IBitSize
```









The ArrayType type exposes the following members.

Properties

	Name	Description
	Attributes [▸ 1727]	Gets the attributes of the IDataType [▸ 2475] (Inherited from DataType [▸ 1721] .)
	BitSize [▸ 1727]	Gets the size of the DataType [▸ 1721] in bits. (Inherited from DataType [▸ 1721] .)
	ByteSize [▸ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▸ 1721] .)
	Category [▸ 1728]	Gets the Data Type category (Inherited from DataType [▸ 1721] .)
	Comment [▸ 1729]	Gets the comment. (Inherited from DataType [▸ 1721] .)
	DimensionCount [▸ 1711]	Gets the dimension count.
	Dimensions [▸ 1711]	Gets the dimensions as read only collection.
	ElementCount [▸ 1712]	Gets the element count.

	Name	Description
	ElementSize [▶ 1712]	Gets the byte-size of a single element of the array
	ElementType [▶ 1712]	Gets the type of the contained elements.
	ElementTypeName [▶ 1713]	Gets the name of the element type.
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721] .)
	Id [▶ 1729]	Gets the ID of the DataType (Inherited from DataType [▶ 1721] .)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721] .)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DataType [▶ 1721] .)
	IsJagged [▶ 1713]	Gets a value indicating whether this instance is jagged.
	IsOversampled [▶ 1714]	Gets a value indicating whether this array instance describes an oversampling type.
	JaggedLevel [▶ 1715]	Gets the jagged level (Non-Jagged Array have level 1)
	ManagedType [▶ 1715]	Gets the corresponding .NET Type if attached. (Overrides DataType.ManagedType [▶ 1733] .)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721] .)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721] .)
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721] .)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721] .)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetDimensions [▶ 1717]	Sets the dimensions.
	SetElementType [▶ 1717]	Sets the type of the element.
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721] .)

Extension Methods


	Name	Description
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

Also see about this

- [IDataType.IsContainer Property](#) [[▶ 1731](#)]
- [IDataType.IsPointer Property](#) [[▶ 1732](#)]
- [IArrayType.IsPrimitive Property](#) [[▶ 1714](#)]
- [IDataType.IsPrimitive Property](#) [[▶ 1732](#)]


 [DataType.IsReference Property \[▶ 1733\]](#)

6.7.2.1 ArrayType Properties

The [ArrayType \[▶ 1707\]](#) type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	DimensionCount [▶ 1711]	Gets the dimension count.
	Dimensions [▶ 1711]	Gets the dimensions as read only collection.
	ElementCount [▶ 1712]	Gets the element count.
	ElementSize [▶ 1712]	Gets the byte-size of a single element of the array
	ElementType [▶ 1712]	Gets the type of the contained elements.
	ElementTypeName [▶ 1713]	Gets the name of the element type.
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the Data Type (Inherited from DataType [▶ 1721].)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DataType [▶ 1721].)
	IsJagged [▶ 1713]	Gets a value indicating whether this instance is jagged.
	IsOversampled [▶ 1714]	Gets a value indicating whether this array instance describes an oversampling type.
	JaggedLevel [▶ 1715]	Gets the jagged level (Non-Jagged Array have level 1)
	ManagedType [▶ 1715]	Gets the corresponding .NET Type if attached. (Overrides DataType.ManagedType [▶ 1733].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)

	Name	Description
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Reference

[ArrayType Class](#) [[▶ 1707](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

Also see about this

- [DataType.IsContainer Property](#) [[▶ 1731](#)]
- [DataType.IsPointer Property](#) [[▶ 1732](#)]
- [ArrayType.IsPrimitive Property](#) [[▶ 1714](#)]
- [DataType.IsPrimitive Property](#) [[▶ 1732](#)]
- [DataType.IsReference Property](#) [[▶ 1733](#)]

6.7.2.1.1 **ArrayType.DimensionCount Property**

Gets the dimension count.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int DimensionCount { get; }
```

Property Value

Type: Int32

The dimension count.

Reference

[ArrayType Class](#) [[▶ 1707](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.2.1.2 **ArrayType.Dimensions Property**

Gets the dimensions as read only collection.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDimensionCollection Dimensions { get; }
```

Property Value

Type: [IDimensionCollection](#) [[▶ 2492](#)]

The dimensions.

Implements

[IArrayType.Dimensions](#) [[▶ 2463](#)]

Reference

[ArrayType Class](#) [[▶ 1707](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.2.1.3 ArrayType.ElementCount Property

Gets the element count.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ElementCount { get; }
```

Property Value

Type: Int32

The element count.

Reference

[ArrayType Class](#) [[▶ 1707](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.2.1.4 ArrayType.ElementSize Property

Gets the byte-size of a single element of the array

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ElementSize { get; }
```

Property Value

Type: Int32

The size of the element.

Reference

[ArrayType Class](#) [[▶ 1707](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.2.1.5 ArrayType.ElementType Property

Gets the type of the contained elements.

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataType? ElementType { get; }
```

Property Value

Type: [IDataType](#) [► 2475]

The type of the element.

Implements

[IArrayType.ElementType](#) [► 2463]

Reference

[ArrayType Class](#) [► 1707]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.2.1.6 **ArrayType.ElementTypeName Property**

Gets the name of the element type.

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string ElementTypeName { get; }
```

Property Value

Type: String

The name of the element type.

Implements

[IArrayType.ElementTypeName](#) [► 2463]

Reference

[ArrayType Class](#) [► 1707]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.2.1.7 **ArrayType.IsJagged Property**

Gets a value indicating whether this instance is jagged.

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsJagged { get; }
```

Property Value

Type: Boolean
true if this instance is jagged; otherwise, false.

Implements

[IArrayType.IsJagged](#) [[▶ 2464](#)]

Reference

[ArrayType Class](#) [[▶ 1707](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.2.1.8 ArrayType.IsOversampled Property

Gets a value indicating whether this array instance describes an oversampling type.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsOversampled { get; }
```

Property Value

Type: Boolean
true if this instance is oversampling; otherwise, false.

Reference

[ArrayType Class](#) [[▶ 1707](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.2.1.9 ArrayType.IsPrimitive Property

Gets a value indicating whether this [IDataType](#) [[▶ 2475](#)] is primitive

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public override bool IsPrimitive { get; }
```

Property Value

Type: [Boolean](#)
true if this instance is primitive; otherwise, false.

Implements

[IDataType.IsPrimitive \[▸ 2481\]](#)

[IDataType.IsPrimitive \[▸ 2481\]](#)

Reference

[ArrayType Class \[▸ 1707\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.2.1.10 ArrayType.JaggedLevel Property

Gets the jagged level (Non-Jagged Array have level 1)

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int JaggedLevel { get; }
```

Property Value

Type: Int32

The jagged level.

Implements

[IArrayType.JaggedLevel \[▸ 2464\]](#)

Reference

[ArrayType Class \[▸ 1707\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.2.1.11 ArrayType.ManagedType Property

Gets the corresponding .NET Type if attached.

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override Type? ManagedType { get; }
```

Property Value

Type: Type

Dot net type.

Reference









[ArrayType Class \[▸ 1707\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.2.2 ArrayType Methods

The [ArrayType \[► 1707\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object .)
	SetDimensions [► 1717]	Sets the dimensions.
	SetElementType [► 1717]	Sets the type of the element.
	ToString [► 1738]	Returns a String that represents this instance. (Inherited from DataType [► 1721] .)

Extension Methods

	Name	Description
	IsArrayOfPrimitives [► 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [► 3021] .)
	IsArrayOfPrimitives(Boolean) [► 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [► 3021] .)
	IsContainer [► 3027]	Overloaded. Gets a value indicating whether this IDataType [► 2475] is a container type (Defined by DataTypeExtension [► 3021] .)
	IsContainer(Boolean) [► 3028]	Overloaded. Gets a value indicating whether this IDataType [► 2475] is a container type (Defined by DataTypeExtension [► 3021] .)
	IsPointer [► 3031]	Gets a value indicating whether this IDataType [► 2475] is a pointer type (Defined by DataTypeExtension [► 3021] .)
	IsPrimitive [► 3033]	Overloaded. Gets a value indicating whether this IDataType [► 2475] is primitive (Defined by DataTypeExtension [► 3021] .)
	IsPrimitive(Boolean) [► 3034]	Overloaded. Gets a value indicating whether this IDataType [► 2475] is primitive (Defined by DataTypeExtension [► 3021] .)
	IsReference [► 3036]	Gets a value indicating whether this IDataType [► 2475] is a reference type (Defined by DataTypeExtension [► 3021] .)

	Name	Description
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve . [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[ArrayType Class](#) [[▶ 1707](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.2.2.1 ArrayType.SetDimensions Method

Sets the dimensions.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected void SetDimensions(
    DimensionCollection dims
)
```

Parameters

dims	Type: TwinCAT.TypeSystem.DimensionCollection [▶ 2133] The dims.
------	--

Reference

[ArrayType Class](#) [[▶ 1707](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.2.2.2 ArrayType.SetElementType Method

Sets the type of the element.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected void SetElementType(
    DataType elementType
)
```

Parameters

elementType	Type: TwinCAT.Ads.TypeSystem.DataType [▸ 1721] Type of the element.
-------------	--

Reference

[ArrayType Class](#) [[▸ 1707](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▸ 1698](#)]

6.7.3 BitMappingType Class

Helper Data Type to implement Bit mapping types.

Inheritance Hierarchy

System.Object

[TwinCAT.Ads.TypeSystem.DataType](#) [[▸ 1721](#)]

[TwinCAT.Ads.TypeSystem.BitMappingType](#)

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▸ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229


Syntax

C#







```
public sealed class BitMappingType : DataType
```









The BitMappingType type exposes the following members.

Constructors





	Name	Description
	BitMappingType [▸ 1719]	Initializes a new instance of the BitMappingType class.

Properties

	Name	Description
	Attributes [▸ 1727]	Gets the attributes of the IDataType [▸ 2475] (Inherited from DataType [▸ 1721].)
	BitSize [▸ 1727]	Gets the size of the DataType [▸ 1721] in bits. (Inherited from DataType [▸ 1721].)
	ByteSize [▸ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▸ 1721].)
	Category [▸ 1728]	Gets the Data Type category (Inherited from DataType [▸ 1721].)
	Comment [▸ 1729]	Gets the comment. (Inherited from DataType [▸ 1721].)
	FullName [▸ 1729]	Gets the full name of the IDataType [▸ 2475] (Namespace + Name) (Inherited from DataType [▸ 1721].)

	Name	Description
	Id [▶ 1729]	Gets the ID of the DataType (Inherited from DataType [▶ 1721].)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DataType [▶ 1721].)
	ManagedType [▶ 1733]	Gets the corresponding .NET Type if attached. (Inherited from DataType [▶ 1721].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)





Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721].)

Reference

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

Also see about this

-  [DataType.IsContainer Property \[▶ 1731\]](#)
-  [DataType.IsPointer Property \[▶ 1732\]](#)
-  [DataType.IsPrimitive Property \[▶ 1732\]](#)
-  [DataType.IsReference Property \[▶ 1733\]](#)

6.7.3.1 BitMappingType Constructor

Initializes a new instance of the [BitMappingType \[▶ 1718\]](#) class.

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public BitMappingType(
    string name,
    int bitSize,
    Type dotnetType
)
```

Parameters

name	Type: System.String The name.
bitSize	Type: System.Int32 The size of the type in bits.
dotnetType	Type: System.Type Type of the dotnet.

Reference















[BitMappingType Class \[▶ 1718\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.3.2 BitMappingType Properties

The [BitMappingType \[▶ 1718\]](#) type exposes the following members.

Properties





	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the DataType [▶ 1721]. (Inherited from DataType [▶ 1721].)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DataType [▶ 1721].)
	ManagedType [▶ 1733]	Gets the corresponding .NET Type if attached. (Inherited from DataType [▶ 1721].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Reference

[BitMappingType Class \[▶ 1718\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)





Also see about this

-  [DataType.IsContainer Property \[▶ 1731\]](#)
-  [DataType.IsPointer Property \[▶ 1732\]](#)
-  [DataType.IsPrimitive Property \[▶ 1732\]](#)
-  [DataType.IsReference Property \[▶ 1733\]](#)

6.7.3.3 BitMappingType Methods

The [BitMappingType \[▶ 1718\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721] .)

Reference

[BitMappingType Class \[▶ 1718\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.4 DataType Class

DataType class

Inheritance Hierarchy

System.Object
 TwinCAT.Ads.TypeSystem.DataType
[More... \[▶ 1723\]](#)

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229


Syntax



C#

```
public class DataType : IDataType, IBitSize
```















The DataType type exposes the following members.

Constructors







	Name	Description
	DataType(DataType) [▶ 1724]	Initializes a new instance of the DataType class (copy Constructor)

	Name	Description
	<code>DataType(DataType Category)</code> [1725]	Initializes a new instance of the <code>DataType</code> class.
	<code>DataType(String, DataTypeCategory, Type)</code> [1725]	Initializes a new instance of the <code>DataType</code> class.

Properties

	Name	Description
	<code>Attributes</code> [1727]	Gets the attributes of the <code>IDataType</code> [2475]
	<code>BitSize</code> [1727]	Gets the size of the <code>DataType</code> in bits.
	<code>ByteSize</code> [1728]	Gets the (aligned) size of of the <code>Type/Instance</code> in Bytes
	<code>Category</code> [1728]	Gets the Data Type category
	<code>Comment</code> [1729]	Gets the comment.
	<code>FullName</code> [1729]	Gets the full name of the <code>IDataType</code> [2475] (Namespace + Name)
	<code>Id</code> [1729]	Gets the ID of the <code>DataType</code>
	<code>IsBitType</code> [1730]	Gets a value indicating whether this <code>IDataType</code> [2475] is a bit mapping Type
	<code>IsByteAligned</code> [1730]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$)
	<code>ManagedType</code> [1733]	Gets the corresponding .NET Type if attached.
	<code>Name</code> [1734]	Gets the name of the Data Type (without namespace)
	<code>Namespace</code> [1734]	Gets the namespace string within the <code>IDataType</code> [2475] exists.
	<code>Size</code> [1735]	Gets the Size of the <code>DataType</code> in Bytes or bits.
	<code>TypeGuid</code> [1735]	Gets the Guid of the <code>DataType</code> (optional)

Methods

	Name	Description
	<code>Equals</code>	Determines whether the specified object is equal to the current object. (Inherited from <code>Object</code> .)
	<code>Finalize</code>	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from <code>Object</code> .)
	<code>GetHashCode</code>	Serves as the default hash function. (Inherited from <code>Object</code> .)
	<code>GetType</code>	Gets the Type of the current instance. (Inherited from <code>Object</code> .)
	<code>MemberwiseClone</code>	Creates a shallow copy of the current <code>Object</code> . (Inherited from <code>Object</code> .)
	<code>ToString</code> [1738]	Returns a <code>String</code> that represents this instance. (Overrides <code>Object.ToString</code> .)

Extension Methods

	Name	Description
	IsArrayOfPrimitives . [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer . [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive . [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive . [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve . [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

Inheritance Hierarchy

System.Object
 TwinCAT.Ads.TypeSystem.DataType
 [TwinCAT.Ads.TypeSystem.AliasType](#) [▶ 1699]
 [TwinCAT.Ads.TypeSystem.ArrayType](#) [▶ 1707]

[TwinCAT.Ads.TypeSystem.BitMappingType](#) [[▶ 1718](#)]
[TwinCAT.Ads.TypeSystem.EnumType.T.](#) [[▶ 1738](#)]
[TwinCAT.Ads.TypeSystem.InterfaceType](#) [[▶ 1957](#)]
[TwinCAT.Ads.TypeSystem.PointerType](#) [[▶ 1790](#)]
[TwinCAT.Ads.TypeSystem.PrimitiveType](#) [[▶ 1796](#)]
[TwinCAT.Ads.TypeSystem.ReferenceType](#) [[▶ 1804](#)]
[TwinCAT.Ads.TypeSystem.StringType](#) [[▶ 1840](#)]
[TwinCAT.Ads.TypeSystem.StructType](#) [[▶ 1846](#)]
[TwinCAT.Ads.TypeSystem.SubRangeType.T.](#) [[▶ 1856](#)]
[TwinCAT.Ads.TypeSystem.UnionType](#) [[▶ 1933](#)]
[TwinCAT.Ads.TypeSystem.WStringType](#) [[▶ 1938](#)]

Also see about this

- [DataType.IsContainer Property](#) [[▶ 1731](#)]
- [DataType.IsPointer Property](#) [[▶ 1732](#)]
- [DataType.IsPrimitive Property](#) [[▶ 1732](#)]
- [DataType.IsReference Property](#) [[▶ 1733](#)]
- [DataType.IsPointerType Method](#) [[▶ 1737](#)]
- [DataType.IsReferenceType Method](#) [[▶ 1737](#)]

6.7.4.1 DataType Constructor

Overload List

	Name	Description
	DataType(DataType) [▶ 1724]	Initializes a new instance of the DataType [▶ 1721] class (copy Constructor)
	DataType(DataType Category) [▶ 1725]	Initializes a new instance of the DataType [▶ 1721] class.
	DataType(String, DataTypeCategory, Type) [▶ 1725]	Initializes a new instance of the DataType [▶ 1721] class.

Reference

[DataType Class](#) [[▶ 1721](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.4.1.1 DataType Constructor (DataType)

Initializes a new instance of the [DataType](#) [[▶ 1721](#)] class (copy Constructor)

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected DataType(
    DataType copy
)
```

Parameters

copy	Type: TwinCAT.Ads.TypeSystem.DataType [▸ 1721] The copy.
------	---

Reference

[DataType Class \[▸ 1721\]](#)

[DataType Overload \[▸ 1724\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.4.1.2 DataType Constructor (DataTypeCategory)

Initializes a new instance of the [DataType \[▸ 1721\]](#) class.

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected DataType(
    DataTypeCategory cat
)
```

Parameters

cat	Type: TwinCAT.TypeSystem.DataTypeCategory [▸ 2111] The cat.
-----	--

Reference

[DataType Class \[▸ 1721\]](#)

[DataType Overload \[▸ 1724\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.4.1.3 DataType Constructor (String, DataTypeCategory, Type)

Initializes a new instance of the [DataType \[▸ 1721\]](#) class.

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected DataType(
    string name,
    DataTypeCategory cat,
    Type dotnetType
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

cat	Type: TwinCAT.TypeSystem.DataTypeCategory [▸ 2111] The cat.
dotnetType	Type: System.Type Type of the dotnet.

Reference

[DataType Class \[▸ 1721\]](#)















[DataType Overload \[▸ 1724\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.4.2 DataType Properties

The [DataType \[▸ 1721\]](#) type exposes the following members.

Properties

	Name	Description
	Attributes [▸ 1727]	Gets the attributes of the IDataType [▸ 2475]
	BitSize [▸ 1727]	Gets the size of the DataType [▸ 1721] in bits.
	ByteSize [▸ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes
	Category [▸ 1728]	Gets the Data Type category
	Comment [▸ 1729]	Gets the comment.
	FullName [▸ 1729]	Gets the full name of the IDataType [▸ 2475] (Namespace + Name)
	Id [▸ 1729]	Gets the ID of the DataType
	IsBitType [▸ 1730]	Gets a value indicating whether this IDataType [▸ 2475] is a bit mapping Type
	IsByteAligned [▸ 1730]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$)
	ManagedType [▸ 1733]	Gets the corresponding .NET Type if attached.
	Name [▸ 1734]	Gets the name of the Data Type (without namespace)
	Namespace [▸ 1734]	Gets the namespace string within the IDataType [▸ 2475] exists.
	Size [▸ 1735]	Gets the Size of the DataType [▸ 1721] in Bytes or bits.
	TypeGuid [▸ 1735]	Gets the Guid of the DataType [▸ 1721] (optional)

Reference

[DataType Class \[▸ 1721\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

Also see about this

 [DataType.IsContainer Property \[▸ 1731\]](#)

- [DataType.IsPointer Property \[▶ 1732\]](#)
- [DataType.IsPrimitive Property \[▶ 1732\]](#)
- [DataType.IsReference Property \[▶ 1733\]](#)

6.7.4.2.1 **DataType.Attributes Property**

Gets the attributes of the [IDataType](#) [▶ 2475]

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ITypeAttributeCollection Attributes { get; }
```

Property Value

Type: [ITypeAttributeCollection](#) [▶ 2726]

The attributes.

Implements

[IDataType.Attributes](#) [▶ 2478]

Exceptions

Exception	Condition
NotImplementedException	

Reference

[DataType Class](#) [▶ 1721]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.4.2.2 **DataType.BitSize Property**

Gets the size of the [DataType](#) [▶ 1721] in bits.

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int BitSize { get; }
```

Property Value

Type: Int32

The size of the bit.

Implements

[IBitSize.BitSize](#) [▶ 2473]

Reference

[DataType Class \[► 1721\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.4.2.3 **DataType.ByteSize Property**

Gets the (aligned) size of of the Type/Instance in Bytes

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ByteSize { get; }
```

Property Value

Type: Int32

The size of the byte.

Implements

[IBitSize.ByteSize \[► 2474\]](#)

Reference

[DataType Class \[► 1721\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.4.2.4 **DataType.Category Property**

Gets the Data Type category

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DataTypeCategory Category { get; protected set; }
```

Property Value

Type: [DataTypeCategory \[► 2111\]](#)

The category.

Implements

[IDataType.Category \[► 2478\]](#)

Reference

[DataType Class \[► 1721\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.4.2.5 DataType.Comment Property

Gets the comment.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Comment { get; }
```

Property Value

Type: String

The comment.

Implements

[IDataType.Comment](#) [[▶ 2479](#)]

Reference

[DataType Class](#) [[▶ 1721](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.4.2.6 DataType.FullName Property

Gets the full name of the [IDataType](#) [[▶ 2475](#)] (Namespace + Name)

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string FullName { get; }
```

Property Value

Type: String

The full name.

Implements

[IDataType.FullName](#) [[▶ 2479](#)]

Reference

[DataType Class](#) [[▶ 1721](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.4.2.7 DataType.Id Property

Gets the ID of the DataType

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Id { get; }
```

Property Value

Type: Int32

The id.

Implements

[IDataType.Id](#) [[▶ 2480](#)]

Reference

[DataType Class](#) [[▶ 1721](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.4.2.8 **DataType.IsBitType Property**

Gets a value indicating whether this [IDataType](#) [[▶ 2475](#)] is a bit mapping Type

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsBitType { get; }
```

Property Value

Type: Boolean

true if this instance is bit mapping subtype; otherwise, false.

Implements

[IBitSize.IsBitType](#) [[▶ 2474](#)]

Reference

[DataType Class](#) [[▶ 1721](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.4.2.9 **DataType.IsByteAligned Property**

Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0)

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsByteAligned { get; }
```

Property Value

Type: Boolean
true if this instance is byte aligned; otherwise, false.

Implements

[IBitSize.IsByteAligned](#) [[▶ 2475](#)]

Reference

[DataType Class](#) [[▶ 1721](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.4.2.10 **DataType.IsContainer Property**

Gets a value indicating whether this [IDataType](#) [[▶ 2475](#)] is a container type

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public virtual bool IsContainer { get; }
```

Property Value

Type: [Boolean](#)
true if this instance is container type; otherwise, false.

Implements

[IDataType.IsContainer](#) [[▶ 2480](#)]

Remarks

Container Types are all types that contain SubElements like

- [Array](#) [[▶ 2111](#)]
- [Pointer](#) [[▶ 2111](#)]
- [Union](#) [[▶ 2111](#)]
- [Struct](#) [[▶ 2111](#)]
- [Function](#) [[▶ 2111](#)]
- [FunctionBlock](#) [[▶ 2111](#)]
- [Program](#) [[▶ 2111](#)]

and the [Alias](#) [[▶ 2111](#)] and [Reference](#) [[▶ 2111](#)] types, if they have a container type as base type.

Reference

[DataType Class](#) [[▶ 1721](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

[IDataType.Category](#) [[▶ 2478](#)]

6.7.4.2.11 DataType.IsPointer Property

Gets a value indicating whether this [IDataType](#) [[▶ 2475](#)] is a pointer type

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public virtual bool IsPointer { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is pointer type; otherwise, false.

Implements

[IDataType.IsPointer](#) [[▶ 2481](#)]

Remarks

Pointer types can be dereferenced with the '^' operator.

Reference

[DataType Class](#) [[▶ 1721](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

[IDataType.Category](#) [[▶ 2478](#)]

6.7.4.2.12 DataType.IsPrimitive Property

Gets a value indicating whether this [IDataType](#) [[▶ 2475](#)] is primitive

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public virtual bool IsPrimitive { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is primitive; otherwise, false.

Implements

[IDataType.IsPrimitive](#) [[▶ 2481](#)]

Reference

[DataType Class](#) [[▶ 1721](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.4.2.13 **DataType.IsReference Property**

Gets a value indicating whether this [IDataType](#) [[▶ 2475](#)] is a reference type

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public virtual bool IsReference { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is container type; otherwise, false.

Implements

[IDataType.IsReference](#) [[▶ 2482](#)]

Remarks

Reference types can be dereferenced.

Reference

[DataType Class](#) [[▶ 1721](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

[IDataType.Category](#) [[▶ 2478](#)]

6.7.4.2.14 **DataType.ManagedType Property**

Gets the corresponding .NET Type if attached.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public virtual Type? ManagedType { get; protected set; }
```

Property Value

Type: [Type](#)

Dot net type.

Reference

[DataType Class \[► 1721\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.4.2.15 DataType.Name Property

Gets the name of the Data Type (without namespace)

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Name { get; protected set; }
```

Property Value

Type: String
The name.

Implements

[IDataType.Name \[► 2482\]](#)

Reference

[DataType Class \[► 1721\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.4.2.16 DataType.Namespace Property

Gets the namespace string within the [IDataType \[► 2475\]](#) exists.

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Namespace { get; protected set; }
```

Property Value

Type: String
The namespace.

Implements

[IDataType.Namespace \[► 2482\]](#)

Reference

[DataType Class \[► 1721\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.4.2.17 DataType.Size Property

Gets the Size of the [DataType](#) [[▶ 1721](#)] in Bytes or bits.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Size { get; protected set; }
```

Property Value

Type: Int32

The size.

Implements

[IBitSize.Size](#) [[▶ 2475](#)]

Reference

[DataType Class](#) [[▶ 1721](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.4.2.18 DataType.TypeGuid Property

Gets the Guid of the [DataType](#) [[▶ 1721](#)] (optional)

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Guid? TypeGuid { get; }
```

Property Value

Type: Nullable.Guid.

The type unique identifier.

Reference


[DataType Class](#) [[▶ 1721](#)]






[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.4.3 DataType Methods

The [DataType](#) [[▶ 1721](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)

	Name	Description
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString [▶ 1738]	Returns a String that represents this instance. (Overrides Object.ToString..)

Extension Methods

	Name	Description
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)

	Name	Description
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[DataType Class](#) [▶ 1721]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

Also see about this

- [DataType.IsPointerType Method](#) [▶ 1737]
- [DataType.IsReferenceType Method](#) [▶ 1737]

6.7.4.3.1 DataType.IsPointerType Method

Determines whether the specified category is a pointer type.

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public static bool IsPointerType(  
    DataTypeCategory cat  
)
```

Parameters

cat Type: [TwinCAT.TypeSystem.DataTypeCategory](#) [▶ 2111]
The data type category.

Return Value

Type: [Boolean](#)
true if [is pointer type] [the specified cat]; otherwise, false.

Reference

[DataType Class](#) [▶ 1721]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.4.3.2 DataType.IsReferenceType Method

Determines whether the specified category is a reference type.

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public static bool IsReferenceType(  
    DataTypeCategory cat  
)
```

Parameters

cat Type: [TwinCAT.TypeSystem.DataTypeCategory](#) [▶ 2111]
The data type category.

Return Value

Type: [Boolean](#)
true if [is reference type] [the specified cat]; otherwise, false.

Reference

[DataType Class](#) [▶ 1721]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.4.3.3 **DataType.ToString Method**

Returns a String that represents this instance.

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```

Return Value

Type: [String](#)
A String that represents this instance.

Reference

[DataType Class](#) [▶ 1721]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.5 **EnumType.T. Class**

Enum [DataType](#) [▶ 1721].

Inheritance Hierarchy

System.Object

[TwinCAT.Ads.TypeSystem.DataType](#) [▶ 1721]

 TwinCAT.Ads.TypeSystem.EnumType.T.

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public sealed class EnumType<T> : DataType,  
    IEnumType<T>, IAliasType, IDataType, IBitSize, IEnumType  
where T : struct, new(), IConvertible
```

Type Parameters





T









The EnumType.T. type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BaseType [▶ 1742]	Gets the Base Type
	BaseTypeName [▶ 1742]	Gets the BaseType name
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	EnumValues [▶ 1743]	Enumeration specification (if enum)
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the Data Type (Inherited from DataType [▶ 1721].)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned ($BitSize \% 8 == 0$) (Inherited from DataType [▶ 1721].)
	ManagedType [▶ 1733]	Gets the corresponding .NET Type if attached. (Inherited from DataType [▶ 1721].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Methods

	Name	Description
	Contains [▶ 1745]	Determines whether the enum values contains the specified name
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetNames [▶ 1746]	Gets the filed names of the IEnumType.T. [▶ 2516]

	Name	Description
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	GetValues [▶ 1746]	Gets the values of the IEnumType.T. [▶ 2516]
	Parse [▶ 1747]	Parses a name of the IEnumType.T. [▶ 2516] and returns the value (as base type)
	ToString. [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721].)
	ToString(T) [▶ 1748]	Returns a String that represents this instance.
	ToString(IConvertible) [▶ 1747]	Returns a String that represents this instance.
	TryParse(String, T.) [▶ 1750]	Tries to parse the Enum Value
	TryParse(String, IEnumValue.) [▶ 1749]	Tries to parse the Enum Value

Extension Methods





	Name	Description
	IsArrayOfPrimitives. [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer. [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive. [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)

	Name	Description
	ResolveableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021] .)
	Resolve. [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021] .)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021] .)

Reference

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)













Also see about this






-  [DataType.IsContainer Property \[▶ 1731\]](#)
-  [DataType.IsPointer Property \[▶ 1732\]](#)
-  [DataType.IsPrimitive Property \[▶ 1732\]](#)
-  [DataType.IsReference Property \[▶ 1733\]](#)

6.7.5.1 EnumType.T. Properties

The [EnumType.T. \[▶ 1738\]](#) generic type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721] .)
	BaseType [▶ 1742]	Gets the Base Type
	BaseTypeName [▶ 1742]	Gets the BaseType name
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721] .)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721] .)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721] .)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721] .)
	EnumValues [▶ 1743]	Enumeration specification (if enum)
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721] .)
	Id [▶ 1729]	Gets the ID of the DataType (Inherited from DataType [▶ 1721] .)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721] .)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DataType [▶ 1721] .)





	Name	Description
	ManagedType [▶ 1733]	Gets the corresponding .NET Type if attached. (Inherited from DataType [▶ 1721].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Reference

[EnumType.T. Class](#) [▶ 1738]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

Also see about this

-  [DataType.IsContainer Property](#) [▶ 1731]
-  [DataType.IsPointer Property](#) [▶ 1732]
-  [DataType.IsPrimitive Property](#) [▶ 1732]
-  [DataType.IsReference Property](#) [▶ 1733]

6.7.5.1.1 EnumType.T..BaseType Property

Gets the Base Type

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDatatype? BaseType { get; }
```

Property Value

Type: [IDataType](#) [▶ 2475]

The type of the base.

Implements

[IAliasType.BaseType](#) [▶ 2439]

Reference

[EnumType.T. Class](#) [▶ 1738]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.5.1.2 EnumType.T..BaseTypeName Property

Gets the BaseType name

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string BaseTypeName { get; }
```

Property Value

Type: String

The name of the base type.

Implements

[IAliasType.BaseTypeName](#) [► 2440]

Reference

[EnumType.T. Class](#) [► 1738]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.5.1.3 EnumType.T..EnumValues Property

Enumeration specification (if enum)

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IEnumerable EnumValues { get; }
```

Property Value

Type: [IEnumerable](#) [► 2527]

The enum specification.

Implements

[EnumType.T..EnumValues](#) [► 2519]

Reference













[EnumType.T. Class](#) [► 1738]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.5.2 EnumType.T. Methods

The [EnumType.T.](#) [► 1738] generic type exposes the following members.

Methods

	Name	Description
	Contains [▶ 1745]	Determines whether the enum values contains the specified name
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetNames [▶ 1746]	Gets the filed names of the IEnumType.T. [▶ 2516]
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	GetValues [▶ 1746]	Gets the values of the IEnumType.T. [▶ 2516]
	Parse [▶ 1747]	Parses a name of the IEnumType.T. [▶ 2516] and returns the value (as base type)
	ToString. [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721].)
	ToString(T) [▶ 1748]	Returns a String that represents this instance.
	ToString(IConvertible) [▶ 1747]	Returns a String that represents this instance.
	TryParse(String, T.) [▶ 1750]	Tries to parse the Enum Value
	TryParse(String, IEnumValue.) [▶ 1749]	Tries to parse the Enum Value

Extension Methods

	Name	Description
	IsArrayOfPrimitives. [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer. [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)

	Name	Description
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[EnumType.T. Class](#) [[▶ 1738](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.5.2.1 EnumType.T..Contains Method

Determines whether the enum values contains the specified name

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(
    string name
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: Boolean
true if contains the value, otherwise, false.

Implements

[IEnumType.T..Contains\(String\)](#) [[▶ 2521](#)]

[IEnumType.Contains\(String\)](#) [[▶ 2512](#)]

Reference

[EnumType.T. Class \[► 1738\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.5.2.2 EnumType.T..GetNames Method

Gets the filed names of the [IEnumType.T. \[► 2516\]](#)

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string[] GetNames()
```

Return Value

Type: [.String](#).
[System.String\[\]](#).

Implements

[IEnumType.T..GetNames. \[► 2521\]](#)

[IEnumType.GetNames. \[► 2513\]](#)

Reference

[EnumType.T. Class \[► 1738\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.5.2.3 EnumType.T..GetValues Method

Gets the values of the [IEnumType.T. \[► 2516\]](#)

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T[] GetValues()
```

Return Value

Type: [.T \[► 1738\]](#).
[T\[\]](#).

Implements

[IEnumType.T..GetValues. \[► 2522\]](#)

Reference

[EnumType.T. Class \[► 1738\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.5.2.4 EnumType.T..Parse Method

Parses a name of the [IEnumType.T. \[▶ 2516\]](#) and returns the value (as base type)

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T Parse(
    string strValue
)
```

Parameters

strValue	Type: System.String Enum Value as string.
----------	--

Return Value

Type: [T \[▶ 1738\]](#)

T.

Implements

[IEnumType.T..Parse\(String\) \[▶ 2522\]](#)




Reference

[EnumType.T. Class \[▶ 1738\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.5.2.5 EnumType.T..ToString Method

Overload List

	Name	Description
	ToString. [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721].)
	ToString(T) [▶ 1748]	Returns a String that represents this instance.
	ToString(IConvertible) [▶ 1747]	Returns a String that represents this instance.

Reference

[EnumType.T. Class \[▶ 1738\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.5.2.5.1 EnumType.T..ToString Method (IConvertible)

Returns a String that represents this instance.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string ToString(  
    IConvertible val  
)
```

Parameters

val	Type: System.IConvertible The value.
-----	---

Return Value

Type: String

A String that represents this instance.

Implements

[IEnumType.ToString\(IConvertible\)](#) [[▶ 2514](#)]

Reference

[EnumType.T.Class](#) [[▶ 1738](#)]

[ToString Overload](#) [[▶ 1747](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.5.2.5.2 EnumType.T..ToString Method (T)

Returns a String that represents this instance.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string ToString(  
    T val  
)
```

Parameters

val	Type: T [▶ 1738] The value.
-----	--

Return Value

Type: String

A String that represents this instance.

Implements

[IEnumType.T..ToString\(T\)](#) [[▶ 2523](#)]

Exceptions

Exception	Condition
ArgumentException	val

Reference



[EnumType.T. Class \[▸ 1738\]](#)

[ToString Overload \[▸ 1747\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.5.2.6 EnumType.T..TryParse Method

Overload List

	Name	Description
	TryParse(String, T.) [▸ 1750]	Tries to parse the Enum Value
	TryParse(String, IEnumValue.) [▸ 1749]	Tries to parse the Enum Value

Reference

[EnumType.T. Class \[▸ 1738\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.5.2.6.1 EnumType.T..TryParse Method (String, IEnumValue.)

Tries to parse the Enum Value

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryParse(
    string strValue,
    out IEnumValue?? value
)
```

Parameters

strValue	Type: System.String Enum value (in string representation).
value	Type: TwinCAT.TypeSystem.IEnumValue [▸ 2524] . The value.

Return Value

Type: Boolean

true if the [EnumType.T. \[▸ 1738\]](#) could be parsed, false otherwise.

Implements

[IEnumType.TryParse\(String, IEnumValue.\)](#) [[▶ 2515](#)]

Reference

[EnumType.T. Class](#) [[▶ 1738](#)]

[TryParse Overload](#) [[▶ 1749](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.5.2.6.2 EnumType.T..TryParse Method (String, T.)

Tries to parse the Enum Value

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public bool TryParse(
    string strValue,
    out T value
)
```

Parameters

strValue	Type: System.String Enum value (in string representation).
value	Type: T [▶ 1738]. The value.

Return Value

Type: Boolean

true if the [EnumType.T.](#) [[▶ 1738](#)] could be parsed/>, false otherwise.

Implements

[IEnumType.T..TryParse\(String, T.\)](#) [[▶ 2523](#)]

Reference

[EnumType.T. Class](#) [[▶ 1738](#)]

[TryParse Overload](#) [[▶ 1749](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.6 Field Class

Represents a field of an Struct/Alias/Union

Inheritance Hierarchy

System.Object

 TwinCAT.Ads.TypeSystem.Instance [[▶ 1765](#)]

 TwinCAT.Ads.TypeSystem.Field

 TwinCAT.Ads.TypeSystem.Member [[▶ 1781](#)]

Namespace: TwinCAT.Ads.TypeSystem [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax








C#

```
public class Field : Instance, IField,
    IAttributedInstance, IInstance, IBitSize
```











The Field type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1768]	Gets the Type Attributes. (Inherited from Instance [▶ 1765].)
	BitSize [▶ 1768]	Gets the size of this Instance [▶ 1765] in bits. (Inherited from Instance [▶ 1765].)
	ByteSize [▶ 1769]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from Instance [▶ 1765].)
	Category [▶ 1769]	Gets the the DataTypeCategory [▶ 2111] of the Instance. (Inherited from Instance [▶ 1765].)
	Comment [▶ 1770]	Gets the comment. (Inherited from Instance [▶ 1765].)
	ContextMask [▶ 1770]	Gets the context mask of this instance. (Inherited from Instance [▶ 1765].)
	DataType [▶ 1771]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from Instance [▶ 1765].)
	HasValue [▶ 1771]	Gets a value indicating whether this instance has a value. (Inherited from Instance [▶ 1765].)
	InstanceName [▶ 1771]	Gets the name of the instance (without periods (.)) (Inherited from Instance [▶ 1765].)
	InstancePath [▶ 1772]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from Instance [▶ 1765].)
	IsBitType [▶ 1773]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from Instance [▶ 1765].)
	IsByteAligned [▶ 1773]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from Instance [▶ 1765].)
	IsPersistent [▶ 1773]	Indicates that this instance is persistent. (Inherited from Instance [▶ 1765].)
	IsPointer [▶ 1774]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
	IsReadOnly [▶ 1774]	Indicates that this instance is read only. (Inherited from Instance [▶ 1765].)
	IsReference [▶ 1775]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
	IsStatic [▶ 1775]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from Instance [▶ 1765].)

	Name	Description
	IsTcComInterfacePointer [▶ 1776]	Indicates that this instance is a TcComInterfacePointer. (Inherited from Instance [▶ 1765].)
	IsTypeGuid [▶ 1776]	Indicates that this instance has set TypeGuid flag. (Inherited from Instance [▶ 1765].)
	Namespace [▶ 1776]	Gets the namespace name. (Inherited from Instance [▶ 1765].)
	ParentType [▶ 1754]	Gets the Parent of this Field [▶ 2535].
	Size [▶ 1777]	Gets the size of the DataType [▶ 2475] in bytes or Bits dependant on IsBitType [▶ 1773] (Inherited from Instance [▶ 1765].)
	TypeName [▶ 1777]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from Instance [▶ 1765].)
	ValueEncoding [▶ 1754]	Gets the value encoding of this Field

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	OnGetSize [▶ 1779]	Handler function getting the size of the Instance [▶ 1765] (Inherited from Instance [▶ 1765].)
	OnSetInstanceName [▶ 1780]	Sets a new InstanceName InstancePath (Inherited from Instance [▶ 1765].)
	SetAttributes [▶ 1780]	Sets the type attributes (Inherited from Instance [▶ 1765].)
	SetContextMask [▶ 1780]	Sets the context mask. (Inherited from Instance [▶ 1765].)
	ToString [▶ 1781]	Returns a String that represents this instance. (Inherited from Instance [▶ 1765].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference



[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.6.1 Field Properties

The [Field \[▶ 1750\]](#) type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1768]	Gets the Type Attributes. (Inherited from Instance [▶ 1765].)
	BitSize [▶ 1768]	Gets the size of this Instance [▶ 1765] in bits. (Inherited from Instance [▶ 1765].)
	ByteSize [▶ 1769]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from Instance [▶ 1765].)
	Category [▶ 1769]	Gets the the DataTypeCategory [▶ 2111] of the Instance. (Inherited from Instance [▶ 1765].)
	Comment [▶ 1770]	Gets the comment. (Inherited from Instance [▶ 1765].)
	ContextMask [▶ 1770]	Gets the context mask of this instance. (Inherited from Instance [▶ 1765].)
	DataType [▶ 1771]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549] . (Inherited from Instance [▶ 1765].)
	HasValue [▶ 1771]	Gets a value indicating whether this instance has a value. (Inherited from Instance [▶ 1765].)
	InstanceName [▶ 1771]	Gets the name of the instance (without periods (.)) (Inherited from Instance [▶ 1765].)
	InstancePath [▶ 1772]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from Instance [▶ 1765].)
	IsBitType [▶ 1773]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from Instance [▶ 1765].)
	IsByteAligned [▶ 1773]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from Instance [▶ 1765].)
	IsPersistent [▶ 1773]	Indicates that this instance is persistent. (Inherited from Instance [▶ 1765].)
	IsPointer [▶ 1774]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
	IsReadOnly [▶ 1774]	Indicates that this instance is read only. (Inherited from Instance [▶ 1765].)
	IsReference [▶ 1775]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
	IsStatic [▶ 1775]	Gets a value indicating whether this Instance [▶ 2549] is static. (Inherited from Instance [▶ 1765].)
	IsTcComInterfacePointer [▶ 1776]	Indicates that this instance is a TcComInterfacePointer . (Inherited from Instance [▶ 1765].)
	IsTypeGuid [▶ 1776]	Indicates that this instance has set TypeGuid flag. (Inherited from Instance [▶ 1765].)
	Namespace [▶ 1776]	Gets the namespace name. (Inherited from Instance [▶ 1765].)
	ParentType [▶ 1754]	Gets the Parent of this Field [▶ 2535] .
	Size [▶ 1777]	Gets the size of the IDataType [▶ 2475] in bytes or Bits dependant on IsBitType [▶ 1773] (Inherited from Instance [▶ 1765].)

	Name	Description
	TypeName [▶ 1777]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from Instance [▶ 1765].)
	ValueEncoding [▶ 1754]	Gets the value encoding of this Field [▶ 1750]

Reference

[Field Class](#) [[▶ 1750](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.6.1.1 Field.ParentType Property

Gets the Parent of this [IField](#) [[▶ 2535](#)].

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDatatype? ParentType { get; }
```

Property Value

Type: [IDatatype](#) [[▶ 2475](#)]

The type of the parent (Alias, Union, Struct)

Implements

[IField.ParentType](#) [[▶ 2538](#)]

Reference

[Field Class](#) [[▶ 1750](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.6.1.2 Field.ValueEncoding Property

Gets the value encoding of this [Field](#) [[▶ 1750](#)]

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public virtual Encoding ValueEncoding { get; }
```

Property Value

Type: [Encoding](#)

The value encoding.

Implements

[IAttributedInstance.ValueEncoding](#) [[▶ 2472](#)]

Reference











[Field Class](#) [[▶ 1750](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.6.2 Field Methods

The [Field](#) [[▶ 1750](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	OnGetSize [▶ 1779]	Handler function getting the size of the Instance [▶ 1765] (Inherited from Instance [▶ 1765].)
	OnSetInstanceName [▶ 1780]	Sets a new InstanceName InstancePath (Inherited from Instance [▶ 1765].)
	SetAttributes [▶ 1780]	Sets the type attributes (Inherited from Instance [▶ 1765].)
	SetContextMask [▶ 1780]	Sets the context mask. (Inherited from Instance [▶ 1765].)
	ToString [▶ 1781]	Returns a String that represents this instance. (Inherited from Instance [▶ 1765].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[Field Class](#) [[▶ 1750](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.7 IAdsSymbol Interface

Interface [IAdsSymbol](#)

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229














Syntax












C#

```
public interface IAdsSymbol : ISymbol,
    IAttributedInstance, IInstance, IBitSize, IProcessImageAddress, IContextMaskProvider
```

The IAdsSymbol type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	ContextMask [▶ 1765]	Gets the ContextMask of the symbol, indicating the task the variable belongs to. If ContextMask is not zero use CyclicInContext [▶ 740] or OnChangeInContext [▶ 740] to add notifications. (Inherited from IContextMaskProvider [▶ 1764].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	DataTypeId [▶ 1759]	DataType identifier of the Symbol AdsDataTypeId [▶ 663]
	ImageBaseAddress [▶ 1759]	Gets the AmsAddress [▶ 752] of the Process Image
	IndexGroup [▶ 2594]	Gets the index group of the Symbol (Inherited from IProcessImageAddress [▶ 2593].)
	IndexOffset [▶ 2594]	Gets the index offset of the Symbol (Inherited from IProcessImageAddress [▶ 2593].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)

	Name	Description
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	IsVirtual [▶ 2595]	Gets a value indicating whether this instance is virtual. (Inherited from IProcessImageAddress [▶ 2593].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]





[TwinCAT.TypeSystem.ISymbol](#) [[▶ 2691](#)]


[TwinCAT.TypeSystem.IProcessImageAddress](#) [[▶ 2593](#)]



6.7.7.1 IAdsSymbol Properties

The [IAdsSymbol](#) [[▶ 1755](#)] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)

	Name	Description
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	ContextMask [▶ 1765]	Gets the ContextMask of the symbol, indicating the task the variable belongs to. If ContextMask is not zero use CyclicInContext [▶ 740] or OnChangeInContext [▶ 740] to add notifications. (Inherited from IContextMaskProvider [▶ 1764].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	DataTypeId [▶ 1759]	DataType identifier of the Symbol AdsDataTypeId [▶ 663]
	ImageBaseAddress [▶ 1759]	Gets the AmsAddress [▶ 752] of the Process Image
	IndexGroup [▶ 2594]	Gets the index group of the Symbol (Inherited from IProcessImageAddress [▶ 2593].)
	IndexOffset [▶ 2594]	Gets the index offset of the Symbol (Inherited from IProcessImageAddress [▶ 2593].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is a container type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	IsVirtual [▶ 2595]	Gets a value indicating whether this instance is virtual. (Inherited from IProcessImageAddress [▶ 2593].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)

	Name	Description
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from Instance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from AttributedInstance [▶ 2469].)

Reference

[IAdsSymbol Interface](#) [[▶ 1755](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.7.1.1 IAdsSymbol.DataTypeId Property

DataType identifier of the Symbol [AdsDataTypeId](#) [[▶ 663](#)]

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsDataTypeId DataTypeId { get; }
```

Property Value

Type: [AdsDataTypeId](#) [[▶ 663](#)]

Data type of the symbol.

Reference

[IAdsSymbol Interface](#) [[▶ 1755](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.7.1.2 IAdsSymbol.ImageBaseAddress Property

Gets the [AmsAddress](#) [[▶ 752](#)] of the Process Image

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AmsAddress? ImageBaseAddress { get; }
```

Property Value

Type: [AmsAddress](#) [[▶ 752](#)]

The address.

Reference

[IAdsSymbol Interface](#) [[▶ 1755](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.7.2 IAdsSymbol Methods

The [IAdsSymbol](#) [[▶ 1755](#)] type exposes the following members.

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[IAdsSymbol Interface](#) [[▶ 1755](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.8 IAdsSymbolLoader Interface

Symbol Loader interface

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229











Syntax

C#






```
public interface IAdsSymbolLoader : ISymbolLoader,
    ISymbolProvider, ISymbolServer
```

The [IAdsSymbolLoader](#) type exposes the following members.

Properties

	Name	Description
	BuildInTypes [▶ 2716]	Gets the build in types. (Inherited from ISymbolLoader [▶ 2714].)
	DataTypes [▶ 2720]	Gets the data types (Inherited from ISymbolServer [▶ 2719].)
 	DefaultNotificationSettings [▶ 1762]	Gets/Sets the default notification settings for this SymbolLoader
	DefaultValueEncoding [▶ 2721]	Gets the default value encoding. (Inherited from ISymbolServer [▶ 2719].)
	ImageBaseAddress [▶ 1763]	Gets the image base address.
	PlatformPointerSize [▶ 1763]	Gets the (byte) size of Pointers on the attached platform system.
	RootNamespaceName [▶ 2718]	Gets the name of the root namespace (Inherited from ISymbolProvider [▶ 2717].)
	Settings [▶ 2716]	Gets or sets the access Method (Inherited from ISymbolLoader [▶ 2714].)
	Symbols [▶ 2721]	Gets the symbols. (Inherited from ISymbolServer [▶ 2719].)

Methods

	Name	Description
	GetDataTypesAsync [▶ 2722]	Gets the data types asynchronously. (Inherited from ISymbolServer [▶ 2719].)
	GetSymbolsAsync [▶ 2723]	Gets the symbols asynchronously (Inherited from ISymbolServer [▶ 2719].)
	ResetCachedSymbolicData [▶ 2724]	Resets the cached symbolic data. (Inherited from ISymbolServer [▶ 2719].)
	TryGetDataTypes [▶ 2723]	Tries to get the symbols from the device target. (Inherited from ISymbolServer [▶ 2719].)
	TryGetSymbols [▶ 2724]	Tries to geth the symbols from the device target. (Inherited from ISymbolServer [▶ 2719].)











Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.8.1 IAdsSymbolLoader Properties

The [IAdsSymbolLoader](#) [[▶ 1760](#)] type exposes the following members.

Properties

	Name	Description
	BuildInTypes [▶ 2716]	Gets the build in types. (Inherited from ISymbolLoader [▶ 2714].)
	DataTypes [▶ 2720]	Gets the data types (Inherited from ISymbolServer [▶ 2719].)
 	DefaultNotificationSettings [▶ 1762]	Gets/Sets the default notification settings for this SymbolLoader
	DefaultValueEncoding [▶ 2721]	Gets the default value encoding. (Inherited from ISymbolServer [▶ 2719].)
	ImageBaseAddress [▶ 1763]	Gets the image base address.
	PlatformPointerSize [▶ 1763]	Gets the (byte) size of Pointers on the attached platform system.
	RootNamespaceName [▶ 2718]	Gets the name of the root namespace (Inherited from ISymbolProvider [▶ 2717].)
	Settings [▶ 2716]	Gets or sets the access Method (Inherited from ISymbolLoader [▶ 2714].)
	Symbols [▶ 2721]	Gets the symbols. (Inherited from ISymbolServer [▶ 2719].)

Reference

[IAdsSymbolLoader Interface](#) [[▶ 1760](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.8.1.1 IAdsSymbolLoader.DefaultNotificationSettings Property

Gets/Sets the default notification settings for this SymbolLoader

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
INotificationSettings DefaultNotificationSettings { get; set; }
```

Property Value

Type: [INotificationSettings](#) [[▶ 1098](#)]

The default notification settings.

Remarks

The Default notification Settings can be set on the SymbolLoader and is used as default on the different Symbols. On the symbol itself the [NotificationSettings](#) [[▶ 1106](#)] can be overridden.

Examples

Setting the DefaultNotificationSettings on the [IAdsSymbolLoader](#) [[▶ 1760](#)] object:

Set DefaultNotificationSettings

```
// Create AdsClient object
using (AdsClient client = new AdsClient())
{
    // No automatic Synchronization (necessary for Console applications without message loop)
    //client.Synchronize = false;

    // Connect to client
    client.Connect(address);

    // Usage of 'dynamic' type/symbol loader
    SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.DynamicTree, ValueAccessMode.IndexGroupOffsetPreferred);
    IAdsSymbolLoader dynLoader = (IAdsSymbolLoader)SymbolLoaderFactory.Create(client, settings);

    // Set the DefaultNotification Properties
    dynLoader.DefaultNotificationSettings = new NotificationSettings(AdsTransMode.ClientOnChange, 200, 2000);

    // Determine the symbols
    dynamic dynamicSymbols = ((IDynamicSymbolLoader)dynLoader).SymbolsDynamic;

    // Task 1 Symbol (build in symbol)
    dynamic task1Symbol = dynamicSymbols._TaskInfo[1];

    // CycleCount Symbol
    dynamic cycleCountSymbol = task1Symbol.CycleCount;

    // Override Notification Setting for Cycle Count Symbol
    cycleCountSymbol.NotificationSettings = new NotificationSettings(AdsTransMode.OnChange, 250, 0);

    // Register Dynamic Value Changed event.
    cycleCountSymbol.ValueChanged += new EventHandler<ValueChangedEventArgs>(cycleCount_ValueChanged);
};
```

Reference

[IAdsSymbolLoader Interface](#) [[▶ 1760](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.8.1.2 IAdsSymbolLoader.ImageBaseAddress Property

Gets the image base address.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AmsAddress ImageBaseAddress { get; }
```

Property Value

Type: [AmsAddress](#) [[▶ 752](#)]

The image base address.

Reference

[IAdsSymbolLoader Interface](#) [[▶ 1760](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.8.1.3 IAdsSymbolLoader.PlatformPointerSize Property

Gets the (byte) size of Pointers on the attached platform system.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int PlatformPointerSize { get; }
```

Property Value

Type: Int32

The size of the platform pointer.

Reference



[IAdsSymbolLoader Interface](#) [[▶ 1760](#)]




[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.8.2 IAdsSymbolLoader Methods

The [IAdsSymbolLoader](#) [[▶ 1760](#)] type exposes the following members.

Methods

	Name	Description
	GetDataTypesAsync [▶ 2722]	Gets the data types asynchronously. (Inherited from ISymbolServer [▶ 2719].)
	GetSymbolsAsync [▶ 2723]	Gets the symbols asynchronously (Inherited from ISymbolServer [▶ 2719].)

	Name	Description
	ResetCachedSymbolicData [▶ 2724]	Resets the cached symbolic data. (Inherited from ISymbolServer [▶ 2719].)
	TryGetDataTypes [▶ 2723]	Tries to get the symbols from the device target. (Inherited from ISymbolServer [▶ 2719].)
	TryGetSymbols [▶ 2724]	Tries to geth the symbols from the device target. (Inherited from ISymbolServer [▶ 2719].)

Reference

[IAdsSymbolLoader Interface](#) [[▶ 1760](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.9 IContextMaskProvider Interface

Interface IContextMaskProvider

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#

```
public interface IContextMaskProvider
```

The IContextMaskProvider type exposes the following members.

Properties

	Name	Description
	ContextMask [▶ 1765]	Gets the ContextMask of the symbol, indicating the task the variable belongs to. If ContextMask is not zero use CyclicInContext [▶ 740] or OnChangeInContext [▶ 740] to add notifications.


Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.9.1 IContextMaskProvider Properties

The [IContextMaskProvider](#) [[▶ 1764](#)] type exposes the following members.

Properties

	Name	Description
	ContextMask [▶ 1765]	Gets the ContextMask of the symbol, indicating the task the variable belongs to. If ContextMask is not zero use CyclicInContext [▶ 740] or OnChangeInContext [▶ 740] to add notifications.

Reference

[IContextMaskProvider Interface](#) [[▶ 1764](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.9.1.1 IContextMaskProvider.ContextMask Property

Gets the ContextMask of the symbol, indicating the task the variable belongs to. If ContextMask is not zero use [CyclicInContext](#) [▶ 740] or [OnChangeInContext](#) [▶ 740] to add notifications.

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
byte ContextMask { get; }
```

Property Value

Type: Byte

Reference

[IContextMaskProvider Interface](#) [▶ 1764]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.10 Instance Class

Instance implementation

Inheritance Hierarchy

System.Object

 TwinCAT.Ads.TypeSystem.Instance

[TwinCAT.Ads.TypeSystem.Field](#) [▶ 1750]

[TwinCAT.Ads.TypeSystem.Symbol](#) [▶ 1863]

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229







Syntax

















C#

```
public class Instance : IInstance, IBitSize
```









The Instance type exposes the following members.




Properties

	Name	Description
	Attributes [▶ 1768]	Gets the Type Attributes.
	BitSize [▶ 1768]	Gets the size of this Instance in bits.
	ByteSize [▶ 1769]	Gets the (aligned) size of of the Type/Instance in Bytes
	Category [▶ 1769]	Gets the the DataTypeCategory [▶ 2111] of the Instance.
	Comment [▶ 1770]	Gets the comment.
	ContextMask [▶ 1770]	Gets the context mask of this instance.

	Name	Description
	DataType [▶ 1771]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549].
	HasValue [▶ 1771]	Gets a value indicating whether this instance has a value.
	InstanceName [▶ 1771]	Gets the name of the instance (without periods (.))
	InstancePath [▶ 1772]	Gets the relative / absolute access path to the instance (with periods (.))
	IsBitType [▶ 1773]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping
	IsByteAligned [▶ 1773]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$)
	IsPersistent [▶ 1773]	Indicates that this instance is persistent.
	IsPointer [▶ 1774]	Gets a value indicating whether this instance is reference.
	IsReadOnly [▶ 1774]	Indicates that this instance is read only.
	IsReference [▶ 1775]	Gets a value indicating whether this instance is reference.
	IsStatic [▶ 1775]	Gets a value indicating whether this Instance [▶ 2549] is static.
	IsTcComInterfacePointer [▶ 1776]	Indicates that this instance is a TcComInterfacePointer .
	IsTypeGuid [▶ 1776]	Indicates that this instance has set TypeGuid flag.
	Namespace [▶ 1776]	Gets the namespace name.
	Size [▶ 1777]	Gets the size of the IDataType [▶ 2475] in bytes or Bits dependant on IsBitType [▶ 1773]
	TypeName [▶ 1777]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549].

Methods

	Name	Description
	AlignTypeName [▶ 1779]	Aligns the type name
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnGetSize [▶ 1779]	Handler function getting the size of the Instance
	OnSetInstanceName [▶ 1780]	Sets a new InstanceName InstancePath

	Name	Description
	SetAttributes [▶ 1780]	Sets the type attributes
	SetContextMask [▶ 1780]	Sets the context mask.
	ToString [▶ 1781]	Returns a String that represents this instance. (Overrides Object.ToString..)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
















Reference








[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.10.1 Instance Properties

The [Instance](#) [▶ 1765] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1768]	Gets the Type Attributes.
	BitSize [▶ 1768]	Gets the size of this Instance [▶ 1765] in bits.
	ByteSize [▶ 1769]	Gets the (aligned) size of of the Type/Instance in Bytes
	Category [▶ 1769]	Gets the the DataTypeCategory [▶ 2111] of the Instance.
	Comment [▶ 1770]	Gets the comment.
	ContextMask [▶ 1770]	Gets the context mask of this instance.
	DataType [▶ 1771]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549].
	HasValue [▶ 1771]	Gets a value indicating whether this instance has a value.
	InstanceName [▶ 1771]	Gets the name of the instance (without periods (.))
	InstancePath [▶ 1772]	Gets the relative / absolute access path to the instance (with periods (.))
	IsBitType [▶ 1773]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping
	IsByteAligned [▶ 1773]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$)
	IsPersistent [▶ 1773]	Indicates that this instance is persistent.
	IsPointer [▶ 1774]	Gets a value indicating whether this instance is reference.
	IsReadOnly [▶ 1774]	Indicates that this instance is read only.

	Name	Description
	IsReference [▶ 1775]	Gets a value indicating whether this instance is reference.
	IsStatic [▶ 1775]	Gets a value indicating whether this IInstance [▶ 2549] is static.
	IsTcComInterfacePointer [▶ 1776]	Indicates that this instance is a TcComInterfacePointer.
	IsTypeGuid [▶ 1776]	Indicates that this instance has set TypeGuid flag.
	Namespace [▶ 1776]	Gets the namespace name.
	Size [▶ 1777]	Gets the size of the IDataType [▶ 2475] in bytes or Bits dependant on IsBitType [▶ 1773]
	TypeName [▶ 1777]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549].

Reference

[Instance Class](#) [[▶ 1765](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.10.1 Instance.Attributes Property

Gets the Type Attributes.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ITypeAttributeCollection Attributes { get; }
```

Property Value

Type: [ITypeAttributeCollection](#) [[▶ 2726](#)]

The attributes.

Reference

[Instance Class](#) [[▶ 1765](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.10.2 Instance.BitSize Property

Gets the size of this [Instance](#) [[▶ 1765](#)] in bits.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public virtual int BitSize { get; }
```


Property Value

Type: Int32
The size of the bit.

Implements

[IBitSize.BitSize](#) [[▶ 2473](#)]

Reference

[Instance Class](#) [[▶ 1765](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.10.1.3 Instance.ByteSize Property

Gets the (aligned) size of of the Type/Instance in Bytes

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ByteSize { get; }
```

Property Value

Type: Int32
The size of the byte.

Implements

[IBitSize.ByteSize](#) [[▶ 2474](#)]

Reference

[Instance Class](#) [[▶ 1765](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.10.1.4 Instance.Category Property

Gets the the [DataTypeCategory](#) [[▶ 2111](#)] of the Instance.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DataTypeCategory Category { get; protected set; }
```

Property Value

Type: [DataTypeCategory](#) [[▶ 2111](#)]
The category.

Remarks

Corresponds to the [Category \[▶ 2478\]](#)

Reference

[Instance Class \[▶ 1765\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.10.1.5 Instance.Comment Property

Gets the comment.

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Comment { get; }
```

Property Value

Type: String

The comment.

Implements

[IInstance.Comment \[▶ 2550\]](#)

Reference

[Instance Class \[▶ 1765\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.10.1.6 Instance.ContextMask Property

Gets the context mask of this instance.

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public byte ContextMask { get; }
```

Property Value

Type: Byte

Remarks

The Size of the internal data is 4-Bit

Reference

[Instance Class \[▶ 1765\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.10.1.7 Instance.DataType Property

Gets the [IDataType \[▸ 2475\]](#) of the [IInstance \[▸ 2549\]](#).

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataType? DataType { get; protected set; }
```

Property Value

Type: [IDataType \[▸ 2475\]](#)

The type of the data.

Implements

[IInstance.DataType \[▸ 2551\]](#)

Reference

[Instance Class \[▸ 1765\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.10.1.8 Instance.HasValue Property

Gets a value indicating whether this instance has a value.

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public virtual bool HasValue { get; }
```

Property Value

Type: Boolean

true if this instance has value; otherwise, false.

Remarks

Reference

[Instance Class \[▸ 1765\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.10.1.9 Instance.InstanceName Property

Gets the name of the instance (without periods (.))

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string InstanceName { get; protected set; }
```

Property Value

Type: String

The name of the instance.

Implements

[IInstance.InstanceName \[▸ 2551\]](#)

Reference

[Instance Class \[▸ 1765\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.10.1.10 Instance.InstancePath Property

Gets the relative / absolute access path to the instance (with periods (.))

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public virtual string InstancePath { get; }
```

Property Value

Type: String

The instance path.

Implements

[IInstance.InstancePath \[▸ 2552\]](#)

Remarks

If this path is relative or absolute depends on the context. [IMember \[▸ 2561\]](#) are using relative paths, [ISymbol \[▸ 2691\]](#)s are using absolute ones.

Reference

[Instance Class \[▸ 1765\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.10.1.11 Instance.IsBitType Property

Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsBitType { get; }
```

Property Value

Type: Boolean

true if this instance is bit mapping; otherwise, false.

Implements

[IBitSize.IsBitType](#) [[▶ 2474](#)]

Reference

[Instance Class](#) [[▶ 1765](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.10.1.12 Instance.IsByteAligned Property

Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$)

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsByteAligned { get; }
```

Property Value

Type: Boolean

true if this instance is byte aligned; otherwise, false.

Implements

[IBitSize.IsByteAligned](#) [[▶ 2475](#)]

Reference

[Instance Class](#) [[▶ 1765](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.10.1.13 Instance.IsPersistent Property

Indicates that this instance is persistent.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsPersistent { get; }
```

Property Value

Type: Boolean

Reference

[Instance Class](#) [[▶ 1765](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.10.1.14 Instance.IsPointer Property

Gets a value indicating whether this instance is reference.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsPointer { get; }
```

Property Value

Type: Boolean

true if this instance is reference; otherwise, false.

Implements

[IInstance.IsPointer](#) [[▶ 2552](#)]

Reference

[Instance Class](#) [[▶ 1765](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.10.1.15 Instance.IsReadOnly Property

Indicates that this instance is read only.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsReadOnly { get; }
```

Property Value

Type: Boolean

Remarks

Actually, this Flag is restricted to TcCOM-Objects readonly Parameters. Within the PLC this is used for the ApplicationName and ProjectName of PLC instances. Write-Access on these Modules will create an [DeviceAccessDenied \[▸ 664\]](#) error.

Reference

[Instance Class \[▸ 1765\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.10.1.16 Instance.IsReference Property

Gets a value indicating whether this instance is reference.

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsReference { get; }
```

Property Value

Type: Boolean

true if this instance is reference; otherwise, false.

Implements

[IInstance.IsReference \[▸ 2552\]](#)

Reference

[Instance Class \[▸ 1765\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.10.1.17 Instance.IsStatic Property

Gets a value indicating whether this [IInstance \[▸ 2549\]](#) is static.

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsStatic { get; protected set; }
```

Property Value

Type: Boolean

true if this instance is static; otherwise, false.

Implements

[IInstance.IsStatic](#) [[▶ 2553](#)]

Reference

[Instance Class](#) [[▶ 1765](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.10.1.18 Instance.IsTcComInterfacePointer Property

Indicates that this instance is a TcComInterfacePointer.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsTcComInterfacePointer { get; }
```

Property Value

Type: Boolean

Reference

[Instance Class](#) [[▶ 1765](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.10.1.19 Instance.IsTypeGuid Property

Indicates that this instance has set TypeGuid flag.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsTypeGuid { get; }
```

Property Value

Type: Boolean

Reference

[Instance Class](#) [[▶ 1765](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.10.1.20 Instance.Namespace Property

Gets the namespace name.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Namespace { get; }
```

Property Value

Type: String

The namespace.

Reference

[Instance Class](#) [[▶ 1765](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.10.1.21 Instance.Size Property

Gets the size of the [IDataType](#) [[▶ 2475](#)] in bytes or Bits dependant on [IsBitType](#) [[▶ 1773](#)]

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Size { get; protected set; }
```

Property Value

Type: Int32

The size of the bit.

Implements

[IBitSize.Size](#) [[▶ 2475](#)]

Reference

[Instance Class](#) [[▶ 1765](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.10.1.22 Instance.TypeName Property

Gets the name of the [DataType](#) [[▶ 2475](#)] that is used for this [Instance](#) [[▶ 2549](#)].

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string TypeName { get; protected set; }
```

Property Value

Type: String
The name of the type.

Implements

[IInstance.TypeName](#) [[▶ 2553](#)]

Reference













[Instance Class](#) [[▶ 1765](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.10.2 Instance Methods

The [Instance](#) [[▶ 1765](#)] type exposes the following members.

Methods

	Name	Description
	AlignTypeName [▶ 1779]	Aligns the type name
		
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	OnGetSize [▶ 1779]	Handler function getting the size of the Instance [▶ 1765]
	OnSetInstanceName [▶ 1780]	Sets a new InstanceName InstancePath
	SetAttributes [▶ 1780]	Sets the type attributes
	SetContextMask [▶ 1780]	Sets the context mask.
	ToString [▶ 1781]	Returns a String that represents this instance. (Overrides Object.ToString..)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[Instance Class](#) [[▶ 1765](#)]

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.10.2.1 Instance.AlignTypeName Method

Aligns the type name

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected static string AlignTypeName(  
    string typeName  
)
```

Parameters

typeName	Type: System.String Name of the type.
----------	--

Return Value

Type: String
System.String.

Exceptions

Exception	Condition
ArgumentException	Type name not valid!

Reference

[Instance Class \[► 1765\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.10.2.2 Instance.OnGetSize Method

Handler function getting the size of the [Instance \[► 1765\]](#)

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual int OnGetSize()
```

Return Value

Type: Int32
System.Int32.

Reference

[Instance Class \[► 1765\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.10.2.3 Instance.OnSetInstanceName Method

Sets a new InstanceName InstancePath

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual void OnSetInstanceName(  
    string instanceName  
)
```

Parameters

instanceName	Type: System.String Instance name.
--------------	---------------------------------------

Reference

[Instance Class](#) [[▶ 1765](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.10.2.4 Instance.SetAttributes Method

Sets the type attributes

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected void SetAttributes(  
    TypeAttributeCollection? coll  
)
```

Parameters

coll	Type: TwinCAT.TypeSystem.TypeAttributeCollection [▶ 2952] The attributes.
------	--

Reference

[Instance Class](#) [[▶ 1765](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.10.2.5 Instance.SetContextMask Method

Sets the context mask.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected void SetContextMask(
    byte contextMask
)
```

Parameters

contextMask	Type: System.Byte The context mask.
-------------	--

Exceptions

Exception	Condition
ArgumentOutOfRangeException	contextMask

Reference

[Instance Class \[► 1765\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.10.2.6 Instance.ToString Method

Returns a String that represents this instance.

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```

Return Value

Type: String

A String that represents this instance.

Reference

[Instance Class \[► 1765\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.11 Member Class

Represents a member of an [StructType \[► 1846\]](#)

Inheritance Hierarchy

System.Object

[TwinCAT.Ads.TypeSystem.Instance \[► 1765\]](#)

[TwinCAT.Ads.TypeSystem.Field \[► 1750\]](#)

[TwinCAT.Ads.TypeSystem.Member](#)

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax








C#

```
public sealed class Member : Field,
    IMember, IField, IAttributedInstance, IInstance, IBitSize
```





The Member type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1768]	Gets the Type Attributes. (Inherited from Instance [▶ 1765].)
	BitOffset [▶ 1785]	Gets the bit offset.
	BitSize [▶ 1768]	Gets the size of this Instance [▶ 1765] in bits. (Inherited from Instance [▶ 1765].)
	ByteOffset [▶ 1785]	Gets the byte offset.
	ByteSize [▶ 1769]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from Instance [▶ 1765].)
	Category [▶ 1769]	Gets the the DataTypeCategory [▶ 2111] of the Instance. (Inherited from Instance [▶ 1765].)
	Comment [▶ 1770]	Gets the comment. (Inherited from Instance [▶ 1765].)
	ContextMask [▶ 1770]	Gets the context mask of this instance. (Inherited from Instance [▶ 1765].)
	DataType [▶ 1771]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from Instance [▶ 1765].)
	HasValue [▶ 1771]	Gets a value indicating whether this instance has a value. (Inherited from Instance [▶ 1765].)
	InstanceName [▶ 1771]	Gets the name of the instance (without periods (.)) (Inherited from Instance [▶ 1765].)
	InstancePath [▶ 1772]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from Instance [▶ 1765].)
	IsBitType [▶ 1773]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from Instance [▶ 1765].)
	IsByteAligned [▶ 1773]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from Instance [▶ 1765].)
	IsPersistent [▶ 1773]	Indicates that this instance is persistent. (Inherited from Instance [▶ 1765].)
	IsPointer [▶ 1774]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
	IsReadOnly [▶ 1774]	Indicates that this instance is read only. (Inherited from Instance [▶ 1765].)
	IsReference [▶ 1775]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
	IsStatic [▶ 1775]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from Instance [▶ 1765].)
	IsTcComInterfacePointer [▶ 1776]	Indicates that this instance is a TcComInterfacePointer . (Inherited from Instance [▶ 1765].)

	Name	Description
	IsTypeGuid [▶ 1776]	Indicates that this instance has set TypeGuid flag. (Inherited from Instance [▶ 1765] .)
	Namespace [▶ 1776]	Gets the namespace name. (Inherited from Instance [▶ 1765] .)
	Offset [▶ 1786]	Gets the offset of the Member within the parent StructType [▶ 1846] in bits or bytes dependent on IsBitType [▶ 1773]
	ParentType [▶ 1754]	Gets the Parent of this IField [▶ 2535] . (Inherited from Field [▶ 1750] .)
	Size [▶ 1777]	Gets the size of the IDataType [▶ 2475] in bytes or Bits dependant on IsBitType [▶ 1773] (Inherited from Instance [▶ 1765] .)
	TypeName [▶ 1777]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549] . (Inherited from Instance [▶ 1765] .)
	ValueEncoding [▶ 1754]	Gets the value encoding of this Field [▶ 1750] (Inherited from Field [▶ 1750] .)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString [▶ 1781]	Returns a String that represents this instance. (Inherited from Instance [▶ 1765] .)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021] .)





Reference

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.11.1 Member Properties

The [Member \[▶ 1781\]](#) type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1768]	Gets the Type Attributes. (Inherited from Instance [▶ 1765] .)
	BitOffset [▶ 1785]	Gets the bit offset.
	BitSize [▶ 1768]	Gets the size of this Instance [▶ 1765] in bits. (Inherited from Instance [▶ 1765] .)
	ByteOffset [▶ 1785]	Gets the byte offset.

	Name	Description
	ByteSize [▶ 1769]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from Instance [▶ 1765].)
	Category [▶ 1769]	Gets the the DataTypeCategory [▶ 2111] of the Instance. (Inherited from Instance [▶ 1765].)
	Comment [▶ 1770]	Gets the comment. (Inherited from Instance [▶ 1765].)
	ContextMask [▶ 1770]	Gets the context mask of this instance. (Inherited from Instance [▶ 1765].)
	DataType [▶ 1771]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549] . (Inherited from Instance [▶ 1765].)
	HasValue [▶ 1771]	Gets a value indicating whether this instance has a value. (Inherited from Instance [▶ 1765].)
	InstanceName [▶ 1771]	Gets the name of the instance (without periods (.)) (Inherited from Instance [▶ 1765].)
	InstancePath [▶ 1772]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from Instance [▶ 1765].)
	IsBitType [▶ 1773]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from Instance [▶ 1765].)
	IsByteAligned [▶ 1773]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from Instance [▶ 1765].)
	IsPersistent [▶ 1773]	Indicates that this instance is persistent. (Inherited from Instance [▶ 1765].)
	IsPointer [▶ 1774]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
	IsReadOnly [▶ 1774]	Indicates that this instance is read only. (Inherited from Instance [▶ 1765].)
	IsReference [▶ 1775]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
	IsStatic [▶ 1775]	Gets a value indicating whether this Instance [▶ 2549] is static. (Inherited from Instance [▶ 1765].)
	IsTcComInterfacePointer [▶ 1776]	Indicates that this instance is a TcComInterfacePointer . (Inherited from Instance [▶ 1765].)
	IsTypeGuid [▶ 1776]	Indicates that this instance has set TypeGuid flag. (Inherited from Instance [▶ 1765].)
	Namespace [▶ 1776]	Gets the namespace name. (Inherited from Instance [▶ 1765].)
	Offset [▶ 1786]	Gets the offset of the Member [▶ 1781] within the parent StructType [▶ 1846] in bits or bytes dependent on IsBitType [▶ 1773]
	ParentType [▶ 1754]	Gets the Parent of this Field [▶ 2535] . (Inherited from Field [▶ 1750].)
	Size [▶ 1777]	Gets the size of the IDataType [▶ 2475] in bytes or Bits dependant on IsBitType [▶ 1773] (Inherited from Instance [▶ 1765].)
	TypeName [▶ 1777]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549] . (Inherited from Instance [▶ 1765].)
	ValueEncoding [▶ 1754]	Gets the value encoding of this Field [▶ 1750] (Inherited from Field [▶ 1750].)

Reference

[Member Class](#) [► 1781]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.11.1.1 Member.BitOffset Property

Gets the bit offset.

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int BitOffset { get; }
```

Property Value

Type: Int32

The bit offset.

Implements

[IMember.BitOffset](#) [► 2563]

Reference

[Member Class](#) [► 1781]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.11.1.2 Member.ByteOffset Property

Gets the byte offset.

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ByteOffset { get; }
```

Property Value

Type: Int32

The byte offset.

Implements

[IMember.ByteOffset](#) [► 2564]

Reference

[Member Class](#) [► 1781]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.11.1.3 Member.Offset Property

Gets the offset of the [Member](#) [▶ 1781] within the parent [StructType](#) [▶ 1846] in bits or bytes dependent on [IsBitType](#) [▶ 1773]

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Offset { get; }
```

Property Value

Type: Int32
The offset.

Implements

[IMember.Offset](#) [▶ 2564]

Reference





[Member Class](#) [▶ 1781]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.11.2 Member Methods

The [Member](#) [▶ 1781] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString [▶ 1781]	Returns a String that represents this instance. (Inherited from Instance [▶ 1765].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[Member Class](#) [▶ 1781]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.12 PCCHType Class

Class PCCHType. This class cannot be inherited. Implements the [PointerType](#) [▶ 1790]

Inheritance Hierarchy

System.Object

[TwinCAT.Ads.TypeSystem.DataType](#) [▶ 1721]

[TwinCAT.Ads.TypeSystem.PointerType](#) [▶ 1790]

[TwinCAT.Ads.TypeSystem.PCCHType](#)

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229
















Syntax


C#

```
public sealed class PCCHType : PointerType
```






The PCCHType type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the DataType (Inherited from DataType [▶ 1721].)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DataType [▶ 1721].)
	ManagedType [▶ 1793]	Gets the corresponding .NET Type if attached. (Inherited from PointerType [▶ 1790].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	ReferencedType [▶ 1794]	Gets the the referenced type. (Inherited from PointerType [▶ 1790].)
	ReferenceTypeName [▶ 1794]	Gets the name of the referenced datatype (Inherited from PointerType [▶ 1790].)
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)

	Name	Description
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Methods





	Name	Description
	Create [▶ 1790]	Creates this instance.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721].)

Reference

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

[TwinCAT.Ads.TypeSystem.PointerType \[▶ 1790\]](#)









Also see about this









-  [DataType.IsContainer Property \[▶ 1731\]](#)
-  [DataType.IsPointer Property \[▶ 1732\]](#)
-  [DataType.IsPrimitive Property \[▶ 1732\]](#)
-  [DataType.IsReference Property \[▶ 1733\]](#)

6.7.12.1 PCCHType Properties

The [PCCHType \[▶ 1787\]](#) type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the DataType [▶ 1721].
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)





	Name	Description
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DataType [▶ 1721].)
	ManagedType [▶ 1793]	Gets the corresponding .NET Type if attached. (Inherited from PointerType [▶ 1790].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	ReferencedType [▶ 1794]	Gets the the referenced type. (Inherited from PointerType [▶ 1790].)
	ReferenceTypeNam e [▶ 1794]	Gets the name of the referenced datatype (Inherited from PointerType [▶ 1790].)
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Reference

[PCCHType Class](#) [▶ 1787]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]






Also see about this

-  [DataType.IsContainer Property](#) [▶ 1731]
-  [DataType.IsPointer Property](#) [▶ 1732]
-  [DataType.IsPrimitive Property](#) [▶ 1732]
-  [DataType.IsReference Property](#) [▶ 1733]

6.7.12.2 PCCHType Methods

The [PCCHType](#) [▶ 1787] type exposes the following members.

Methods

	Name	Description
	Create [▶ 1790]	Creates this instance.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721].)

Reference

[PCCHType Class](#) [▶ 1787]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.12.2.1 PCCHType.Create Method

Creates this instance.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static PCCHType Create()
```

Return Value

Type: [PCCHType](#) [[▶ 1787](#)]

PCCHType.

Reference

[PCCHType Class](#) [[▶ 1787](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.13 PointerType Class

Represents a pointer type.

Inheritance Hierarchy

System.Object

[TwinCAT.Ads.TypeSystem.DataType](#) [[▶ 1721](#)]

[TwinCAT.Ads.TypeSystem.PointerType](#)

[TwinCAT.Ads.TypeSystem.PCCHType](#) [[▶ 1787](#)]

[TwinCAT.Ads.TypeSystem.PVoidType](#) [[▶ 1801](#)]

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229






Syntax












C#

```
public class PointerType : DataType,
    IPointerType, IDataType, IBitSize
```







The PointerType type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)

	Name	Description
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721] .)
	Id [▶ 1729]	Gets the ID of the DataType [▶ 1721] (Inherited from DataType [▶ 1721] .)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721] .)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DataType [▶ 1721] .)
	ManagedType [▶ 1793]	Gets the corresponding .NET Type if attached. (Overrides DataType.ManagedType [▶ 1733] .)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721] .)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721] .)
	ReferencedType [▶ 1794]	Gets the the referenced type.
	ReferenceTypeName [▶ 1794]	Gets the name of the referenced datatype
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721] .)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721] .)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721] .)

Extension Methods

	Name	Description
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021] .)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021] .)
	IsContainer [▶ 3027]	Overloaded.

	Name	Description
		Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive. [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve. [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]
















Also see about this

- [IDataType.IsContainer Property](#) [[▶ 1731](#)]
- [IDataType.IsPointer Property](#) [[▶ 1732](#)]
- [IDataType.IsPrimitive Property](#) [[▶ 1732](#)]
- [IDataType.IsReference Property](#) [[▶ 1733](#)]

6.7.13.1 PointerType Properties

The [PointerType](#) [[▶ 1790](#)] type exposes the following members.

Properties





	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the DataType [▶ 1721].)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DataType [▶ 1721].)
	ManagedType [▶ 1793]	Gets the corresponding .NET Type if attached. (Overrides DataType.ManagedType [▶ 1733].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	ReferencedType [▶ 1794]	Gets the the referenced type.
	ReferenceTypeName [▶ 1794]	Gets the name of the referenced datatype
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Reference

[PointerType Class \[▶ 1790\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

Also see about this

-  [DataType.IsContainer Property \[▶ 1731\]](#)
-  [DataType.IsPointer Property \[▶ 1732\]](#)
-  [DataType.IsPrimitive Property \[▶ 1732\]](#)
-  [DataType.IsReference Property \[▶ 1733\]](#)

6.7.13.1.1 PointerType.ManagedType Property

Gets the corresponding .NET Type if attached.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override Type? ManagedType { get; }
```

Property Value

Type: Type
Dot net type.

Reference

[PointerType Class](#) [[▶ 1790](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.13.1.2 PointerType.ReferencedType Property

Gets the the referenced type.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataTypes? ReferencedType { get; }
```

Property Value

Type: [IDataTypes](#) [[▶ 2475](#)]
The type of the referenced.

Implements

[IPointerType.ReferencedType](#) [[▶ 2586](#)]

Reference

[PointerType Class](#) [[▶ 1790](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.13.1.3 PointerType.ReferenceTypeName Property

Gets the name of the referenced datatype

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string ReferenceTypeName { get; }
```

Property Value

Type: String
The name of the reference datatype.

Implements

[IPointerType.ReferenceTypeName](#) [[▶ 2587](#)]

Reference







[PointerType Class](#) [[▶ 1790](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.13.2 PointerType Methods

The [PointerType](#) [[▶ 1790](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721].)

Extension Methods

	Name	Description
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive [▶ 3033]	Overloaded.

	Name	Description
		Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[PointerType Class](#) [[▶ 1790](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.14 PrimitiveType Class

Class PrimitiveType.

Inheritance Hierarchy

System.Object

[TwinCAT.Ads.TypeSystem.DataType](#) [[▶ 1721](#)]

[TwinCAT.Ads.TypeSystem.PrimitiveType](#)

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229















Syntax

C#







```
public class PrimitiveType : DataType,
    IPrimitiveType, IDataType, IBitSize
```

The PrimitiveType type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the DataType [▶ 1721].)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DataType [▶ 1721].)
	ManagedType [▶ 1733]	Gets the corresponding .NET Type if attached. (Inherited from DataType [▶ 1721].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	PrimitiveFlags [▶ 1800]	Indicates types of different PrimitiveTypes with flags.
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721].)

Extension Methods

Name	Description
IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)



Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

[TwinCAT.Ads.TypeSystem.DataType](#) [[▶ 1721](#)]

Also see about this
















- [IDataType.IsContainer Property](#) [[▶ 1731](#)]
- [IDataType.IsPointer Property](#) [[▶ 1732](#)]

-  [DataType.IsPrimitive Property \[▶ 1732\]](#)
-  [DataType.IsReference Property \[▶ 1733\]](#)

6.7.14.1 PrimitiveType Properties

The [PrimitiveType \[▶ 1796\]](#) type exposes the following members.

Properties





	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the DataType [▶ 1721].
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DataType [▶ 1721].)
	ManagedType [▶ 1733]	Gets the corresponding .NET Type if attached. (Inherited from DataType [▶ 1721].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	PrimitiveFlags [▶ 1800]	Indicates types of different PrimitiveTypes with flags.
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Reference

[PrimitiveType Class \[▶ 1796\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

Also see about this

-  [DataType.IsContainer Property \[▶ 1731\]](#)
-  [DataType.IsPointer Property \[▶ 1732\]](#)
-  [DataType.IsPrimitive Property \[▶ 1732\]](#)
-  [DataType.IsReference Property \[▶ 1733\]](#)

6.7.14.1 PrimitiveType.PrimitiveFlags Property

Indicates types of different PrimitiveTypes with flags.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public PrimitiveTypeFlags PrimitiveFlags { get; }
```

Property Value

Type: [PrimitiveTypeFlags](#) [[▶ 2812](#)]

The primitive flags.

Implements

[IPrimitiveType.PrimitiveFlags](#) [[▶ 2591](#)]

Reference







[PrimitiveType Class](#) [[▶ 1796](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.14.2 PrimitiveType Methods

The [PrimitiveType](#) [[▶ 1796](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721].)

Extension Methods

	Name	Description
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)

	Name	Description
	IsContainer. [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypesExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypesExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypesExtension [▶ 3021].)
	IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypesExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypesExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypesExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypesExtension [▶ 3021].)
	ResolvableAsPrimitive. [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypesExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypesExtension [▶ 3021].)
	Resolve. [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypesExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypesExtension [▶ 3021].)

Reference

[PrimitiveType Class](#) [[▶ 1796](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.15 PVoidType Class

Class PVoidType. This class cannot be inherited. Implements the [PointerType](#) [[▶ 1790](#)]

Inheritance Hierarchy

System.Object

[TwinCAT.Ads.TypeSystem.DataType](#) [[▶ 1721](#)]

[TwinCAT.Ads.TypeSystem.PointerType](#) [[▶ 1790](#)]

[TwinCAT.Ads.TypeSystem.PVoidType](#)

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#




```
public sealed class PVoidType : PointerType
```


The PVoidType type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the Data Type (Inherited from DataType [▶ 1721].)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DataType [▶ 1721].)
	ManagedType [▶ 1793]	Gets the corresponding .NET Type if attached. (Inherited from PointerType [▶ 1790].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	ReferencedType [▶ 1794]	Gets the the referenced type. (Inherited from PointerType [▶ 1790].)
	ReferenceTypeName [▶ 1794]	Gets the name of the referenced datatype (Inherited from PointerType [▶ 1790].)
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)





	Name	Description
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721].)

Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

[TwinCAT.Ads.TypeSystem.PointerType](#) [[▶ 1790](#)]















Also see about this



-  [DataType.IsContainer Property](#) [[▶ 1731](#)]
-  [DataType.IsPointer Property](#) [[▶ 1732](#)]
-  [DataType.IsPrimitive Property](#) [[▶ 1732](#)]
-  [DataType.IsReference Property](#) [[▶ 1733](#)]

6.7.15.1 PVoidType Properties

The [PVoidType](#) [[▶ 1801](#)] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the DataType (Inherited from DataType [▶ 1721].)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned ($BitSize \% 8 == 0$) (Inherited from DataType [▶ 1721].)
	ManagedType [▶ 1793]	Gets the corresponding .NET Type if attached. (Inherited from PointerType [▶ 1790].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	ReferencedType [▶ 1794]	Gets the the referenced type. (Inherited from PointerType [▶ 1790].)
	ReferenceTypeName [▶ 1794]	Gets the name of the referenced datatype (Inherited from PointerType [▶ 1790].)





	Name	Description
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721] .)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721] .)

Reference

[PVoidType Class \[▶ 1801\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)





Also see about this

-  [DataType.IsContainer Property \[▶ 1731\]](#)
-  [DataType.IsPointer Property \[▶ 1732\]](#)
-  [DataType.IsPrimitive Property \[▶ 1732\]](#)
-  [DataType.IsReference Property \[▶ 1733\]](#)

6.7.15.2 PVoidType Methods

The [PVoidType \[▶ 1801\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721] .)

Reference

[PVoidType Class \[▶ 1801\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.16 ReferenceType Class

Represents a reference type

Inheritance Hierarchy

System.Object

[TwinCAT.Ads.TypeSystem.DataType \[▶ 1721\]](#)

[TwinCAT.Ads.TypeSystem.ReferenceType](#)

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#



```
public sealed class ReferenceType : DataType,
    IReferenceType, IDataType, IBitSize
```



The ReferenceType type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the Data Type (Inherited from DataType [▶ 1721].)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DataType [▶ 1721].)
	ManagedType [▶ 1809]	Gets the corresponding .NET Type if attached. (Overrides DataType.ManagedType [▶ 1733].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	ReferencedType [▶ 1810]	Gets the the referenced type.
	ReferencedTypeNa me [▶ 1810]	Gets the name of the referenced type.
	ResolvedByteSize [▶ 1811]	Gets the size of the resolved byte.
	ResolvedCategory [▶ 1811]	Gets the resolved category.
	ResolvedType [▶ 1811]	Gets the type of the resolved.
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)

	Name	Description
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721].)







Extension Methods

	Name	Description
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

Also see about this



-  [ReferenceType.IsContainer Property \[▶ 1808\]](#)
-  [DataType.IsContainer Property \[▶ 1731\]](#)
-  [DataType.IsPointer Property \[▶ 1732\]](#)
-  [ReferenceType.IsPrimitive Property \[▶ 1809\]](#)
-  [DataType.IsPrimitive Property \[▶ 1732\]](#)
-  [DataType.IsReference Property \[▶ 1733\]](#)

6.7.16.1 ReferenceType Properties

The [ReferenceType \[▶ 1804\]](#) type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the DataType (Inherited from DataType [▶ 1721].)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DataType [▶ 1721].)
	ManagedType [▶ 1809]	Gets the corresponding .NET Type if attached. (Overrides DataType.ManagedType [▶ 1733].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	ReferencedType [▶ 1810]	Gets the the referenced type.
	ReferencedTypeNa me [▶ 1810]	Gets the name of the referenced type.
	ResolvedByteSize [▶ 1811]	Gets the size of the resolved byte.
	ResolvedCategory [▶ 1811]	Gets the resolved category.
	ResolvedType [▶ 1811]	Gets the type of the resolved.







	Name	Description
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721] .)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721] .)

Reference

[ReferenceType Class \[▶ 1804\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

Also see about this

-  [ReferenceType.IsContainer Property \[▶ 1808\]](#)
-  [DataType.IsContainer Property \[▶ 1731\]](#)
-  [DataType.IsPointer Property \[▶ 1732\]](#)
-  [ReferenceType.IsPrimitive Property \[▶ 1809\]](#)
-  [DataType.IsPrimitive Property \[▶ 1732\]](#)
-  [DataType.IsReference Property \[▶ 1733\]](#)

6.7.16.1.1 ReferenceType.IsContainer Property

Gets a value indicating whether this [IDataType \[▶ 2475\]](#) is a container type

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public override bool IsContainer { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is container type; otherwise, false.

Implements

[IDataType.IsContainer \[▶ 2480\]](#)

[IDataType.IsContainer \[▶ 2480\]](#)

Remarks

Container Types are all types that contain SubElements like

- [Array \[▶ 2111\]](#)
- [Pointer \[▶ 2111\]](#)
- [Union \[▶ 2111\]](#)
- [Struct \[▶ 2111\]](#)
- [Function \[▶ 2111\]](#)
- [FunctionBlock \[▶ 2111\]](#)
- [Program \[▶ 2111\]](#)

and the [Alias](#) [▸ 2111] and [Reference](#) [▸ 2111] types, if they have a container type as base type.

Reference

[ReferenceType Class](#) [▸ 1804]

[TwinCAT.Ads.TypeSystem Namespace](#) [▸ 1698]

[IDataType.Category](#) [▸ 2478]

6.7.16.1.2 ReferenceType.IsPrimitive Property

Gets a value indicating whether this [IDataType](#) [▸ 2475] is primitive

Namespace: [TwinCAT.Ads.TypeSystem](#) [▸ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public override bool IsPrimitive { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is primitive; otherwise, false.

Implements

[IDataType.IsPrimitive](#) [▸ 2481]

[IDataType.IsPrimitive](#) [▸ 2481]

Reference

[ReferenceType Class](#) [▸ 1804]

[TwinCAT.Ads.TypeSystem Namespace](#) [▸ 1698]

6.7.16.1.3 ReferenceType.ManagedType Property

Gets the corresponding .NET Type if attached.

Namespace: [TwinCAT.Ads.TypeSystem](#) [▸ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override Type? ManagedType { get; }
```

Property Value

Type: [Type](#)

Dot net type.

Reference

[ReferenceType Class](#) [▸ 1804]

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.16.1.4 ReferenceType.ReferencedType Property

Gets the the referenced type.

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataTypeInfo ReferencedType { get; }
```

Property Value

Type: [IDataTypeInfo \[► 2475\]](#)

The type of the referenced.

Implements

[IReferenceType.ReferencedType \[► 2603\]](#)

Reference

[ReferenceType Class \[► 1804\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.16.1.5 ReferenceType.ReferencedTypeName Property

Gets the name of the referenced type.

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string ReferencedTypeName { get; }
```

Property Value

Type: String

The name of the referenced type.

Implements

[IReferenceType.ReferencedTypeName \[► 2604\]](#)

Reference

[ReferenceType Class \[► 1804\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.16.1.6 ReferenceType.ResolvedByteSize Property

Gets the size of the resolved byte.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ResolvedByteSize { get; }
```

Property Value

Type: Int32

The size of the resolved byte.

Implements

[IReferenceType.ResolvedByteSize](#) [[▶ 2604](#)]

Reference

[ReferenceType Class](#) [[▶ 1804](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.16.1.7 ReferenceType.ResolvedCategory Property

Gets the resolved category.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DataTypeCategory ResolvedCategory { get; }
```

Property Value

Type: [DataTypeCategory](#) [[▶ 2111](#)]

The resolved category.

Implements

[IReferenceType.ResolvedCategory](#) [[▶ 2604](#)]

Reference

[ReferenceType Class](#) [[▶ 1804](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.16.1.8 ReferenceType.ResolvedType Property

Gets the type of the resolved.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataType ResolvedType { get; }
```

Property Value

Type: [IDataType](#) [[▶ 2475](#)]

The type of the resolved.

Implements

[IReferenceType.ResolvedType](#) [[▶ 2605](#)]

Reference





[ReferenceType Class](#) [[▶ 1804](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.16.2 ReferenceType Methods

The [ReferenceType](#) [[▶ 1804](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721].)

Extension Methods

	Name	Description
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)

Name	Description
IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive . [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
Resolve . [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[ReferenceType Class](#) [[▶ 1804](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.17 RpcMethod Class

RPC Method Description

Inheritance Hierarchy

System.Object

 TwinCAT.Ads.TypeSystem.RpcMethod

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#











```
public class RpcMethod : IRpcMethod
```

The `RpcMethod` type exposes the following members.





Constructors



	Name	Description
	RpcMethod(String) [▶ 1822]	Initializes a new instance of the RpcMethod class.
	RpcMethod(String, RpcMethodParameterCollection) [▶ 1822]	Initializes a new instance of the RpcMethod class.
	RpcMethod(String, RpcMethodParameterCollection, IDataTypes) [▶ 1823]	Initializes a new instance of the RpcMethod class.
	RpcMethod(String, RpcMethodParameterCollection, IDataTypes, String) [▶ 1824]	Initializes a new instance of the RpcMethod class.

Properties

	Name	Description
	Comment [▶ 1816]	Gets the Method comment.
	InParameters [▶ 1816]	Gets the In-Parameters of the IRpcMethod [▶ 2625]
	IsVoid [▶ 1817]	Gets a value indicating whether this IRpcMethod [▶ 2625] has no return parameter
	Name [▶ 1817]	Gets the name of the method
	OutParameters [▶ 1818]	Gets the Out-Parameters of the IRpcMethod [▶ 2625]
	Parameters [▶ 1818]	Gets all parameters (In, Out and ref parameters) of the IRpcMethod [▶ 2625]
	ReturnAlignSize [▶ 1819]	Gets the size of the biggest element in bytes for Alignment
	ReturnType [▶ 1819]	Gets the return type.
	ReturnTypeSize [▶ 1819]	Gets the Byte size of the return type.
	VTableIndex [▶ 1820]	Gets the V-table index of the method.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)

	Name	Description
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString [▶ 1821]	Returns a String that represents this instance. (Overrides Object.ToString..)

Extension Methods

	Name	Description
		Overloaded.
	AddParameter(IRpcMethodParameter) [▶ 1949]	Adds a parameter to the RpcMethod (Defined by FluentRpcMethodExtension [▶ 1947].)
	AddParameter(String, IDataTypeInfo, MethodParamFlags) [▶ 1950]	Overloaded. Adds a parameter to the RpcMethod (Defined by FluentRpcMethodExtension [▶ 1947].)
	AddParameter(String, IDataTypeInfo, MethodParamFlags, Int32) [▶ 1951]	Overloaded. Adds a parameter to the RpcMethod (Defined by FluentRpcMethodExtension [▶ 1947].)
	SetReturnType [▶ 1952]	Sets the ReturnType of the RpcMethod. (Defined by FluentRpcMethodExtension [▶ 1947].)










Reference


[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.17.1 RpcMethod Properties

The [RpcMethod](#) [[▶ 1813](#)] type exposes the following members.

Properties

	Name	Description
	Comment [▶ 1816]	Gets the Method comment.
	InParameters [▶ 1816]	Gets the In-Parameters of the IRpcMethod [▶ 2625]
	IsVoid [▶ 1817]	Gets a value indicating whether this IRpcMethod [▶ 2625] has no return parameter
	Name [▶ 1817]	Gets the name of the method
	OutParameters [▶ 1818]	Gets the Out-Parameters of the IRpcMethod [▶ 2625]
	Parameters [▶ 1818]	Gets all parameters (In, Out and ref parameters) of the . [▶ 2625]
	ReturnAlignSize [▶ 1819]	Gets the size of the biggest element in bytes for Alignment
	ReturnType [▶ 1819]	Gets the return type.
	ReturnTypeSize [▶ 1819]	Gets the Byte size of the return type.

	Name	Description
	VTableIndex [▶ 1820]	Gets the V-table index of the method.

Reference

[RpcMethod Class](#) [[▶ 1813](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.17.1.1 RpcMethod.Comment Property

Gets the Method comment.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Comment { get; }
```

Property Value

Type: String
The comment.

Implements

[IRpcMethod.Comment](#) [[▶ 2626](#)]

Reference

[RpcMethod Class](#) [[▶ 1813](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.17.1.2 RpcMethod.InParameters Property

Gets the In-Parameters of the [IRpcMethod](#) [[▶ 2625](#)]

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IRpcMethodParameterCollection InParameters { get; }
```

Property Value

Type: [IRpcMethodParameterCollection](#) [[▶ 2639](#)]
The In- and Ref-Parameters

Implements

[IRpcMethod.InParameters](#) [[▶ 2626](#)]

Reference

[RpcMethod Class](#) [[▶ 1813](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.17.1.3 RpcMethod.IsVoid Property

Gets a value indicating whether this [IRpcMethod](#) [[▶ 2625](#)] has no return parameter

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsVoid { get; }
```

Property Value

Type: Boolean

true if this instance is void; otherwise, false.

Implements

[IRpcMethod.IsVoid](#) [[▶ 2627](#)]

Exceptions

Exception	Condition
NotImplementedException	

Reference

[RpcMethod Class](#) [[▶ 1813](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.17.1.4 RpcMethod.Name Property

Gets the name of the method

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Name { get; }
```

Property Value

Type: String

The name.

Implements

[IRpcMethod.Name](#) [[▶ 2627](#)]

Reference

[RpcMethod Class](#) [► 1813]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.17.1.5 RpcMethod.OutParameters Property

Gets the Out-Parameters of the [IRpcMethod](#) [► 2625]

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IRpcMethodParameterCollection OutParameters { get; }
```

Property Value

Type: [IRpcMethodParameterCollection](#) [► 2639]

The In- and Ref-Parameters

Implements

[IRpcMethod.OutParameters](#) [► 2628]

Reference

[RpcMethod Class](#) [► 1813]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.17.1.6 RpcMethod.Parameters Property

Gets all parameters (In, Out and ref parameters) of the [IRpcMethod](#) [► 2625]

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IRpcMethodParameterCollection Parameters { get; }
```

Property Value

Type: [IRpcMethodParameterCollection](#) [► 2639]

The parameters.

Implements

[IRpcMethod.Parameters](#) [► 2628]

Reference

[RpcMethod Class](#) [► 1813]

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.17.1.7 RpcMethod.ReturnAlignSize Property

Gets the size of the biggest element in bytes for Alignment

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ReturnAlignSize { get; }
```

Property Value

Type: Int32

The size of the return align.

Reference

[RpcMethod Class \[▸ 1813\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.17.1.8 RpcMethod.ReturnType Property

Gets the return type.

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string ReturnType { get; }
```

Property Value

Type: String

Return type.

Implements

[IRpcMethod.ReturnType \[▸ 2628\]](#)

Reference

[RpcMethod Class \[▸ 1813\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.17.1.9 RpcMethod.ReturnTypeSize Property

Gets the Byte size of the return type.

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ReturnTypeInfoSize { get; }
```

Property Value

Type: Int32

The size of the return type.

Implements

[IRpcMethod.ReturnTypeInfoSize](#) [[▶ 2629](#)]

Reference

[RpcMethod Class](#) [[▶ 1813](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.17.1.10 RpcMethod.VTableIndex Property

Gets the V-table index of the method.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int VTableIndex { get; }
```

Property Value

Type: Int32

The index of the v table.

Reference





[RpcMethod Class](#) [[▶ 1813](#)]



[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.17.2 RpcMethod Methods

The [RpcMethod](#) [[▶ 1813](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)

	Name	Description
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString [▶ 1821]	Returns a String that represents this instance. (Overrides Object.ToString..)

Extension Methods

	Name	Description
		Overloaded.
	AddParameter(IRpcMethodParameter) [▶ 1949]	Adds a parameter to the RpcMethod (Defined by FluentRpcMethodExtension [▶ 1947].)
	AddParameter(String, IDataType, MethodParamFlags) [▶ 1950]	Overloaded. Adds a parameter to the RpcMethod (Defined by FluentRpcMethodExtension [▶ 1947].)
	AddParameter(String, IDataType, MethodParamFlags, Int32) [▶ 1951]	Overloaded. Adds a parameter to the RpcMethod (Defined by FluentRpcMethodExtension [▶ 1947].)
	SetReturnType [▶ 1952]	Sets the ReturnType of the RpcMethod. (Defined by FluentRpcMethodExtension [▶ 1947].)

Reference

[RpcMethod Class](#) [[▶ 1813](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.17.2.1 RpcMethod.ToString Method

Returns a String that represents this instance.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```

Return Value

Type: String

A String that represents this instance.

Reference

[RpcMethod Class](#) [[▶ 1813](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.17.3 RpcMethod Constructor

Overload List

	Name	Description
	RpcMethod(String) [▸ 1822]	Initializes a new instance of the RpcMethod [▸ 1813] class.
	RpcMethod(String, RpcMethodParameterCollection) [▸ 1822]	Initializes a new instance of the RpcMethod [▸ 1813] class.
	RpcMethod(String, RpcMethodParameterCollection, IDataType) [▸ 1823]	Initializes a new instance of the RpcMethod [▸ 1813] class.
	RpcMethod(String, RpcMethodParameterCollection, IDataType, String) [▸ 1824]	Initializes a new instance of the RpcMethod [▸ 1813] class.

Reference

[RpcMethod Class](#) [[▸ 1813](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▸ 1698](#)]

6.7.17.3.1 RpcMethod Constructor (String)

Initializes a new instance of the [RpcMethod](#) [[▸ 1813](#)] class.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▸ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RpcMethod(
    string name
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Reference

[RpcMethod Class](#) [[▸ 1813](#)]

[RpcMethod Overload](#) [[▸ 1822](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▸ 1698](#)]

6.7.17.3.2 RpcMethod Constructor (String, RpcMethodParameterCollection)

Initializes a new instance of the [RpcMethod](#) [[▸ 1813](#)] class.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RpcMethod(
    string name,
    RpcMethodParameterCollection? parameters
)
```

Parameters

name	Type: System.String The name.
parameters	Type: TwinCAT.TypeSystem.RpcMethodParameterCollection [▶ 2915] The parameters.

Reference

- [RpcMethod Class](#) [[▶ 1813](#)]
- [RpcMethod Overload](#) [[▶ 1822](#)]
- [TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.17.3.3 RpcMethod Constructor (String, RpcMethodParameterCollection, IDataTypes)

Initializes a new instance of the [RpcMethod](#) [[▶ 1813](#)] class.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RpcMethod(
    string name,
    RpcMethodParameterCollection? parameters,
    IDataTypes? returnType
)
```

Parameters

name	Type: System.String The name.
parameters	Type: TwinCAT.TypeSystem.RpcMethodParameterCollection [▶ 2915] The parameters.
returnType	Type: TwinCAT.TypeSystem.IDataTypes [▶ 2475] Type of the return.

Reference

- [RpcMethod Class](#) [[▶ 1813](#)]
- [RpcMethod Overload](#) [[▶ 1822](#)]
- [TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.17.3.4 RpcMethod Constructor (String, RpcMethodParameterCollection, IDataTypes, String)

Initializes a new instance of the [RpcMethod](#) [▸ 1813] class.

Namespace: [TwinCAT.Ads.TypeSystem](#) [▸ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RpcMethod(
    string name,
    RpcMethodParameterCollection? parameters,
    IDataTypes? returnType,
    string? comment
)
```

Parameters

name	Type: System.String The name.
parameters	Type: TwinCAT.TypeSystem.RpcMethodParameterCollection [▸ 2915] The parameters.
returnType	Type: TwinCAT.TypeSystem.IDataTypes [▸ 2475] Type of the return.
comment	Type: System.String The comment.

Reference

[RpcMethod Class](#) [▸ 1813]

[RpcMethod Overload](#) [▸ 1822]

[TwinCAT.Ads.TypeSystem Namespace](#) [▸ 1698]

6.7.18 RpcMethodParameter Class

Class [RpcMethodParameter](#).

Inheritance Hierarchy

System.Object
 TwinCAT.Ads.TypeSystem.RpcMethodParameter

Namespace: [TwinCAT.Ads.TypeSystem](#) [▸ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#












```
public class RpcMethodParameter : IRpcMethodParameter
```

The [RpcMethodParameter](#) type exposes the following members.




Constructors




	Name	Description
	RpcMethodParameter(String, IDataTypeInfo, MethodParamFlags) [▶ 1833]	Initializes a new instance of the RpcMethodParameter class.
	RpcMethodParameter(String, IDataTypeInfo, MethodParamFlags, Int32) [▶ 1833]	Initializes a new instance of the RpcMethodParameter class.
	RpcMethodParameter(String, IDataTypeInfo, MethodParamFlags, Int32, String) [▶ 1834]	Initializes a new instance of the RpcMethodParameter class.

Properties

	Name	Description
	AlignSize [▶ 1826]	Gets the size of biggest element for alignment
	Comment [▶ 1827]	Gets the Parameter Comment.
	HasLengthsParameter [▶ 1827]	Gets a value indicating whether this instance has a related Lengths Parameter.
	IsInput [▶ 1830]	Gets a value indicating whether this instance is input.
	IsOutput [▶ 1831]	Gets a value indicating whether this instance is output.
	LengthsParameterIndex [▶ 1828]	Gets the index of the Lengths parameter (within the MethodParameter List)
	Name [▶ 1828]	Gets the Parameter Name
	ParameterFlags [▶ 1829]	Gets the parameter flags.
	Size [▶ 1829]	Gets the size of the RpcMethodParameter
	TypeGuid [▶ 1830]	Gets the Unique identifier of the parameters data type.
	TypeName [▶ 1830]	Gets the Data type of the Parameter

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)

	Name	Description
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString [▶ 1832]	Returns a String that represents this instance. (Overrides Object.ToString..)












Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.18.1 RpcMethodParameter Properties

The [RpcMethodParameter](#) [[▶ 1824](#)] type exposes the following members.

Properties

	Name	Description
	AlignSize [▶ 1826]	Gets the size of biggest element for alignment
	Comment [▶ 1827]	Gets the Parameter Comment.
	HasLengthsParameter [▶ 1827]	Gets a value indicating whether this instance has a related Lengths Parameter.
	IsInput [▶ 1830]	Gets a value indicating whether this instance is input.
	IsOutput [▶ 1831]	Gets a value indicating whether this instance is output.
	LengthsParameterIndex [▶ 1828]	Gets the index of the Lengths parameter (within the MethodParameter List)
	Name [▶ 1828]	Gets the Parameter Name
	ParameterFlags [▶ 1829]	Gets the parameter flags.
	Size [▶ 1829]	Gets the size of the RpcMethodParameter [▶ 1824]
	TypeGuid [▶ 1830]	Gets the Unique identifier of the parameters data type.
	TypeName [▶ 1830]	Gets the Data type of the Parameter

Reference

[RpcMethodParameter Class](#) [[▶ 1824](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.18.1.1 RpcMethodParameter.AlignSize Property

Gets the size of biggest element for alignment

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int AlignSize { get; }
```

Property Value

Type: Int32

The size of the align.

Reference

[RpcMethodParameter Class \[▸ 1824\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.18.1.2 RpcMethodParameter.Comment Property

Gets the Parameter Comment.

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Comment { get; }
```

Property Value

Type: String

The comment.

Reference

[RpcMethodParameter Class \[▸ 1824\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.18.1.3 RpcMethodParameter.HasLengthsParameter Property

Gets a value indicating whether this instance has a related Lengths Parameter.

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool HasLengthIsParameter { get; }
```

Property Value

Type: Boolean

true if this instance has a Lengths parameter; otherwise, false.

Implements

[IRpcMethodParameter.HasLengthIsParameter \[▸ 2636\]](#)

Reference

[RpcMethodParameter Class](#) [► 1824]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.18.1.4 RpcMethodParameter.LengthIsParameterIndex Property

Gets the index of the LengthIs parameter (within the MethodParameter List)

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int LengthIsParameterIndex { get; }
```

Property Value

Type: Int32

The index of the length is parameter.

Implements

[IRpcMethodParameter.LengthIsParameterIndex](#) [► 2637]

Remarks

This field references to the Parameter that defines the length for this generic one. Equally to the marshalling attributes of COM (sizeof, length) this enables to transport parameter of type (PVOID)

Reference

[RpcMethodParameter Class](#) [► 1824]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.18.1.5 RpcMethodParameter.Name Property

Gets the Parameter Name

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Name { get; }
```

Property Value

Type: String

The name.

Implements

[IRpcMethodParameter.Name](#) [► 2637]

Reference

[RpcMethodParameter Class](#) [[▶ 1824](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.18.1.6 RpcMethodParameter.ParameterFlags Property

Gets the parameter flags.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public MethodParamFlags ParameterFlags { get; }
```

Property Value

Type: [MethodParamFlags](#) [[▶ 2812](#)]

The parameter flags.

Implements

[IRpcMethodParameter.ParameterFlags](#) [[▶ 2637](#)]

Reference

[RpcMethodParameter Class](#) [[▶ 1824](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.18.1.7 RpcMethodParameter.Size Property

Gets the size of the [RpcMethodParameter](#) [[▶ 1824](#)]

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Size { get; }
```

Property Value

Type: Int32

The size.

Implements

[IRpcMethodParameter.Size](#) [[▶ 2638](#)]

Reference

[RpcMethodParameter Class](#) [[▶ 1824](#)]

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.18.1.8 RpcMethodParameter.TypeGuid Property

Gets the Unique identifier of the parameters data type.

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Guid TypeGuid { get; }
```

Property Value

Type: Guid

The type unique identifier.

Reference

[RpcMethodParameter Class \[▸ 1824\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.18.1.9 RpcMethodParameter.TypeName Property

Gets the Data type of the Parameter

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string TypeName { get; }
```

Property Value

Type: String

The type.

Implements

[IRpcMethodParameter.TypeName \[▸ 2638\]](#)

Reference

[RpcMethodParameter Class \[▸ 1824\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.18.1.10 RpcMethodParameter.IsInput Property

Gets a value indicating whether this instance is input.

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsInput { get; }
```

Property Value

Type: Boolean
true if this instance is input; otherwise, false.

Implements

[IRpcMethodParameter.IsInput](#) [[▶ 2639](#)]

Reference

[RpcMethodParameter Class](#) [[▶ 1824](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.18.1.11 RpcMethodParameter.IsOutput Property

Gets a value indicating whether this instance is output.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsOutput { get; }
```

Property Value

Type: Boolean
true if this instance is output; otherwise, false.

Implements

[IRpcMethodParameter.IsOutput](#) [[▶ 2639](#)]

Reference


[RpcMethodParameter Class](#) [[▶ 1824](#)]






[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.18.2 RpcMethodParameter Methods

The [RpcMethodParameter](#) [[▶ 1824](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)

	Name	Description
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString [▶ 1832]	Returns a String that represents this instance. (Overrides Object.ToString..)

Reference

[RpcMethodParameter Class](#) [[▶ 1824](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.18.2.1 RpcMethodParameter.ToString Method

Returns a String that represents this instance.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```

Return Value

Type: String

A String that represents this instance.

Reference

[RpcMethodParameter Class](#) [[▶ 1824](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.18.3 RpcMethodParameter Constructor

Overload List

	Name	Description
	RpcMethodParameter (String, IDataTypes, MethodParamFlags) [▶ 1833]	Initializes a new instance of the RpcMethodParameter [▶ 1824] class.
	RpcMethodParameter (String, IDataTypes, MethodParamFlags, Int32) [▶ 1833]	Initializes a new instance of the RpcMethodParameter [▶ 1824] class.

	Name	Description
	RpcMethodParameter(String, IDataTypes, MethodParamFlags, Int32, String) [▸ 1834]	Initializes a new instance of the RpcMethodParameter [▸ 1824] class.

Reference

[RpcMethodParameter Class \[▸ 1824\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.18.3.1 RpcMethodParameter Constructor (String, IDataTypes, MethodParamFlags)

Initializes a new instance of the [RpcMethodParameter \[▸ 1824\]](#) class.

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RpcMethodParameter(
    string name,
    IDataTypes type,
    MethodParamFlags flags
)
```

Parameters

name	Type: System.String The name.
type	Type: TwinCAT.TypeSystem.IDataTypes [▸ 2475] The type.
flags	Type: TwinCAT.TypeSystem.MethodParamFlags [▸ 2812] The flags.

Reference

[RpcMethodParameter Class \[▸ 1824\]](#)

[RpcMethodParameter Overload \[▸ 1832\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.18.3.2 RpcMethodParameter Constructor (String, IDataTypes, MethodParamFlags, Int32)

Initializes a new instance of the [RpcMethodParameter \[▸ 1824\]](#) class.

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RpcMethodParameter(
    string name,
    IDataType type,
    MethodParamFlags flags,
    int lengthIsIndex
)
```

Parameters

name	Type: System.String The name.
type	Type: TwinCAT.TypeSystem.IDataType [▶ 2475] The type.
flags	Type: TwinCAT.TypeSystem.MethodParamFlags [▶ 2812] The flags.
lengthIsIndex	Type: System.Int32 Index of the length is.

Reference

[RpcMethodParameter Class](#) [▶ 1824]

[RpcMethodParameter Overload](#) [▶ 1832]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.18.3 RpcMethodParameter Constructor (String, IDataType, MethodParamFlags, Int32, String)

Initializes a new instance of the [RpcMethodParameter](#) [▶ 1824] class.

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RpcMethodParameter(
    string name,
    IDataType type,
    MethodParamFlags flags,
    int lengthIsIndex,
    string? comment
)
```

Parameters

name	Type: System.String The name.
type	Type: TwinCAT.TypeSystem.IDataType [▶ 2475] The type.
flags	Type: TwinCAT.TypeSystem.MethodParamFlags [▶ 2812] The flags.
lengthIsIndex	Type: System.Int32 Index of the length is.
comment	Type: System.String The comment.

Reference

- [RpcMethodParameter Class \[▶ 1824\]](#)
- [RpcMethodParameter Overload \[▶ 1832\]](#)
- [TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.19 RpcStructType Class

StructType which is callable by RPC Methods.

Inheritance Hierarchy

- [System.Object](#)
- [TwinCAT.Ads.TypeSystem.DataType \[▶ 1721\]](#)
- [TwinCAT.Ads.TypeSystem.StructType \[▶ 1846\]](#)
- [TwinCAT.Ads.TypeSystem.RpcStructType](#)

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5












Syntax












C#

```
public sealed class RpcStructType : StructType,
    IRpcStructType, IRpcCallableType, IStructType, IInterfaceType, IDataType,
    IBitSize
```





The RpcStructType type exposes the following members.

Properties


	Name	Description
	AllMembers [▶ 1851]	Gets all members (down the derivation hierarchy) (Inherited from StructType [▶ 1846] .)
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721] .)
	BaseType [▶ 1851]	Gets the structs Base Type (Null if not derived). (Inherited from StructType [▶ 1846] .)
	BaseTypeName [▶ 1851]	Gets the the Name of the Base class (if derived) (Inherited from StructType [▶ 1846] .)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721] .)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721] .)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721] .)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721] .)
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721] .)
	HasRpcMethods [▶ 1838]	Gets a value indicating whether this instance has RPC Methods. (Overrides StructType.HasRpcMethods [▶ 1852] .)
	Id [▶ 1729]	Gets the ID of the DataType (Inherited from DataType [▶ 1721] .)



	Name	Description
	InterfaceImplementationNames [▶ 1853]	Gets the names of the interfaces, this IDataType [▶ 2475] implements. (Inherited from StructType [▶ 1846].)
	InterfaceImplementations [▶ 1854]	Gets the resolved interface types, this IDataType [▶ 2475] implments. (Inherited from StructType [▶ 1846].)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DataType [▶ 1721].)
	IsContainer [▶ 1731]	Gets a value indicating whether this IDataType [▶ 2475] is a container type (Inherited from DataType [▶ 1721].)
	IsDerived [▶ 1852]	Gets a value indicating whether this instance is derived. (Inherited from StructType [▶ 1846].)
	IsPointer [▶ 1732]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Inherited from DataType [▶ 1721].)
	IsPrimitive [▶ 1732]	Gets a value indicating whether this IDataType [▶ 2475] is primitive (Inherited from DataType [▶ 1721].)
	IsReference [▶ 1733]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Inherited from DataType [▶ 1721].)
	ManagedType [▶ 1733]	Gets the corresponding .NET Type if attached. (Inherited from DataType [▶ 1721].)
	Members [▶ 1853]	Gets a read only collection of the Members [▶ 2561] of the IStructType [▶ 2671]. (Inherited from StructType [▶ 1846].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	RpcMethods [▶ 1839]	Gets the Method descriptions for the IRpcCallableType [▶ 2624]
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721].)

Extension Methods

	Name	Description
	AddAligned(IMember) [▶ 1953]	Overloaded. Adds a member (Defined by FluentRpcStructTypeExtension [▶ 1952].)

	Name	Description
	AddAligned(IMember) [▶ 1956]	Overloaded. Adds a member to the StructType [▶ 1846] (Defined by FluentStructTypeExtension [▶ 1955] .)
	AddMethod [▶ 1954]	Adds a RpcMethod. (Defined by FluentRpcStructTypeExtension [▶ 1952] .)







Reference











[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.19.1 RpcStructType Properties

The [RpcStructType \[▶ 1835\]](#) type exposes the following members.

Properties

	Name	Description
	AllMembers [▶ 1851]	Gets all members (down the derivation hierarchy) (Inherited from StructType [▶ 1846] .)
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721] .)
	BaseType [▶ 1851]	Gets the structs Base Type (Null if not derived). (Inherited from StructType [▶ 1846] .)
	BaseTypeName [▶ 1851]	Gets the the Name of the Base class (if derived) (Inherited from StructType [▶ 1846] .)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721] .)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721] .)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721] .)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721] .)
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721] .)
	HasRpcMethods [▶ 1838]	Gets a value indicating whether this instance has RPC Methods. (Overrides StructType.HasRpcMethods [▶ 1852] .)
	Id [▶ 1729]	Gets the ID of the DataType (Inherited from DataType [▶ 1721] .)
	InterfaceImplementationNames [▶ 1853]	Gets the names of the interfaces, this IDataType [▶ 2475] implements. (Inherited from StructType [▶ 1846] .)
	InterfaceImplementations [▶ 1854]	Gets the resolved interface types, this IDataType [▶ 2475] implments. (Inherited from StructType [▶ 1846] .)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721] .)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DataType [▶ 1721] .)
	IsContainer [▶ 1731]	Gets a value indicating whether this IDataType [▶ 2475] is a container type (Inherited from DataType [▶ 1721] .)

	Name	Description
	IsDerived [▶ 1852]	Gets a value indicating whether this instance is derived. (Inherited from StructType [▶ 1846].)
	IsPointer [▶ 1732]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Inherited from DataType [▶ 1721].)
	IsPrimitive [▶ 1732]	Gets a value indicating whether this IDataType [▶ 2475] is primitive (Inherited from DataType [▶ 1721].)
	IsReference [▶ 1733]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Inherited from DataType [▶ 1721].)
	ManagedType [▶ 1733]	Gets the corresponding .NET Type if attached. (Inherited from DataType [▶ 1721].)
	Members [▶ 1853]	Gets a read only collection of the Members [▶ 2561] of the IStructType [▶ 2671]. (Inherited from StructType [▶ 1846].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	RpcMethods [▶ 1839]	Gets the Method descriptions for the IRpcCallableType [▶ 2624]
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)

Reference

[RpcStructType Class](#) [[▶ 1835](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.19.1.1 RpcStructType.HasRpcMethods Property

Gets a value indicating whether this instance has RPC Methods.

Namespace: TwinCAT.Ads.TypeSystem

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public override bool HasRpcMethods { get; }
```

Property Value

Type: [Boolean](#)

true if this instance has methods; otherwise, false.

Implements

[InterfaceType.HasRpcMethods](#) [[▶ 2986](#)]

[InterfaceType.HasRpcMethods](#) [[▶ 2986](#)]

Remarks

The [DataType](#) (Structure) must be marked with the [PlcAttribute](#) 'TcRpcEnable' to enable [RpcMethods](#), otherwise [RpcMethods](#) are not passed through to the ADS symbolic information.

Reference

[RpcStructType Class \[▶ 1835\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.19.1.2 RpcStructType.RpcMethods Property

Gets the Method descriptions for the [IRpcCallableType \[▶ 2624\]](#)

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public IRpcMethodCollection RpcMethods { get; }
```

Property Value

Type: [IRpcMethodCollection \[▶ 2629\]](#)

The methods.

Implements

[IRpcCallableType.RpcMethods \[▶ 2624\]](#)

Remarks

The DataType (Structure) must be marked with the PlcAttribute 'TcRpcEnable' to enable RpcMethods, otherwise RpcMethods are not passed through to the ADS symbolic information.

Reference





[RpcStructType Class \[▶ 1835\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)




6.7.19.2 RpcStructType Methods

The [RpcStructType \[▶ 1835\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721] .)

Extension Methods

	Name	Description
	AddAligned(IMember) [▸ 1953]	Overloaded. Adds a member (Defined by FluentRpcStructTypeExtension [▸ 1952].)
	AddAligned(IMember) [▸ 1956]	Overloaded. Adds a member to the StructType [▸ 1846] (Defined by FluentStructTypeExtension [▸ 1955].)
	AddMethod [▸ 1954]	Adds a RpcMethod. (Defined by FluentRpcStructTypeExtension [▸ 1952].)

Reference

[RpcStructType Class \[▸ 1835\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.20 StringType Class

String DataType

Inheritance Hierarchy

System.Object

[TwinCAT.Ads.TypeSystem.DataType \[▸ 1721\]](#)

[TwinCAT.Ads.TypeSystem.StringType](#)

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229









Syntax










C#

```
public sealed class StringType : DataType,
    IStringType, IDataType, IBitSize
```





The StringType type exposes the following members.

Properties

	Name	Description
	Attributes [▸ 1727]	Gets the attributes of the IDataType [▸ 2475] (Inherited from DataType [▸ 1721].)
	BitSize [▸ 1727]	Gets the size of the DataType [▸ 1721] in bits. (Inherited from DataType [▸ 1721].)
	ByteSize [▸ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▸ 1721].)
	Category [▸ 1728]	Gets the Data Type category (Inherited from DataType [▸ 1721].)
	Comment [▸ 1729]	Gets the comment. (Inherited from DataType [▸ 1721].)
	Encoding [▸ 1843]	Gets the encoding of the String (Encoding.Default (Ansi Codepage, STRING) or Encoding.UNICODE (WSTRING))
	FullName [▸ 1729]	Gets the full name of the IDataType [▸ 2475] (Namespace + Name) (Inherited from DataType [▸ 1721].)
	Id [▸ 1729]	Gets the ID of the DataType (Inherited from DataType [▸ 1721].)

	Name	Description
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DataType [▶ 1721].)
	IsFixedLength [▶ 1844]	Gets a value indicating whether the string is of fixed length.
	Length [▶ 1844]	Gets the number of characters within the string.
	ManagedType [▶ 1733]	Gets the corresponding .NET Type if attached. (Inherited from DataType [▶ 1721].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString [▶ 1846]	Returns a String that represents this instance. (Overrides DataType.ToString . [▶ 1738].)

Extension Methods





	Name	Description
	IsArrayOfPrimitives . [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer . [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive . [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)

	Name	Description
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve . [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]







Also see about this












-  [DataType.IsContainer Property](#) [▶ 1731]
-  [DataType.IsPointer Property](#) [▶ 1732]
-  [DataType.IsPrimitive Property](#) [▶ 1732]
-  [DataType.IsReference Property](#) [▶ 1733]

6.7.20.1 StringType Properties

The [StringType](#) [▶ 1840] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	Encoding [▶ 1843]	Gets the encoding of the String (Encoding.Default (Ansi Codepage, STRING) or Encoding.UNICODE (WSTRING))





	Name	Description
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the DataType (Inherited from DataType [▶ 1721].)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DataType [▶ 1721].)
	IsFixedLength [▶ 1844]	Gets a value indicating whether the string is of fixed length.
	Length [▶ 1844]	Gets the number of characters within the string.
	ManagedType [▶ 1733]	Gets the corresponding .NET Type if attached. (Inherited from DataType [▶ 1721].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Reference

[StringType Class](#) [[▶ 1840](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

Also see about this

-  [DataType.IsContainer Property](#) [[▶ 1731](#)]
-  [DataType.IsPointer Property](#) [[▶ 1732](#)]
-  [DataType.IsPrimitive Property](#) [[▶ 1732](#)]
-  [DataType.IsReference Property](#) [[▶ 1733](#)]

6.7.20.1.1 StringType.Encoding Property

Gets the encoding of the String (Encoding.Default (Ansi Codepage, STRING) or Encoding.UNICODE (WSTRING))

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Encoding Encoding { get; }
```

Property Value

Type: Encoding
The encoding.

Implements

[IStringType.Encoding](#) [► 2664]

Reference

[StringType Class](#) [► 1840]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.20.1.2 StringType.IsFixedLength Property

Gets a value indicating whether the string is of fixed length.

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsFixedLength { get; }
```

Property Value

Type: Boolean

true if this instance is fixed length; otherwise, false.

Implements

[IStringType.IsFixedLength](#) [► 2665]

Exceptions

Exception	Condition
NotImplementedException	

Reference

[StringType Class](#) [► 1840]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.20.1.3 StringType.Length Property

Gets the number of characters within the string.

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Length { get; }
```

Property Value

Type: Int32

The length.

Implements

[IStringType.Length](#) [[▶ 2665](#)]

Reference





[StringType Class](#) [[▶ 1840](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.20.2 StringType Methods

The [StringType](#) [[▶ 1840](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString [▶ 1846]	Returns a String that represents this instance. (Overrides DataType.ToString . [▶ 1738].)

Extension Methods

	Name	Description
	IsArrayOfPrimitives . [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer . [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive . [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)

	Name	Description
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[StringType Class](#) [[▶ 1840](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.20.2.1 StringType.ToString Method

Returns a String that represents this instance.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```

Return Value

Type: String

A String that represents this instance.

Reference

[StringType Class](#) [[▶ 1840](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.21 StructType Class

Represents a struct type

Inheritance Hierarchy

System.Object

 TwinCAT.Ads.TypeSystem.DataType [[▶ 1721](#)]

 TwinCAT.Ads.TypeSystem.StructType

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229







Syntax







C#

```
public class StructType : DataType, IStructType,
    IInterfaceType, IDataType, IBitSize, IRpcCallableType, IRpcStructType
```







The StructType type exposes the following members.

Properties

	Name	Description
	AllMembers [▶ 1851]	Gets all members (down the derivation hierarchy)
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BaseType [▶ 1851]	Gets the structs Base Type (Null if not derived).
	BaseTypeName [▶ 1851]	Gets the the Name of the Base class (if derived)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	HasRpcMethods [▶ 1852]	Gets a value indicating whether this instance has RPC Methods.
	Id [▶ 1729]	Gets the ID of the DataType (Inherited from DataType [▶ 1721].)
	InterfaceImplementationNames [▶ 1853]	Gets the names of the interfaces, this IDataType [▶ 2475] implements.
	InterfaceImplementations [▶ 1854]	Gets the resolved interface types, this IDataType [▶ 2475] implments.
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DataType [▶ 1721].)
	IsDerived [▶ 1852]	Gets a value indicating whether this instance is derived.
	ManagedType [▶ 1733]	Gets the corresponding .NET Type if attached. (Inherited from DataType [▶ 1721].)

	Name	Description
	Members [▶ 1853]	Gets a read only collection of the Members [▶ 2561] of the IStructType [▶ 2671].
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	RpcMethods [▶ 1854]	Gets the Method descriptions for the IRpcCallableType [▶ 2624]
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721].)

Extension Methods





	Name	Description
	AddAligned [▶ 1956]	Adds a member to the StructType (Defined by FluentStructTypeExtension [▶ 1955].)
	AddMethod [▶ 1956]	Adds a RpcMethod. (Defined by FluentStructTypeExtension [▶ 1955].)
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive [▶ 3033]	Overloaded.

	Name	Description
		Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]






Also see about this

-  [DataType.IsContainer Property](#) [[▶ 1731](#)]
-  [DataType.IsPointer Property](#) [[▶ 1732](#)]
-  [DataType.IsPrimitive Property](#) [[▶ 1732](#)]
-  [DataType.IsReference Property](#) [[▶ 1733](#)]

6.7.21.1 StructType Properties

The [StructType](#) [[▶ 1846](#)] type exposes the following members.

Properties

	Name	Description
	AllMembers [▶ 1851]	Gets all members (down the derivation hierarchy)
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BaseType [▶ 1851]	Gets the structs Base Type (Null if not derived).
	BaseTypeName [▶ 1851]	Gets the the Name of the Base class (if derived)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)





	Name	Description
	ByteSize [▸ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▸ 1721].)
	Category [▸ 1728]	Gets the Data Type category (Inherited from DataType [▸ 1721].)
	Comment [▸ 1729]	Gets the comment. (Inherited from DataType [▸ 1721].)
	FullName [▸ 1729]	Gets the full name of the IDataType [▸ 2475] (Namespace + Name) (Inherited from DataType [▸ 1721].)
	HasRpcMethods [▸ 1852]	Gets a value indicating whether this instance has RPC Methods.
	Id [▸ 1729]	Gets the ID of the DataType (Inherited from DataType [▸ 1721].)
	InterfaceImplementationNames [▸ 1853]	Gets the names of the interfaces, this IDataType [▸ 2475] implements.
	InterfaceImplementations [▸ 1854]	Gets the resolved interface types, this IDataType [▸ 2475] implments.
	IsBitType [▸ 1730]	Gets a value indicating whether this IDataType [▸ 2475] is a bit mapping Type (Inherited from DataType [▸ 1721].)
	IsByteAligned [▸ 1730]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DataType [▸ 1721].)
	IsDerived [▸ 1852]	Gets a value indicating whether this instance is derived.
	ManagedType [▸ 1733]	Gets the corresponding .NET Type if attached. (Inherited from DataType [▸ 1721].)
	Members [▸ 1853]	Gets a read only collection of the Members [▸ 2561] of the IStructType [▸ 2671].
	Name [▸ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▸ 1721].)
	Namespace [▸ 1734]	Gets the namespace string within the IDataType [▸ 2475] exists. (Inherited from DataType [▸ 1721].)
	RpcMethods [▸ 1854]	Gets the Method descriptions for the IRpcCallableType [▸ 2624]
	Size [▸ 1735]	Gets the Size of the DataType [▸ 1721] in Bytes or bits. (Inherited from DataType [▸ 1721].)
	TypeGuid [▸ 1735]	Gets the Guid of the DataType [▸ 1721] (optional) (Inherited from DataType [▸ 1721].)

Reference

[StructType Class \[▸ 1846\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

Also see about this

-  [DataType.IsContainer Property \[▸ 1731\]](#)
-  [DataType.IsPointer Property \[▸ 1732\]](#)
-  [DataType.IsPrimitive Property \[▸ 1732\]](#)
-  [DataType.IsReference Property \[▸ 1733\]](#)

6.7.21.1.1 StructType.AllMembers Property

Gets all members (down the derivation hierarchy)

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ICollection AllMembers { get; }
```

Property Value

Type: [ICollection](#) [[▶ 2565](#)]

All members.

Implements

[IInterfaceType.AllMembers](#) [[▶ 2987](#)]

Reference

[StructType Class](#) [[▶ 1846](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.21.1.2 StructType.BaseType Property

Gets the structs Base Type (Null if not derived).

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataType? BaseType { get; }
```

Property Value

Type: [IDataType](#) [[▶ 2475](#)]

Implements

[IInterfaceType.BaseType](#) [[▶ 2988](#)]

Reference

[StructType Class](#) [[▶ 1846](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.21.1.3 StructType.BaseTypeName Property

Gets the the Name of the Base class (if derived)

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string BaseTypeName { get; }
```

Property Value

Type: String

Empty if not derived.

Implements

[IInterfaceType.BaseTypeName](#) [[▶ 2988](#)]

Reference

[StructType Class](#) [[▶ 1846](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.21.1.4 StructType.HasRpcMethods Property

Gets a value indicating whether this instance has RPC Methods.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool HasRpcMethods { get; }
```

Property Value

Type: Boolean

true if this instance has methods; otherwise, false.

Implements

[IInterfaceType.HasRpcMethods](#) [[▶ 2986](#)]

Remarks

The DataType (Structure) must be marked with the PlcAttribute 'TcRpcEnable' to enable RpcMethods, otherwise RpcMethods are not passed through to the ADS symbolic information.

Reference

[StructType Class](#) [[▶ 1846](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.21.1.5 StructType.IsDerived Property

Gets a value indicating whether this instance is derived.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsDerived { get; }
```

Property Value

Type: Boolean

true if this instance is derived; otherwise, false.

Reference

[StructType Class](#) [[▶ 1846](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.21.1.6 StructType.Members Property

Gets a read only collection of the [Members](#) [[▶ 2561](#)] of the [IStructType](#) [[▶ 2671](#)].

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IMemberCollection Members { get; }
```

Property Value

Type: [IMemberCollection](#) [[▶ 2565](#)]

The members as read only collection.

Implements

[IInterfaceType.Members](#) [[▶ 2988](#)]

Remarks

If the [IStructType](#) [[▶ 2671](#)] is derived, only the extended members are returned. To get all supported members down the inheritance chain, use the [AllMembers](#) [[▶ 2987](#)] property.

Reference

[StructType Class](#) [[▶ 1846](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.21.1.7 StructType.InterfaceImplementationNames Property

Gets the names of the interfaces, this [IDataType](#) [[▶ 2475](#)] implements.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string[] InterfaceImplementationNames { get; }
```

Property Value

Type: `.String`.
The interface implementations.

Implements

[IInterfaceType.InterfaceImplementationNames](#) [[▶ 2987](#)]

Reference

[StructType Class](#) [[▶ 1846](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.21.1.8 StructType.InterfaceImplementations Property

Gets the resolved interface types, this [IDataType](#) [[▶ 2475](#)] implements.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IInterfaceType?[]? InterfaceImplementations { get; }
```

Property Value

Type: `.IInterfaceType` [[▶ 2983](#)].
The implements.

Implements

[IInterfaceType.InterfaceImplementations](#) [[▶ 2987](#)]

Reference

[StructType Class](#) [[▶ 1846](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.21.1.9 StructType.RpcMethods Property

Gets the Method descriptions for the [IRpcCallableType](#) [[▶ 2624](#)]

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IRpcMethodCollection RpcMethods { get; }
```

Property Value

Type: [IRpcMethodCollection](#) [[▶ 2629](#)]
 The methods.

Implements

[IRpcCallableType.RpcMethods](#) [[▶ 2624](#)]

Remarks

The `DataType` (Structure) must be marked with the `PlcAttribute` 'TcRpcEnable' to enable `RpcMethods`, otherwise `RpcMethods` are not passed through to the ADS symbolic information.

Reference







[StructType Class](#) [[▶ 1846](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.21.2 StructType Methods

The [StructType](#) [[▶ 1846](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721].)

Extension Methods

	Name	Description
	AddAligned [▶ 1956]	Adds a member to the StructType [▶ 1846] (Defined by FluentStructTypeExtension [▶ 1955].)
	AddMethod [▶ 1956]	Adds a RpcMethod . (Defined by FluentStructTypeExtension [▶ 1955].)
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the <code>dataType</code> is an array of Primitives . (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the <code>dataType</code> is an array of Primitives . (Defined by DataTypeExtension [▶ 3021].)
	IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)

Name	Description
	Overloaded.
IsContainer(Boolean) [▶ 3028]	Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021] .)
IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021] .)
	Overloaded.
IsPrimitive. [▶ 3033]	Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021] .)
	Overloaded.
IsPrimitive(Boolean) [▶ 3034]	Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021] .)
IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021] .)
ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021] .)
	Overloaded.
ResolvableAsPrimitive. [▶ 3039]	Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021] .)
	Overloaded.
ResolvableAsPrimitive(Boolean) [▶ 3039]	Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021] .)
Resolve. [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021] .)
	Overloaded.
Resolve(DataTypeResolveStrategy) [▶ 3041]	Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021] .)

Reference

[StructType Class \[▶ 1846\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.22 SubRangeType.T. Class

Represents a SubRangType

Inheritance Hierarchy

System.Object

[TwinCAT.Ads.TypeSystem.DataType \[▶ 1721\]](#)

[TwinCAT.Ads.TypeSystem.SubRangeType.T.](#)

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public sealed class SubRangeType<T> : DataType,
    ISubRangeType<T>, ISubRangeType, IDataType, IBitSize
where T : struct, new()
```

Type Parameters





T

The SubRangeType.T. type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BaseType [▶ 1860]	Gets the the base type.
	BaseTypeName [▶ 1860]	Gets the name of the base type.
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the Data Type (Inherited from DataType [▶ 1721].)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DataType [▶ 1721].)
	LowerBound [▶ 1861]	Gets the lower bound.
	ManagedType [▶ 1861]	Gets the corresponding .NET Type if attached. (Overrides DataType.ManagedType [▶ 1733].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)
	UpperBound [▶ 1862]	Gets the upper bound.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721].)

Extension Methods





	Name	Description
	IsArrayOfPrimitives . [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer . [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive . [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive . [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve . [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)

	Name	Description
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]
















Also see about this




-  [DataType.IsContainer Property](#) [[▶ 1731](#)]
-  [DataType.IsPointer Property](#) [[▶ 1732](#)]
-  [DataType.IsPrimitive Property](#) [[▶ 1732](#)]
-  [DataType.IsReference Property](#) [[▶ 1733](#)]

6.7.22.1 SubRangeType.T. Properties

The [SubRangeType.T](#) [[▶ 1856](#)] generic type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BaseType [▶ 1860]	Gets the the base type.
	BaseTypeName [▶ 1860]	Gets the name of the base type.
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the DataType (Inherited from DataType [▶ 1721].)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DataType [▶ 1721].)
	LowerBound [▶ 1861]	Gets the lower bound.
	ManagedType [▶ 1861]	Gets the corresponding .NET Type if attached. (Overrides DataType.ManagedType [▶ 1733].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)





	Name	Description
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)
	UpperBound [▶ 1862]	Gets the upper bound.

Reference

[SubRangeType.T. Class](#) [[▶ 1856](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

Also see about this

-  [DataType.IsContainer Property](#) [[▶ 1731](#)]
-  [DataType.IsPointer Property](#) [[▶ 1732](#)]
-  [DataType.IsPrimitive Property](#) [[▶ 1732](#)]
-  [DataType.IsReference Property](#) [[▶ 1733](#)]

6.7.22.1.1 SubRangeType.T..BaseType Property

Gets the the base type.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDatatype? BaseType { get; }
```

Property Value

Type: [IDatatype](#) [[▶ 2475](#)]

The type of the referenced.

Implements

[ISubRangeType.BaseType](#) [[▶ 2684](#)]

Reference

[SubRangeType.T. Class](#) [[▶ 1856](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.22.1.2 SubRangeType.T..BaseTypeName Property

Gets the name of the base type.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string BaseTypeName { get; }
```

Property Value

Type: String

The name of the base type.

Implements

[ISubRangeType.BaseTypeName](#) [[▶ 2685](#)]

Reference

[SubRangeType.T. Class](#) [[▶ 1856](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.22.1.3 SubRangeType.T..LowerBound Property

Gets the lower bound.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T LowerBound { get; }
```

Property Value

Type: [T](#) [[▶ 1856](#)]

The lower bound.

Implements

[ISubRangeType.T..LowerBound](#) [[▶ 2689](#)]

Reference

[SubRangeType.T. Class](#) [[▶ 1856](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.22.1.4 SubRangeType.T..ManagedType Property

Gets the corresponding .NET Type if attached.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override Type ManagedType { get; }
```

Property Value

Type: Type
Dot net type.

Reference

[SubRangeType.T. Class \[▶ 1856\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.22.1.5 SubRangeType.T.UpperBound Property

Gets the upper bound.

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public T UpperBound { get; }
```

Property Value

Type: [T \[▶ 1856\]](#)
The upper bound.

Implements

[ISubRangeType.T.UpperBound \[▶ 2690\]](#)

Reference





[SubRangeType.T. Class \[▶ 1856\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.22.2 SubRangeType.T. Methods

The [SubRangeType.T. \[▶ 1856\]](#) generic type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721] .)

Extension Methods

	Name	Description
	IsArrayOfPrimitives [▶ 3025]	Overloaded.

	Name	Description
		Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer. [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive. [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve. [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[SubRangeType.T. Class](#) [▶ 1856]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.23 Symbol Class

Symbol class

Inheritance Hierarchy

System.Object

[TwinCAT.Ads.TypeSystem.Instance](#) [▶ 1765]

[TwinCAT.Ads.TypeSystem.Symbol](#)

[TwinCAT.Ads.TypeSystem.UnionInstance](#) [▶ 1992]

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

















Syntax

C#

```
public class Symbol : Instance, IEquatable<Symbol>,
    IValueSymbol, IValueRawSymbol, IHierarchicalSymbol, ISymbol, IAttributedInstance,
    IInstance, IBitSize, IValueAnySymbol, IValueAccessorProvider, ISymbolFactoryServicesProvider,
    IAdsSymbol, IProcessImageAddress, IContextMaskProvider
```

The Symbol type exposes the following members.




















Properties






	Name	Description
	AccessRights [▶ 1872]	Gets the access rights.
	Attributes [▶ 1768]	Gets the Type Attributes. (Inherited from Instance [▶ 1765].)
	BitSize [▶ 1768]	Gets the size of this Instance [▶ 1765] in bits. (Inherited from Instance [▶ 1765].)
	ByteSize [▶ 1769]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from Instance [▶ 1765].)
	Category [▶ 1769]	Gets the the DataTypeCategory [▶ 2111] of the Instance. (Inherited from Instance [▶ 1765].)
	Comment [▶ 1770]	Gets the comment. (Inherited from Instance [▶ 1765].)
	Connection [▶ 1873]	Gets the connection that produces values for this IValueSymbol [▶ 2775]
	ContextMask [▶ 1770]	Gets the context mask of this instance. (Inherited from Instance [▶ 1765].)
	DataType [▶ 1771]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549]. (Inherited from Instance [▶ 1765].)
	HasValue [▶ 1771]	Gets a value indicating whether this instance has a value. (Inherited from Instance [▶ 1765].)
	ImageBaseAddress [▶ 1873]	Gets the AmsAddress [▶ 752] of the Process Image
	IndexGroup [▶ 1874]	Gets the index group of the Symbol
	IndexOffset [▶ 1874]	Gets the index offset of the Symbol
	InstanceName [▶ 1771]	Gets the name of the instance (without periods (.)) (Inherited from Instance [▶ 1765].)
	InstancePath [▶ 1874]	Gets the relative / absolute access path to the instance (with periods (.)) (Overrides Instance.InstancePath [▶ 1772].)
	IsBitType [▶ 1773]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from Instance [▶ 1765].)

	Name	Description
	IsByteAligned [▶ 1773]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from Instance [▶ 1765].)
	IsContainerType [▶ 1875]	Gets a value indicating whether the Symbols datatype is a Container type.
	IsDereferencedPointer [▶ 1876]	Gets or sets a value indicating whether an ancestor is a dereferenced Pointer
	IsDereferencedReference [▶ 1876]	Gets or sets a value indicating whether an ancestor is a dereferenced Reference
	IsPersistent [▶ 1773]	Indicates that this instance is persistent. (Inherited from Instance [▶ 1765].)
	IsPointer [▶ 1774]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
	IsPrimitiveType [▶ 1877]	Gets a value indicating whether this instance is primitive.
	IsReadOnly [▶ 1774]	Indicates that this instance is read only. (Inherited from Instance [▶ 1765].)
	IsRecursive [▶ 1877]	Gets a value indicating whether this instance is recursive.
	IsReference [▶ 1775]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
	IsStatic [▶ 1775]	Gets a value indicating whether this Instance [▶ 2549] is static. (Inherited from Instance [▶ 1765].)
	IsTcComInterfacePointer [▶ 1776]	Indicates that this instance is a TcComInterfacePointer. (Inherited from Instance [▶ 1765].)
	IsTypeGuid [▶ 1776]	Indicates that this instance has set TypeGuid flag. (Inherited from Instance [▶ 1765].)
	IsVirtual [▶ 1878]	Gets a value indicating whether this instance is virtual.
	Namespace [▶ 1776]	Gets the namespace name. (Inherited from Instance [▶ 1765].)
	NotificationSettings [▶ 1878]	Gets or sets the notification settings.
	Parent [▶ 1879]	Gets the parent Symbol
	Size [▶ 1777]	Gets the size of the IDataType [▶ 2475] in bytes or Bits dependant on IsBitType [▶ 1773] (Inherited from Instance [▶ 1765].)
	SubSymbolCount [▶ 1879]	Gets the number of SubSymbols
	SubSymbols [▶ 1880]	Gets the SubSymbols of the ISymbol [▶ 2691]
	TypeName [▶ 1777]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from Instance [▶ 1765].)
	ValueEncoding [▶ 1880]	Gets the value encoding.



Methods

	Name	Description
	EnsureRights [▶ 1885]	Ensures that the AccessRights are matched.
	Equals(ISymbol) [▶ 1885]	Equals the specified inst.
	Equals(Object) [▶ 1886]	Equals (Overrides Object.Equals(Object).)
	Equals(Symbol) [▶ 1887]	Equals the specified inst.
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode [▶ 1887]	Gets the HashCode of the Address (Overrides Object.GetHashCode..)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	OnGetSize [▶ 1779]	Handler function getting the size of the Instance [▶ 1765] (Inherited from Instance [▶ 1765].)
	OnReadRawValue [▶ 1888]	Handler function for reading the raw value
	OnReadRawValueAsync [▶ 1888]	Handler function for reading the raw value
	OnReadValue [▶ 1889]	Handler function for reading the dynamic value.
	OnReadValueAsync [▶ 1889]	Handler function for reading the dynamic value.
	OnSetInstanceName [▶ 1890]	Sets a new InstanceName InstancePath (Overrides Instance.OnSetInstanceName(String) [▶ 1780].)
	OnTryReadValue [▶ 1890]	Handler function for reading the dynamic value.
	OnTryWriteValue [▶ 1891]	Handler function for writing the dynamic value
	OnWriteRawValue [▶ 1891]	Handler function for writing the RawValue
	OnWriteRawValueAsync [▶ 1892]	Handler function Writing the raw value asynchronously.
	OnWriteValue [▶ 1893]	Handler function for writing the dynamic value
	OnWriteValueAsync [▶ 1893]	Handler function for writing the dynamic value
	ReadAnyValue(Type) [▶ 1894]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type
	ReadAnyValue(Type, Int32) [▶ 1895]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type





	Name	Description
	ReadAnyValue.T. [▶ 1896]	Reads the value of this Value [▶ 2753] into a new created instance of the managed type
	ReadAnyValue.T. (Int32) [▶ 1896]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type
	ReadAnyValueAsync (Type, CancellationTok en) [▶ 1898]	Reads the (AnyType) value asynchronously.
	ReadAnyValueAsync .T. (CancellationTok en) [▶ 1898]	Read any value as an asynchronous operation.
	ReadRawValue. [▶ 1899]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	ReadRawValue(Int32) [▶ 1900]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	ReadRawValueAsyn c [▶ 1901]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) asynchronously.
	ReadValue. [▶ 1902]	Reads the Value of the IValueSymbol [▶ 2775]
	ReadValue(Int32) [▶ 1902]	Reads the Value of the IValueSymbol [▶ 2775]
	ReadValueAsync [▶ 1903]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously.
	SetAttributes [▶ 1780]	Sets the type attributes (Inherited from Instance [▶ 1765].)
	SetContextMask [▶ 1780]	Sets the context mask. (Inherited from Instance [▶ 1765].)
	SetParent [▶ 1904]	Sets the parent symbol.
	ToString [▶ 1904]	Returns a String that represents this instance. (Overrides Instance.ToString. [▶ 1781].)
	TryReadValue [▶ 1905]	Reads the Value of the IValueSymbol [▶ 2775]
	TryWriteValue [▶ 1905]	Writes the Value of the IValueSymbol [▶ 2775]
	UpdateAnyValue(O bject.) [▶ 1907]	Reads the value of this Value [▶ 2775] into the specified managed value.
	UpdateAnyValue(O bject., Int32) [▶ 1907]	Reads the value of this Value [▶ 2775] into the specified managed value.
	WriteRawValue.(Byt e.) [▶ 1908]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)

	Name	Description
	WriteRawValue(Byte, Int32) [▶ 1909]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	WriteRawValueAsync [▶ 1910]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	WriteValue(Object) [▶ 1910]	Writes the Value of the IValueSymbol [▶ 2775]
	WriteValue(Object, Int32) [▶ 1911]	Writes the Value of the IValueSymbol [▶ 2775]
	WriteValueAsync [▶ 1912]	Writes the Value of the IValueSymbol [▶ 2775]


Events

	Name	Description
	RawValueChanged [▶ 1913]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed.
	ValueChanged [▶ 1913]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed.





Operators














	Name	Description
 	Equality [▶ 1914]	Operator==
 	Inequality [▶ 1915]	Implements the != operator.


Fields

	Name	Description
	syncObject [▶ 1916]	Synchronization object

Extension Methods

	Name	Description
	PollValues(IObservableUnit) [▶ 1355]	Overloaded. Poll symbol values on trigger signals. (Defined by ValueSymbolExtensions [▶ 1333].)
 	PollValues(TimeSpan) [▶ 1356]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues(IObservableUnit, Boolean) [▶ 1357]	Overloaded. Polls symbol values on trigger signals. (Defined by ValueSymbolExtensions [▶ 1333].)

	Name	Description
	PollValues(TimeSpan, Boolean) [▶ 1358]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues.T.(IObservable.Unit.) [▶ 1362]	Overloaded. Poll symbol values as a value sequence on trigger signals (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues.T.(TimeSpan) [▶ 1363]	Overloaded. Polls the symbol as value sequence of object values with a specified period time (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues.T.(IObservable.Unit., Func.ResultReadValueAccess2.IValueSymbol, Object., T.) [▶ 1365]	Overloaded. Poll symbol values on trigger signals (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues.T.(IObservable.Unit., Boolean) [▶ 1364]	Overloaded. Poll symbol values on trigger signals (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues.T.(TimeSpan, Boolean) [▶ 1367]	Overloaded. Polls the symbol as value sequence of object values with a specified period time (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues2(IObservable.Unit.) [▶ 1368]	Overloaded. Poll symbol values as a sequence of annotated results (Value + ErrorCode) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues2(TimeSpan) [▶ 1369]	Overloaded. Poll symbol values with communication return codes. (Defined by ValueSymbolExtensions [▶ 1333].)
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	WhenValueChanged [▶ 1344]	Gets an observable sequence when the value of the IValueSymbol [▶ 2775] has changed. (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues(IObservable.Object.) [▶ 1347]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775]. (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues(IObservable.Object., Action.Exception.) [▶ 1348]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775]. (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues(IObservable.Object., Cancellation.Token) [▶ 1349]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775]. (Defined by ValueSymbolExtensions [▶ 1333].)

	Name	Description
	WriteValues(IObservable.Object., Action.Exception., CancellationToken) [▶ 1350]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775] . (Defined by ValueSymbolExtensions [▶ 1333] .)

Remarks

A Symbol is a (named) memory object within the Process Image with a fixed address indicated by Index Group and Index Offset. Symbols can optionally be addressed by instance path and are bound to a specific [DataType \[▶ 1721\]](#).

Reference

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

[TwinCAT.Ads.TypeSystem.Instance \[▶ 1765\]](#)

[TwinCAT.TypeSystem.IValueSymbol \[▶ 2775\]](#)

[TwinCAT.TypeSystem.IValueAnySymbol \[▶ 2753\]](#)

[TwinCAT.TypeSystem.IValueAccessorProvider \[▶ 2752\]](#)

[TwinCAT.TypeSystem.ISymbolFactoryServicesProvider \[▶ 2711\]](#)

[TwinCAT.TypeSystem.IHierarchicalSymbol \[▶ 2545\]](#)

[ISymbolValueChangeNotify](#)



[TwinCAT.Ads.TypeSystem.IContextMaskProvider \[▶ 1764\]](#)

[IInstanceInternal](#)

[ISymbolInternal](#)

[TwinCAT.Ads.TypeSystem.IAdsSymbol \[▶ 1755\]](#)





Also see about this















-  [Symbol.Equals Method \[▶ 1885\]](#)
-  [Symbol.ReadAnyValueAsync Method \[▶ 1897\]](#)












6.7.23.1 Symbol Properties

The [Symbol \[▶ 1863\]](#) type exposes the following members.

Properties

	Name	Description
	AccessRights [▶ 1872]	Gets the access rights.
	Attributes [▶ 1768]	Gets the Type Attributes. (Inherited from Instance [▶ 1765] .)
	BitSize [▶ 1768]	Gets the size of this Instance [▶ 1765] in bits. (Inherited from Instance [▶ 1765] .)
	ByteSize [▶ 1769]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from Instance [▶ 1765] .)

	Name	Description
	Category [▶ 1769]	Gets the the DataTypeCategory [▶ 2111] of the Instance. (Inherited from Instance [▶ 1765].)
	Comment [▶ 1770]	Gets the comment. (Inherited from Instance [▶ 1765].)
	Connection [▶ 1873]	Gets the connection that produces values for this IValueSymbol [▶ 2775]
	ContextMask [▶ 1770]	Gets the context mask of this instance. (Inherited from Instance [▶ 1765].)
	DataType [▶ 1771]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549]. (Inherited from Instance [▶ 1765].)
	HasValue [▶ 1771]	Gets a value indicating whether this instance has a value. (Inherited from Instance [▶ 1765].)
	ImageBaseAddress [▶ 1873]	Gets the AmsAddress [▶ 752] of the Process Image
	IndexGroup [▶ 1874]	Gets the index group of the Symbol
	IndexOffset [▶ 1874]	Gets the index offset of the Symbol
	InstanceName [▶ 1771]	Gets the name of the instance (without periods (.)) (Inherited from Instance [▶ 1765].)
	InstancePath [▶ 1874]	Gets the relative / absolute access path to the instance (with periods (.)) (Overrides Instance.InstancePath [▶ 1772].)
	IsBitType [▶ 1773]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from Instance [▶ 1765].)
	IsByteAligned [▶ 1773]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from Instance [▶ 1765].)
	IsContainerType [▶ 1875]	Gets a value indicating whether the Symbols datatype is a Container type.
	IsDereferencedPointer [▶ 1876]	Gets or sets a value indicating whether an ancestor is a dereferenced Pointer
	IsDereferencedReference [▶ 1876]	Gets or sets a value indicating whether an ancestor is a dereferenced Reference
	IsPersistent [▶ 1773]	Indicates that this instance is persistent. (Inherited from Instance [▶ 1765].)
	IsPointer [▶ 1774]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
	IsPrimitiveType [▶ 1877]	Gets a value indicating whether this instance is primitive.
	IsReadOnly [▶ 1774]	Indicates that this instance is read only. (Inherited from Instance [▶ 1765].)
	IsRecursive [▶ 1877]	Gets a value indicating whether this instance is recursive.
	IsReference [▶ 1775]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
	IsStatic [▶ 1775]	Gets a value indicating whether this Instance [▶ 2549] is static. (Inherited from Instance [▶ 1765].)

	Name	Description
	IsTcComInterfacePointer [▸ 1776]	Indicates that this instance is a TcComInterfacePointer. (Inherited from Instance [▸ 1765].)
	IsTypeGuid [▸ 1776]	Indicates that this instance has set TypeGuid flag. (Inherited from Instance [▸ 1765].)
	IsVirtual [▸ 1878]	Gets a value indicating whether this instance is virtual.
	Namespace [▸ 1776]	Gets the namespace name. (Inherited from Instance [▸ 1765].)
	NotificationSettings [▸ 1878]	Gets or sets the notification settings.
	Parent [▸ 1879]	Gets the parent Symbol
	Size [▸ 1777]	Gets the size of the IDataType [▸ 2475] in bytes or Bits dependant on IsBitType [▸ 1773] (Inherited from Instance [▸ 1765].)
	SubSymbolCount [▸ 1879]	Gets the number of SubSymbols
	SubSymbols [▸ 1880]	Gets the SubSymbols of the ISymbol [▸ 2691]
	TypeName [▸ 1777]	Gets the name of the DataType [▸ 2475] that is used for this Instance [▸ 2549]. (Inherited from Instance [▸ 1765].)
	ValueEncoding [▸ 1880]	Gets the value encoding.

Reference

[Symbol Class](#) [[▸ 1863](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▸ 1698](#)]

6.7.23.1.1 Symbol.AccessRights Property

Gets the access rights.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▸ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolAccessRights AccessRights { get; }
```

Property Value

Type: [SymbolAccessRights](#) [[▸ 2928](#)]

The access rights.

Implements

[IValueSymbol.AccessRights](#) [[▸ 2780](#)]

Reference

[Symbol Class](#) [[▸ 1863](#)]

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.23.1.2 Symbol.Connection Property

Gets the connection that produces values for this [IValueSymbol \[▶ 2775\]](#)

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IConnection? Connection { get; }
```

Property Value

Type: [IConnection \[▶ 79\]](#)

The connection object.

Implements

[IValueSymbol.Connection \[▶ 2781\]](#)

Reference

[Symbol Class \[▶ 1863\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.23.1.3 Symbol.ImageBaseAddress Property

Gets the [AmsAddress \[▶ 752\]](#) of the Process Image

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsAddress? ImageBaseAddress { get; }
```

Property Value

Type: [AmsAddress \[▶ 752\]](#)

The address.

Implements

[IAdsSymbol.ImageBaseAddress \[▶ 1759\]](#)

Reference

[Symbol Class \[▶ 1863\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.23.1.4 Symbol.IndexGroup Property

Gets the index group of the Symbol

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint IndexGroup { get; }
```

Property Value

Type: UInt32

The index group.

Implements

[IProcessImageAddress.IndexGroup](#) [[▶ 2594](#)]

Reference

[Symbol Class](#) [[▶ 1863](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.1.5 Symbol.IndexOffset Property

Gets the index offset of the Symbol

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint IndexOffset { get; }
```

Property Value

Type: UInt32

The index offset.

Implements

[IProcessImageAddress.IndexOffset](#) [[▶ 2594](#)]

Reference

[Symbol Class](#) [[▶ 1863](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.1.6 Symbol.InstancePath Property

Gets the relative / absolute access path to the instance (with periods (.))

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string InstancePath { get; }
```

Property Value

Type: String

The instance path.

Implements

[IInstance.InstancePath \[▶ 2552\]](#)

[IInstance.InstancePath \[▶ 2552\]](#)

Remarks

If this path is relative or absolute depends on the context. [IMember \[▶ 2561\]](#) are using relative paths, [ISymbol \[▶ 2691\]](#)s are using absolute ones.

Reference

[Symbol Class \[▶ 1863\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.23.1.7 Symbol.IsContainerType Property

Gets a value indicating whether the Symbols datatype is a Container type.

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public virtual bool IsContainerType { get; }
```

Property Value

Type: Boolean

true if this instance is container type; otherwise, false.

Implements

[ISymbol.IsContainerType \[▶ 2694\]](#)

Remarks

Container Types are all types that contain SubElements like

- [Array \[▶ 2111\]](#)
- [Pointer \[▶ 2111\]](#)
- [Union \[▶ 2111\]](#)

- [Struct \[▸ 2111\]](#)
- [Function \[▸ 2111\]](#)
- [FunctionBlock \[▸ 2111\]](#)
- [Program \[▸ 2111\]](#)

and the [Alias \[▸ 2111\]](#) and [Reference \[▸ 2111\]](#) types, if they have a container type as base type.

Reference

[Symbol Class \[▸ 1863\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

[IDataType.Category \[▸ 2478\]](#)

6.7.23.1.8 Symbol.IsDereferencedPointer Property

Gets or sets a value indicating whether an ancestor is a dereferenced Pointer

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsDereferencedPointer { get; }
```

Property Value

Type: Boolean

true if this instance is ancestor is pointer; otherwise, false.

Reference

[Symbol Class \[▸ 1863\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.23.1.9 Symbol.IsDereferencedReference Property

Gets or sets a value indicating whether an ancestor is a dereferenced Reference

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsDereferencedReference { get; }
```

Property Value

Type: Boolean

true if this instance is ancestor is reference; otherwise, false.

Reference

[Symbol Class \[▸ 1863\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.23.1.10 Symbol.IsPrimitiveType Property

Gets a value indicating whether this instance is primitive.

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public virtual bool IsPrimitiveType { get; }
```

Property Value

Type: Boolean

true if this instance is primitive; otherwise, false.

Implements

[ISymbol.IsPrimitiveType \[► 2695\]](#)

Reference

[Symbol Class \[► 1863\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.23.1.11 Symbol.IsRecursive Property

Gets a value indicating whether this instance is recursive.

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsRecursive { get; }
```

Property Value

Type: Boolean

true if this instance is recursive; otherwise, false.

Implements

[ISymbol.IsRecursive \[► 2696\]](#)

Reference

[Symbol Class \[► 1863\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.23.1.12 Symbol.IsVirtual Property

Gets a value indicating whether this instance is virtual.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsVirtual { get; }
```

Property Value

Type: Boolean

true if this instance is virtual; otherwise, false.

Implements

[IProcessImageAddress.IsVirtual](#) [[▶ 2595](#)]

Remarks

Virtual symbols are only organizational elements within the Symbols Hierarchy and cannot be accessed separately by IndexGroup/IndexOffset, Value Read/Writes, notifications or handles.

Reference

[Symbol Class](#) [[▶ 1863](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.1.13 Symbol.NotificationSettings Property

Gets or sets the notification settings.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public INotificationSettings? NotificationSettings { get; set; }
```

Property Value

Type: [INotificationSettings](#) [[▶ 1098](#)]

The notification settings.

Implements

[IValueSymbol.NotificationSettings](#) [[▶ 2781](#)]

Remarks

The NotificationSettings will be inherited from [Parent](#) [[▶ 1879](#)] if the setting is not overwritten. If the Root Symbol also doesn't contain the settings, then the [DefaultNotificationSettings](#) [[▶ 1762](#)] will be returned.

Reference

[Symbol Class](#) [► 1863]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.23.1.14 Symbol.Parent Property

Gets the parent Symbol

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISymbol? Parent { get; }
```

Property Value

Type: [ISymbol](#) [► 2691]

The parent.

Implements

[ISymbol.Parent](#) [► 2696]

Reference

[Symbol Class](#) [► 1863]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.23.1.15 Symbol.SubSymbolCount Property

Gets the number of SubSymbols

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int SubSymbolCount { get; }
```

Property Value

Type: Int32

The Number of SubSymbols.

Remarks

If the [SubSymbols](#) [► 1880] collection is not generated yet (WeakReference), then this method is less memory and cpu consuming to use for just determining the the number of child symbols (instead of using `SubSymbols.Count`)>

Reference

[Symbol Class](#) [► 1863]

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.23.1.16 Symbol.SubSymbols Property

Gets the SubSymbols of the [ISymbol \[▶ 2691\]](#)

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISymbolCollection<ISymbol> SubSymbols { get; }
```

Property Value

Type: [ISymbolCollection \[▶ 2700\]](#).[ISymbol \[▶ 2691\]](#).

Implements

[ISymbol.SubSymbols \[▶ 2696\]](#)

Remarks

Used for Array, Struct, Pointer and Reference instances. Otherwise empty

Reference

[Symbol Class \[▶ 1863\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.23.1.17 Symbol.ValueEncoding Property

Gets the value encoding.

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public virtual Encoding ValueEncoding { get; }
```

Property Value

Type: Encoding
The value encoding.

Implements

[IAttributedInstance.ValueEncoding \[▶ 2472\]](#)

Reference

[Symbol Class \[▶ 1863\]](#)




















[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)








6.7.23.2 Symbol Methods

The [Symbol](#) [[▶ 1863](#)] type exposes the following members.












Methods

	Name	Description
	EnsureRights [▶ 1885]	Ensures that the AccessRights are matched.
	Equals(ISymbol) [▶ 1885]	Equals the specified inst.
	Equals(Object) [▶ 1886]	Equals (Overrides Object.Equals(Object).)
	Equals(Symbol) [▶ 1887]	Equals the specified inst.
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode [▶ 1887]	Gets the GetHashCode of the Address (Overrides Object.GetHashCode..)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	OnGetSize [▶ 1779]	Handler function getting the size of the Instance [▶ 1765] (Inherited from Instance [▶ 1765].)
	OnReadRawValue [▶ 1888]	Handler function for reading the raw value
	OnReadRawValueAsync [▶ 1888]	Handler function for reading the raw value
	OnReadValue [▶ 1889]	Handler function for reading the dynamic value.
	OnReadValueAsync [▶ 1889]	Handler function for reading the dynamic value.
	OnSetInstanceName [▶ 1890]	Sets a new InstanceName InstancePath (Overrides Instance.OnSetInstanceName(String) [▶ 1780].)
	OnTryReadValue [▶ 1890]	Handler function for reading the dynamic value.
	OnTryWriteValue [▶ 1891]	Handler function for writing the dynamic value
	OnWriteRawValue [▶ 1891]	Handler function for writing the RawValue
	OnWriteRawValueAsync [▶ 1892]	Handler function Writing the raw value asynchronously.
	OnWriteValue [▶ 1893]	Handler function for writing the dynamic value
	OnWriteValueAsync [▶ 1893]	Handler function for writing the dynamic value

	Name	Description
	ReadAnyValue(Type) [▸ 1894]	Reads the value of this Value [▸ 2775] into a new created instance of the managed type
	ReadAnyValue(Type, Int32) [▸ 1895]	Reads the value of this Value [▸ 2775] into a new created instance of the managed type
	ReadAnyValue.T.. [▸ 1896]	Reads the value of this Value [▸ 2753] into a new created instance of the managed type
	ReadAnyValue.T.(Int32) [▸ 1896]	Reads the value of this Value [▸ 2775] into a new created instance of the managed type
	ReadAnyValueAsync(Type, CancellationToken) [▸ 1898]	Reads the (AnyType) value asynchronously.
	ReadAnyValueAsync.T.(CancellationToken) [▸ 1898]	Read any value as an asynchronous operation.
	ReadRawValue. [▸ 1899]	Reads the raw value of the IValueSymbol [▸ 2775] (Ads Read / Write)
	ReadRawValue(Int32) [▸ 1900]	Reads the raw value of the IValueSymbol [▸ 2775] (Ads Read / Write)
	ReadRawValueAsync [▸ 1901]	Reads the raw value of the IValueSymbol [▸ 2775] (Ads Read / Write) asynchronously.
	ReadValue. [▸ 1902]	Reads the Value of the IValueSymbol [▸ 2775]
	ReadValue(Int32) [▸ 1902]	Reads the Value of the IValueSymbol [▸ 2775]
	ReadValueAsync [▸ 1903]	Reads the Value of the IValueSymbol [▸ 2775] asynchronously.
	SetAttributes [▸ 1780]	Sets the type attributes (Inherited from Instance [▸ 1765] .)
	SetContextMask [▸ 1780]	Sets the context mask. (Inherited from Instance [▸ 1765] .)
	SetParent [▸ 1904]	Sets the parent symbol.
	ToString [▸ 1904]	Returns a String that represents this instance. (Overrides Instance.ToString. [▸ 1781] .)
	TryReadValue [▸ 1905]	Reads the Value of the IValueSymbol [▸ 2775]
	TryWriteValue [▸ 1905]	Writes the Value of the IValueSymbol [▸ 2775]
	UpdateAnyValue(Object.) [▸ 1907]	Reads the value of this Value [▸ 2775] into the specified managed value.

	Name	Description
	UpdateAnyValue(Object, Int32) [▶ 1907]	Reads the value of this Value [▶ 2775] into the specified managed value.
	WriteRawValue(Byte) [▶ 1908]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	WriteRawValue(Byte, Int32) [▶ 1909]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	WriteRawValueAsync [▶ 1910]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	WriteValue(Object) [▶ 1910]	Writes the Value of the IValueSymbol [▶ 2775]
	WriteValue(Object, Int32) [▶ 1911]	Writes the Value of the IValueSymbol [▶ 2775]
	WriteValueAsync [▶ 1912]	Writes the Value of the IValueSymbol [▶ 2775]

Extension Methods

	Name	Description
	PollValues(IObservable.Unit) [▶ 1355]	Overloaded. Poll symbol values on trigger signals. (Defined by ValueSymbolExtensions [▶ 1333] .)
 	PollValues(TimeSpan) [▶ 1356]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by ValueSymbolExtensions [▶ 1333] .)
	PollValues(IObservable.Unit, Boolean) [▶ 1357]	Overloaded. Polls symbol values on trigger signals. (Defined by ValueSymbolExtensions [▶ 1333] .)
 	PollValues(TimeSpan, Boolean) [▶ 1358]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by ValueSymbolExtensions [▶ 1333] .)
	PollValues.T.(IObservable.Unit) [▶ 1362]	Overloaded. Poll symbol values as a value sequence on trigger signals (typed) (Defined by ValueSymbolExtensions [▶ 1333] .)
 	PollValues.T.(TimeSpan) [▶ 1363]	Overloaded. Polls the symbol as value sequence of object values with a specified period time (typed) (Defined by ValueSymbolExtensions [▶ 1333] .)
 	PollValues.T.(IObservable.Unit, Func.ResultReadValueAccess2.IValueSymbol, Object, T) [▶ 1365]	Overloaded. Poll symbol values on trigger signals (typed) (Defined by ValueSymbolExtensions [▶ 1333] .)



	Name	Description
	PollValues.T. (IObservable.Unit. , Boolean) [▶ 1364]	Overloaded. Poll symbol values on trigger signals (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues.T. (TimeSpan , Boolean) [▶ 1367]	Overloaded. Polls the symbol as value sequence of object values with a specified period time (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues2(IObservable.Unit.) [▶ 1368]	Overloaded. Poll symbol values as a sequence of annotated results (Value + ErrorCode) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues2(TimeSpan) [▶ 1369]	Overloaded. Poll symbol values with communication return codes. (Defined by ValueSymbolExtensions [▶ 1333].)
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
 	WhenValueChanged [▶ 1344]	Gets an observable sequence when the value of the IValueSymbol [▶ 2775] has changed. (Defined by ValueSymbolExtensions [▶ 1333].)
 	WriteValues(IObservable.Object.) [▶ 1347]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775]. (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues(IObservable.Object., Action.Exception.) [▶ 1348]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775]. (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues(IObservable.Object., CancellationToken) [▶ 1349]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775]. (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues(IObservable.Object., Action.Exception., CancellationToken) [▶ 1350]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775]. (Defined by ValueSymbolExtensions [▶ 1333].)

Reference

[Symbol Class](#) [[▶ 1863](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

Also see about this

-  [Symbol.Equals Method](#) [[▶ 1885](#)]
-  [Symbol.ReadAnyValueAsync Method](#) [[▶ 1897](#)]

6.7.23.2.1 Symbol.EnsureRights Method

Ensures that the AccessRights are matched.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected void EnsureRights(
    SymbolAccessRights requested
)
```

Parameters

requested	Type: TwinCAT.TypeSystem.SymbolAccessRights [▶ 2928] The requested rights.
-----------	---

Exceptions

Exception	Condition
InsufficientAccessRightsException [▶ 2573]	

Reference

[Symbol Class](#) [[▶ 1863](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.2.2 Symbol.Equals Method

Overload List

	Name	Description
	Equals(ISymbol) [▶ 1885]	Equalses the specified inst.
	Equals(Object) [▶ 1886]	Equals (Overrides Object.Equals(Object).)
	Equals(Symbol) [▶ 1887]	Equalses the specified inst.

Reference

[Symbol Class](#) [[▶ 1863](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

Also see about this

 [TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.2.2.1 Symbol.Equals Method (ISymbol)

Equalses the specified inst.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Equals(  
    ISymbol? inst  
)
```

Parameters

inst	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The inst.
------	--

Return Value

Type: Boolean

true if the Symbols are Equal, false otherwise.

Reference

[Symbol Class](#) [[▶ 1863](#)]

[Equals Overload](#) [[▶ 1885](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.2.2 Symbol.Equals Method (Object)

Equals

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool Equals(  
    Object? obj  
)
```

Parameters

obj	Type: System.Object The object to compare with the current object.
-----	---

Return Value

Type: Boolean

true if the specified Object is equal to this instance; otherwise, false.

Reference

[Symbol Class](#) [[▶ 1863](#)]

[Equals Overload](#) [[▶ 1885](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.2.3 Symbol.Equals Method (Symbol)

Equalses the specified inst.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Equals(  
    Symbol? inst  
)
```

Parameters

inst	Type: TwinCAT.Ads.TypeSystem.Symbol [▶ 1863] The inst.
------	---

Return Value

Type: Boolean
true if the Symbols are equal, false otherwise.

Implements

[IEquatable.T..Equals\(T\)](#)

Reference

[Symbol Class](#) [[▶ 1863](#)]

[Equals Overload](#) [[▶ 1885](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.2.3 Symbol.GetHashCode Method

Gets the HashCode of the Address

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override int GetHashCode()
```

Return Value

Type: Int32
A hash code for this instance, suitable for use in hashing algorithms and data structures like a hash table.

Reference

[Symbol Class](#) [[▶ 1863](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.2.4 Symbol.OnReadRawValue Method

Handler function for reading the raw value

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual byte[] OnReadRawValue(
    int timeout
)
```

Parameters

timeout	Type: System.Int32
---------	--------------------

Return Value

Type: .Byte.
System.Byte[].

Exceptions

Exception	Condition
AdsErrorException [▶ 682]	
AdsErrorException [▶ 682]	

Reference

[Symbol Class](#) [[▶ 1863](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.2.5 Symbol.OnReadRawValueAsync Method

Handler function for reading the raw value

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<ResultReadRawAccess> OnReadRawValueAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken
--------	--

Return Value

Type: [Task.ResultReadRawAccess](#) [[▶ 3206](#)].
System.Byte[].

Exceptions

Exception	Condition
AdsErrorException [▶ 682]	
AdsErrorException [▶ 682]	

Reference

[Symbol Class \[▶ 1863\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.23.2.6 Symbol.OnReadValue Method

Handler function for reading the dynamic value.

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Object OnReadValue (
    int timeout
)
```

Parameters

timeout	Type: System.Int32
---------	--------------------

Return Value

Type: Object
The Value

Reference

[Symbol Class \[▶ 1863\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.23.2.7 Symbol.OnReadValueAsync Method

Handler function for reading the dynamic value.

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<ResultReadValueAccess> OnReadValueAsync (
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken
--------	--

Return Value

Type: [Task.ResultReadValueAccess](#) [[▶ 3209](#)].
The Value

Reference

[Symbol Class](#) [[▶ 1863](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.2.8 Symbol.OnSetInstanceName Method

Sets a new InstanceName InstancePath

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
protected override void OnSetInstanceName(
    string instanceName
)
```

Parameters

instanceName	Type: System.String Instance name.
--------------	---------------------------------------

Reference

[Symbol Class](#) [[▶ 1863](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.2.9 Symbol.OnTryReadValue Method

Handler function for reading the dynamic value.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
protected virtual int OnTryReadValue(
    int timeout,
    out Object?? value
)
```

Parameters

timeout	Type: System.Int32
value	Type: System.Object.

Return Value

Type: Int32
The Value

Reference

[Symbol Class](#) [▶ 1863]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.23.2.10 Symbol.OnTryWriteValue Method

Handler function for writing the dynamic value

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual int OnTryWriteValue(
    Object value,
    int timeout
)
```

Parameters

value	Type: System.Object The value.
timeout	Type: System.Int32 The timeout.

Return Value

Type: Int32
System.Int32.

Exceptions

Exception	Condition
CannotAccessVirtualSymbolException [▶ 2103]	
CannotAccessVirtualSymbolException [▶ 2103]	

Reference

[Symbol Class](#) [▶ 1863]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.23.2.11 Symbol.OnWriteRawValue Method

Handler function for writing the RawValue

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual void OnWriteRawValue(
    ReadOnlyMemory<byte> value,
    int timeout
)
```

Parameters

value	Type: System.ReadOnlyMemory.Byte. The value.
timeout	Type: System.Int32 The timeout.

Exceptions

Exception	Condition
CannotAccessVirtualSymbolException [▶ 2103]	
SymbolException [▶ 2933]	

Reference

[Symbol Class](#) [▶ 1863]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.23.2.12 Symbol.OnWriteRawValueAsync Method

Handler function Writing the raw value asynchronously.

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<ResultWriteAccess> OnWriteRawValueAsync(
    ReadOnlyMemory<byte> value,
    CancellationToken cancel
)
```

Parameters

value	Type: System.ReadOnlyMemory.Byte. The value.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWriteAccess](#) [▶ 3217].

Task<ResultWriteAccess>.

Exceptions

Exception	Condition
CannotAccessVirtualSymbolException > 2103	

Reference

[Symbol Class |> 1863](#)

[TwinCAT.Ads.TypeSystem Namespace |> 1698](#)

6.7.23.2.13 Symbol.OnWriteValue Method

Handler function for writing the dynamic value

Namespace: [TwinCAT.Ads.TypeSystem |> 1698](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual void OnWriteValue(
    Object value,
    int timeout
)
```

Parameters

value	Type: System.Object The value.
timeout	Type: System.Int32 The timeout.

Exceptions

Exception	Condition
CannotAccessVirtualSymbolException > 2103	
CannotAccessVirtualSymbolException > 2103	

Reference

[Symbol Class |> 1863](#)

[TwinCAT.Ads.TypeSystem Namespace |> 1698](#)

6.7.23.2.14 Symbol.OnWriteValueAsync Method

Handler function for writing the dynamic value

Namespace: [TwinCAT.Ads.TypeSystem |> 1698](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<ResultWriteAccess> OnWriteValueAsync(
    Object value,
    CancellationToken cancel
)
```

Parameters

value	Type: System.Object The value.
cancel	Type: System.Threading.CancellationToken The cancellation token..

Return Value

Type: [Task.ResultWriteAccess](#) [▶ 3217].
Task<ResultWriteAccess>.

Exceptions

Exception	Condition
CannotAccessVirtualSymbolException [▶ 2103]	





Reference

[Symbol Class](#) [▶ 1863]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.23.2.15 Symbol.ReadAnyValue Method

Overload List

	Name	Description
	ReadAnyValue.T. [▶ 1896]	Reads the value of this Value [▶ 2753] into a new created instance of the managed type
	ReadAnyValue.T.(Int32) [▶ 1896]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type
	ReadAnyValue(Type) [▶ 1894]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type
	ReadAnyValue(Type, Int32) [▶ 1895]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type

Reference

[Symbol Class](#) [▶ 1863]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.23.2.15.1 Symbol.ReadAnyValue Method (Type)

Reads the value of this [Value](#) [▶ 2775] into a new created instance of the managed type

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object ReadAnyValue(
    Type managedType
)
```

Parameters

managedType	Type: System.Type The tp.
-------------	------------------------------

Return Value

Type: Object

Read value (System.Object).

Implements

[IValueAnySymbol.ReadAnyValue\(Type\)](#) [[▶ 2758](#)]

Reference

[Symbol Class](#) [[▶ 1863](#)]

[ReadAnyValue Overload](#) [[▶ 1894](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

[IValueAnySymbol.WriteAnyValue\(Object\)](#) [[▶ 2764](#)]

[IValueAnySymbol.UpdateAnyValue\(Object.\)](#) [[▶ 2762](#)]

6.7.23.2.15.2 Symbol.ReadAnyValue Method (Type, Int32)

Reads the value of this [Value](#) [[▶ 2775](#)] into a new created instance of the managed type

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object ReadAnyValue(
    Type managedType,
    int timeout
)
```

Parameters

managedType	Type: System.Type The tp.
timeout	Type: System.Int32 The timeout in ms.

Return Value

Type: Object
Read value (System.Object).

Implements

[IValueAnySymbol.ReadAnyValue\(Type, Int32\) \[► 2759\]](#)

Reference

[Symbol Class \[► 1863\]](#)

[ReadAnyValue Overload \[► 1894\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

[IValueAnySymbol.WriteAnyValue\(Object\) \[► 2764\]](#)

[IValueAnySymbol.UpdateAnyValue\(Object.\) \[► 2762\]](#)

6.7.23.2.15.3 Symbol.ReadAnyValue.T. Method

Reads the value of this [Value \[► 2753\]](#) into a new created instance of the managed type

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public T ReadAnyValue<T>()
```

Type Parameters

T	Type of the Value to be read.
---	-------------------------------

Return Value

Type: T
T.

Implements

[IValueAnySymbol.ReadAnyValue.T.. \[► 2759\]](#)

Reference

[Symbol Class \[► 1863\]](#)

[ReadAnyValue Overload \[► 1894\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.23.2.15.4 Symbol.ReadAnyValue.T. Method (Int32)

Reads the value of this [Value \[► 2775\]](#) into a new created instance of the managed type

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T ReadAnyValue<T>(
    int timeout
)
```

Parameters

timeout	Type: System.Int32 The timeout in ms.
---------	--

Type Parameters

T

Return Value

Type: T
 Read value (System.Object).

Implements

[IValueAnySymbol.ReadAnyValue.T.\(Int32\)](#) [[▶ 2760](#)]

Reference

- [Symbol Class](#) [[▶ 1863](#)]
- [ReadAnyValue Overload](#) [[▶ 1894](#)]
- [TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]
- [IValueAnySymbol.WriteAnyValue\(Object\)](#) [[▶ 2764](#)]
- [IValueAnySymbol.UpdateAnyValue\(Object.\)](#) [[▶ 2762](#)]

6.7.23.2.16 Symbol.ReadAnyValueAsync Method

Overload List

	Name	Description
	ReadAnyValueAsync .T. (CancellationTok en) [▶ 1898]	Read any value as an asynchronous operation.
	ReadAnyValueAsync (Type. CancellationTok en) [▶ 1898]	Reads the (AnyType) value asynchronously.

Reference[Symbol Class \[► 1863\]](#)[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)**Also see about this**[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)**6.7.23.2.16.1 Symbol.ReadAnyValueAsync.T. Method (CancellationToken)**

Read any value as an asynchronous operation.

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public Task<ResultReadValueAccess<T>> ReadAnyValueAsync<T>(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Type Parameters

T

Return Value

Type: Task.ResultReadValueAccess [► 3211].T..

A Task<ResultReadValueAccess`1> representing the asynchronous operation.

Implements[IValueAnySymbol.ReadAnyValueAsync.T.\(CancellationToken\) \[► 2761\]](#)**Reference**[Symbol Class \[► 1863\]](#)[ReadAnyValueAsync Overload \[► 1897\]](#)[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)**6.7.23.2.16.2 Symbol.ReadAnyValueAsync Method (Type, CancellationToken)**

Reads the (AnyType) value asynchronously.

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultReadValueAccess> ReadAnyValueAsync (
    Type managedType,
    CancellationToken cancel
)
```

Parameters

managedType	Type: System.Type Managed type of the value to read.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultReadValueAccess](#) [[▶ 3209](#)].
A task object that is representing the asynchronous 'ReadAnyValue' operation. The result will be returned in a [ResultReadValueAccess](#) [[▶ 3209](#)], which contains the [Value](#) [[▶ 3213](#)] and the [ErrorCode](#) [[▶ 3202](#)].

Implements



[IValueAnySymbol.ReadAnyValueAsync\(Type, CancellationToken\)](#) [[▶ 2761](#)]

Reference

- [Symbol Class](#) [[▶ 1863](#)]
- [ReadAnyValueAsync Overload](#) [[▶ 1897](#)]
- [TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.2.17 Symbol.ReadRawValue Method

Overload List

	Name	Description
	ReadRawValue. [▶ 1899]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	ReadRawValue(Int32) [▶ 1900]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)

Reference

- [Symbol Class](#) [[▶ 1863](#)]
- [TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.2.17.1 Symbol.ReadRawValue Method

Reads the raw value of the [IValueSymbol](#) [[▶ 2775](#)] (Ads Read / Write)

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public byte[] ReadRawValue()
```

Field Value

Type: .Byte.
The raw value.

Return Value

Type: .Byte.
System.Byte[].

Implements

[IValueRawSymbol.ReadRawValue](#). [[▶ 2770](#)]

Reference

[Symbol Class](#) [[▶ 1863](#)]

[ReadRawValue Overload](#) [[▶ 1899](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.2.17.2 Symbol.ReadRawValue Method (Int32)

Reads the raw value of the [IValueSymbol](#) [[▶ 2775](#)] (Ads Read / Write)

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public byte[] ReadRawValue(  
    int timeout  
)
```

Parameters

timeout	Type: System.Int32
---------	--------------------

Field Value

Type: .Byte.
The raw value.

Return Value

Type: .Byte.
System.Byte[].

Implements

[IValueRawSymbol.ReadRawValue\(Int32\)](#) [[▶ 2771](#)]

Reference

[Symbol Class \[▶ 1863\]](#)

[ReadRawValue Overload \[▶ 1899\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.23.2.18 Symbol.ReadRawValueAsync Method

Reads the raw value of the [IValueSymbol \[▶ 2775\]](#) (Ads Read / Write) asynchronously.

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultReadRawAccess> ReadRawValueAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Field Value

Type: [Task.ResultReadRawAccess \[▶ 3206\]](#).
The raw value.

Return Value

Type: [Task.ResultReadRawAccess \[▶ 3206\]](#).
System.Byte[].

Implements

[IValueRawSymbol.ReadRawValueAsync\(CancellationToken\) \[▶ 2772\]](#)



Reference

[Symbol Class \[▶ 1863\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.23.2.19 Symbol.ReadValue Method

Overload List

	Name	Description
	ReadValue. [▶ 1902]	Reads the Value of the IValueSymbol [▶ 2775]
	ReadValue(Int32) [▶ 1902]	Reads the Value of the IValueSymbol [▶ 2775]

Reference

[Symbol Class](#) [► 1863]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.23.2.19.1 Symbol.ReadValue Method

Reads the Value of the [IValueSymbol](#) [► 2775]

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object ReadValue()
```

Field Value

Type: Object
The value.

Return Value

Type: Object
System.Object.

Implements

[IValueSymbol.ReadValue.](#) [► 2784]

Remarks

Calling on primitive types, a call of this method will return the primitive value. On complex types (structures and arrays) it depends on the [ISymbolLoader](#) [► 2714] settings what will happen. In non dynamic modes: the raw byte Array will be returned, in dynamic mode: A Value will be created on the fly.

Reference

[Symbol Class](#) [► 1863]

[ReadValue Overload](#) [► 1901]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.23.2.19.2 Symbol.ReadValue Method (Int32)

Reads the Value of the [IValueSymbol](#) [► 2775]

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object ReadValue(  
    int timeout  
)
```

Parameters

timeout	Type: System.Int32 The timeout in ms.
---------	--

Field Value

Type: Object
The value.

Return Value

Type: Object
System.Object.

Implements

[IValueSymbol.ReadValue\(Int32\)](#) [[▶ 2785](#)]

Remarks

Calling on primitive types, a call of this method will return the primitive value. On complex types (structures and arrays) it depends on the [ISymbolLoader](#) [[▶ 2714](#)] settings what will happen. In non dynamic modes: the raw byte Array will be returned, in dynamic mode: A Value will be created on the fly. A negative timeout indicates that the Default Timeout for the communication will be used. 0 means timeout is switched off.

Reference

[Symbol Class](#) [[▶ 1863](#)]

[ReadValue Overload](#) [[▶ 1901](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.2.20 Symbol.ReadValueAsync Method

Reads the Value of the [IValueSymbol](#) [[▶ 2775](#)] asynchronously.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultReadValueAccess> ReadValueAsync (
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultReadValueAccess](#) [[▶ 3209](#)].

A tasks that represents the asynchronous 'ReadValue' operation. The read result is stored in the [ResultReadValueAccess](#) [[▶ 3209](#)] return value and contains the [Value](#) [[▶ 3213](#)] and the [ErrorCode](#) [[▶ 3202](#)].

Implements

[IValueSymbol.ReadValueAsync\(CancellationTokens\)](#) [[▶ 2785](#)]

Remarks

Calling on primitive types, a call of this method will return the primitive value. On complex types (structures and arrays) it depends on the [ISymbolLoader](#) [[▶ 2714](#)] settings what will happen. In non dynamic modes: the raw byte Array will be returned, in dynamic mode: A Value will be created on the fly.

Reference

[Symbol Class](#) [[▶ 1863](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.2.21 Symbol.SetParent Method

Sets the parent symbol.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void SetParent(  
    ISymbol? parent  
)
```

Parameters

parent	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The parent.
--------	--

Implements

[IHierarchicalSymbol.SetParent\(ISymbol\)](#) [[▶ 2548](#)]

Reference

[Symbol Class](#) [[▶ 1863](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.2.22 Symbol.ToString Method

Returns a String that represents this instance.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```


Return Value

Type: String
A String that represents this instance.

Reference

[Symbol Class](#) [► 1863]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.23.2.23 Symbol.TryReadValue Method

Reads the Value of the [IValueSymbol](#) [► 2775]

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int TryReadValue(  
    int timeout,  
    out Object?? value  
)
```

Parameters

timeout	Type: System.Int32 The timeout in ms.
value	Type: System.Object. The read value.

Return Value

Type: Int32
The error Code..

Implements

[IValueSymbol.TryReadValue\(Int32, Object.\)](#) [► 2786]

Remarks

Calling on primitive types, a call of this method will return the primitive value. On complex types (structures and arrays) it depends on the [ISymbolLoader](#) [► 2714] settings what will happen. In non dynamic modes: the raw byte Array will be returned, in dynamic mode: A Value will be created on the fly. A negative timeout indicates that the Default Timeout for the communication will be used. 0 means timeout is switched off.

Reference

[Symbol Class](#) [► 1863]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.23.2.24 Symbol.TryWriteValue Method

Writes the Value of the [IValueSymbol](#) [► 2775]

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int TryWriteValue(
    Object value,
    int timeout
)
```

Parameters

value	Type: System.Object The value.
timeout	Type: System.Int32 The timeout in ms.

Return Value

Type: Int32
The error code.

Implements

[IValueSymbol.TryWriteValue\(Object, Int32\)](#) [[▶ 2787](#)]

Remarks

Calling on primitive types, a call of this method will directly write this Value. On complex types (structs and arrays) it depends on the [ISymbolLoader](#) [[▶ 2714](#)] settings what will happen. In non dynamic modes: Only byte Arrays (of correct size) can be written) in dynamic mode: A Value that represents the value will be accepted also. A negative timeout indicates that the Default Timeout for the communication will be used. 0 means timeout is switched off.



Reference

[Symbol Class](#) [[▶ 1863](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.2.25 Symbol.UpdateAnyValue Method

Overload List

	Name	Description
	UpdateAnyValue(Object) [▶ 1907]	Reads the value of this Value [▶ 2775] into the specified managed value.
	UpdateAnyValue(Object, Int32) [▶ 1907]	Reads the value of this Value [▶ 2775] into the specified managed value.

Reference

[Symbol Class](#) [[▶ 1863](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.2.25.1 Symbol.UpdateAnyValue Method (Object.)

Reads the value of this [Value](#) [[▶ 2775](#)] into the specified managed value.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void UpdateAnyValue(  
    ref Object managedObject  
)
```

Parameters

managedObject	Type: System.Object. The managed object.
---------------	---

Return Value

Type:

Read value (System.Object).

Implements

[IValueAnySymbol.UpdateAnyValue\(Object.\)](#) [[▶ 2762](#)]

Reference

[Symbol Class](#) [[▶ 1863](#)]

[UpdateAnyValue Overload](#) [[▶ 1906](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

[IValueAnySymbol.ReadAnyValue\(Type\)](#) [[▶ 2758](#)]

[IValueAnySymbol.WriteAnyValue\(Object\)](#) [[▶ 2764](#)]

6.7.23.2.25.2 Symbol.UpdateAnyValue Method (Object., Int32)

Reads the value of this [Value](#) [[▶ 2775](#)] into the specified managed value.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void UpdateAnyValue(  
    ref Object managedObject,  
    int timeout  
)
```

Parameters

managedObject	Type: System.Object. The managed object.
timeout	Type: System.Int32 The timeout.

Return Value

Type:
Read value (System.Object).

Implements

[IValueAnySymbol.UpdateAnyValue\(Object., Int32\)](#) [► 2763]

Reference

[Symbol Class](#) [► 1863]



[UpdateAnyValue Overload](#) [► 1906]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

[IValueAnySymbol.ReadAnyValue\(Type\)](#) [► 2758]

[IValueAnySymbol.WriteAnyValue\(Object\)](#) [► 2764]

6.7.23.2.26 Symbol.WriteRawValue Method**Overload List**

	Name	Description
	WriteRawValue(Byte e.) [► 1908]	Writes the raw value of the IValueSymbol [► 2775] (Ads Read / Write)
	WriteRawValue(Byte e., Int32) [► 1909]	Writes the raw value of the IValueSymbol [► 2775] (Ads Read / Write)

Reference

[Symbol Class](#) [► 1863]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.23.2.26.1 Symbol.WriteRawValue Method (.Byte.)

Writes the raw value of the [IValueSymbol](#) [► 2775] (Ads Read / Write)

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public void WriteRawValue(
    byte[] rawValue
)
```

Parameters

rawValue	Type: .System.Byte. The value as byte array.
----------	---

Field Value

Type:
The value.

Implements

[IValueRawSymbol.WriteRawValue\(Byte.\)](#) [[2772](#)]

Reference

[Symbol Class](#) [[1863](#)]

[WriteRawValue Overload](#) [[1908](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[1698](#)]

6.7.23.2.26.2 Symbol.WriteRawValue Method (.Byte., Int32)

Writes the raw value of the [IValueSymbol](#) [[2775](#)] (Ads Read / Write)

Namespace: [TwinCAT.Ads.TypeSystem](#) [[1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public void WriteRawValue(
    byte[] rawValue,
    int timeout
)
```

Parameters

rawValue	Type: <code>.System.Byte.</code> The raw value.
timeout	Type: <code>System.Int32</code> The timeout.

Field Value

Type:
The raw value.

Implements

[IValueRawSymbol.WriteRawValue\(Byte., Int32\)](#) [[2773](#)]

Remarks

A negative timeout indicates that the Default Timeout for the communication will be used. 0 means timeout is switched off.

Reference

[Symbol Class](#) [[1863](#)]

[WriteRawValue Overload](#) [[1908](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[1698](#)]

6.7.23.2.27 **Symbol.WriteRawValueAsync Method**

Writes the raw value of the [IValueSymbol](#) [▶ 2775] (Ads Read / Write)

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWriteAccess> WriteRawValueAsync (
    byte[] rawValue,
    CancellationToken cancel
)
```

Parameters

rawValue	Type: <code>.System.Byte</code> . The value as byte array.
cancel	Type: <code>System.Threading.CancellationToken</code> The cancellation token.

Return Value

Type: [Task.ResultWriteAccess](#) [▶ 3217].

A task that represents the asynchronous read operation. The [ResultRead](#) [▶ 1143] parameter contains the total number of bytes read into the buffer ([ReadBytes](#) [▶ 1145]) and the [ErrorCode](#) [▶ 1120] after execution..

Implements

[IValueRawSymbol.WriteRawValueAsync\(Byte., CancellationToken\)](#) [▶ 2774]



Reference

[Symbol Class](#) [▶ 1863]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.23.2.28 **Symbol.WriteValue Method**

Overload List

	Name	Description
	WriteValue(Object) [▶ 1910]	Writes the Value of the IValueSymbol [▶ 2775]
	WriteValue(Object, Int32) [▶ 1911]	Writes the Value of the IValueSymbol [▶ 2775]

Reference

[Symbol Class](#) [▶ 1863]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.23.2.28.1 **Symbol.WriteValue Method (Object)**

Writes the Value of the [IValueSymbol](#) [▶ 2775]

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteValue(
    Object value
)
```

Parameters

value	Type: System.Object The value.
-------	-----------------------------------

Implements

[IValueSymbol.WriteValue\(Object\)](#) [[▶ 2788](#)]

Remarks

Calling on primitive types, a call of this method will directly write this Value. On complex types (structs and arrays) it depends on the [ISymbolLoader](#) [[▶ 2714](#)] settings what will happen. In non dynamic modes: Only byte Arrays (of correct size) can be written) in dynamic mode: A Value that represents the value will be accepted also.

Reference

[Symbol Class](#) [[▶ 1863](#)]

[WriteValue Overload](#) [[▶ 1910](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.23.2.28.2 Symbol.WriteValue Method (Object, Int32)

Writes the Value of the [IValueSymbol](#) [[▶ 2775](#)]

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteValue(
    Object value,
    int timeout
)
```

Parameters

value	Type: System.Object The value.
timeout	Type: System.Int32 The timeout in ms.

Implements

[IValueSymbol.WriteValue\(Object, Int32\)](#) [[▶ 2788](#)]

Remarks

Calling on primitive types, a call of this method will directly write this Value. On complex types (structs and arrays) it depends on the [ISymbolLoader \[▶ 2714\]](#) settings what will happen. In non dynamic modes: Only byte Arrays (of correct size) can be written) in dynamic mode: A Value that represents the value will be accepted also. A negative timeout indicates that the Default Timeout for the communication will be used. 0 means timeout is switched off.

Reference

[Symbol Class \[▶ 1863\]](#)

[WriteValue Overload \[▶ 1910\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.23.2.29 Symbol.WriteValueAsync Method

Writes the Value of the [IValueSymbol \[▶ 2775\]](#)

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultWriteAccess> WriteValueAsync (
    Object value,
    CancellationToken cancel
)
```

Parameters

value	Type: System.Object The value.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: Task.[ResultWriteAccess \[▶ 3217\]](#).

A tasks that represents the asynchronous 'ReadValue' operation. The read result is stored in the [ResultWriteAccess \[▶ 3217\]](#) return value and contains the [ErrorCode \[▶ 3202\]](#).

Implements

[IValueSymbol.WriteValueAsync\(Object, CancellationToken\) \[▶ 2789\]](#)

Remarks

Calling on primitive types, a call of this method will directly write this Value. On complex types (structs and arrays) it depends on the [ISymbolLoader \[▶ 2714\]](#) settings what will happen. In non dynamic modes: Only byte Arrays (of correct size) can be written) in dynamic mode: A Value that represents the value will be accepted also.

Reference



[Symbol Class \[▶ 1863\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.23.3 Symbol Events

The [Symbol](#) [[1863](#)] type exposes the following members.

Events

	Name	Description
	RawValueChanged [1913]	Occurs when the RawValue of the IValueSymbol [2775] has changed.
	ValueChanged [1913]	Occurs when the (Primitive) value of the IValueSymbol [2775] has changed.

Reference

[Symbol Class](#) [[1863](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[1698](#)]

6.7.23.3.1 Symbol.RawValueChanged Event

Occurs when the RawValue of the [IValueSymbol](#) [[2775](#)] has changed.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<RawValueChangedEventArgs> RawValueChanged
```

Value

Type: System.EventHandler.[RawValueChangedEventArgs](#) [[2813](#)].

Implements

[IValueRawSymbol.RawValueChanged](#) [[2774](#)]

Reference

[Symbol Class](#) [[1863](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[1698](#)]

6.7.23.3.2 Symbol.ValueChanged Event

Occurs when the (Primitive) value of the [IValueSymbol](#) [[2775](#)] has changed.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<ValueChangedEventArgs> ValueChanged
```

Value

Type: System.EventHandler.ValueChangedEventArgs [▶ 2971].

Implements

IValueSymbol.ValueChanged [▶ 2790]

Reference





Symbol Class [▶ 1863]

TwinCAT.Ads.TypeSystem Namespace [▶ 1698]

6.7.23.4 Symbol Operators

The Symbol [▶ 1863] type exposes the following members.

Operators

	Name	Description
 	Equality [▶ 1914]	Operator==
 	Inequality [▶ 1915]	Implements the != operator.

Reference

Symbol Class [▶ 1863]

TwinCAT.Ads.TypeSystem Namespace [▶ 1698]

6.7.23.4.1 Symbol.Equality Operator

Operator==

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static bool operator ==(
    Symbol? o1,
    Symbol? o2
)
```

Parameters

o1	Type: TwinCAT.Ads.TypeSystem.Symbol [▶ 1863] The o1.
o2	Type: TwinCAT.Ads.TypeSystem.Symbol [▶ 1863] The o2.

Return Value

Type: Boolean
The result of the operator.

Reference

[Symbol Class](#) [► 1863]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.23.4.2 Symbol.Inequality Operator

Implements the != operator.

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static bool operator !=(
    Symbol? o1,
    Symbol? o2
)
```

Parameters

o1	Type: TwinCAT.Ads.TypeSystem.Symbol [► 1863] The o1.
o2	Type: TwinCAT.Ads.TypeSystem.Symbol [► 1863] The o2.

Return Value

Type: Boolean
The result of the operator.

Reference


[Symbol Class](#) [► 1863]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.23.5 Symbol Fields

The [Symbol](#) [► 1863] type exposes the following members.

Fields

	Name	Description
	syncObject [► 1916]	Synchronization object

Reference

[Symbol Class](#) [► 1863]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.23.5.1 Symbol.syncObject Field

Synchronization object

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected Object syncObject
```

Field Value

Type: Object

Reference

[Symbol Class](#) [[▶ 1863](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.24 SymbolIterator Class

Iterator class for enumerations of [Symbols](#) [[▶ 2691](#)].

Inheritance Hierarchy

System.Object

[TwinCAT.TypeSystem.Generic.SymbolIterator](#) [[▶ 3172](#)].[ISymbol](#) [[▶ 2691](#)].

[TwinCAT.Ads.TypeSystem.SymbolIterator](#)

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229




Syntax



C#

```
public class SymbolIterator : SymbolIterator<ISymbol>
```



The SymbolIterator type exposes the following members.

Constructors








	Name	Description
	SymbolIterator (IInstanceCollection . ISymbol .) [▶ 1919]	Initializes a new instance of the SymbolIterator class.
	SymbolIterator (IEnumerable . ISymbol ., Boolean) [▶ 1920]	Initializes a new instance of the SymbolIterator class.
	SymbolIterator (IInstanceCollection . ISymbol ., Func . ISymbol ., Boolean .) [▶ 1919]	Initializes a new instance of the SymbolIterator class.

	Name	Description
	SymbolIterator(IEnumerable.ISymbol, Boolean, Func.ISymbol, Boolean.) [▶ 1920]	Initializes a new instance of the SymbolIterator class.
	SymbolIterator(IEnumerable.ISymbol, Boolean, SymbolIterationMask, Func.ISymbol, Boolean.) [▶ 1921]	Initializes a new instance of the SymbolIterator class.

Properties

	Name	Description
	Mask [▶ 3178]	Gets or sets the SymbolIterationMask [▶ 3019] (Inherited from SymbolIterator.T. [▶ 3172].)
	SymbolRecursionDetection [▶ 3178]	Gets or sets a value indicating whether the iterator checks for Symbol recursions (true by default). (Inherited from SymbolIterator.T. [▶ 3172].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 3179]	Gets the enumerator that enumerates through a collection (Inherited from SymbolIterator.T. [▶ 3172].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Remarks

This iterator class can be used to iterate over collections of symbol trees (root symbols + sub symbols). By constructor the user can choose if the iterator works recursively within the symbol tree and optionally a filter function to select only specific symbols (predicate).

Examples

The following example shows how to determine, browse and filter symbols.

Browsing and filtering Symbols

```

using (AdsClient client = new AdsClient())
{
    CancellationToken cancel = CancellationToken.None;

    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);

    // Load all Symbols + DataTypes
    ISymbolLoader loader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);

    ResultSymbols resultSymbols = await loader.GetSymbolsAsync(cancel);

    if (resultSymbols.Succeeded)
    {
        Symbol symbol = (Symbol)resultSymbols.Symbols["MAIN.nCounter"];

        // Works for ALL Primitive 'ANY TYPES' Symbols
        ResultWriteAccess resultWrite = await symbol.WriteValueAsync(valueToWrite, cancel);
        ResultReadValueAccess resultRead = await symbol.ReadValueAsync(cancel);

        if (resultRead.Succeeded)
            valueToRead = (uint)resultRead.Value;

        // Simple filtering of Symbols
        Regex filterExpression = new Regex(pattern: @"^MAIN.*"); // Everything that starts with "MAIN"

        // FilterFunction that filters for the InstancePath
        Func<ISymbol, bool> filter = s => filterExpression.IsMatch(s.InstancePath);
        SymbolIterator iterator = new SymbolIterator(symbols: resultSymbols.Symbols, recurse: true, selector: filter);

        foreach (ISymbol filteredSymbol in iterator)
        {
            Console.WriteLine(filteredSymbol.InstancePath);
        }
    }
}





```


Reference

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.24.1 SymbolIterator Constructor

Overload List

	Name	Description
	SymbolIterator(IInstanceCollection, ISymbol.) [► 1919]	Initializes a new instance of the SymbolIterator [► 1916] class.
	SymbolIterator(IEnumerable, ISymbol, Boolean) [► 1920]	Initializes a new instance of the SymbolIterator [► 1916] class.
	SymbolIterator(IInstanceCollection, ISymbol, Func<ISymbol, Boolean>) [► 1919]	Initializes a new instance of the SymbolIterator [► 1916] class.
	SymbolIterator(IEnumerable, ISymbol,	Initializes a new instance of the SymbolIterator [► 1916] class.

	Name	Description
	Boolean, Func.ISymbol, Boolean.) [▶_1920]	
	SymbolIterator(ICollection.IEnumerable.ISymbol, Boolean, SymbolIterationMask, Func.ISymbol, Boolean., Func.ISymbol, Boolean.) [▶_1921]	Initializes a new instance of the SymbolIterator [▶_1916] class.

Reference

[SymbolIterator Class \[\[▶_1916\]\(#\)\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[\[▶_1698\]\(#\)\]](#)

6.7.24.1.1 SymbolIterator Constructor (ICollection.ISymbol.)

Initializes a new instance of the [SymbolIterator \[\[▶_1916\]\(#\)\]](#) class.

Namespace: [TwinCAT.Ads.TypeSystem \[\[▶_1698\]\(#\)\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolIterator(
    ICollection<ISymbol> symbols
)
```

Parameters

symbols	Type: TwinCAT.TypeSystem.ICollection [▶_2554].ISymbol [▶_2691] . The symbol collection
---------	---

Reference

[SymbolIterator Class \[\[▶_1916\]\(#\)\]](#)

[SymbolIterator Overload \[\[▶_1918\]\(#\)\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[\[▶_1698\]\(#\)\]](#)

6.7.24.1.2 SymbolIterator Constructor (ICollection.ISymbol., Func.ISymbol, Boolean.)

Initializes a new instance of the [SymbolIterator \[\[▶_1916\]\(#\)\]](#) class.

Namespace: [TwinCAT.Ads.TypeSystem \[\[▶_1698\]\(#\)\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolIterator(
    IEnumerable<ISymbol> symbols,
    Func<ISymbol, bool>? selector
)
```

Parameters

symbols	Type: TwinCAT.TypeSystem.IInstanceCollection [▶ 2554]. ISymbol [▶ 2691]. The symbol collection.
selector	Type: System.Func.ISymbol [▶ 2691], Boolean. Predicate function indicating that symbols are visible for the iteration. The default value null returns all symbols (of the specified mask).

Reference

[SymbolIterator Class](#) [[▶ 1916](#)]

[SymbolIterator Overload](#) [[▶ 1918](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.24.1.3 SymbolIterator Constructor (IEnumerable.ISymbol., Boolean)

Initializes a new instance of the [SymbolIterator](#) [[▶ 1916](#)] class.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolIterator(
    IEnumerable<ISymbol> symbols,
    bool recurse
)
```

Parameters

symbols	Type: System.Collections.Generic.IEnumerable.ISymbol [▶ 2691]. The symbol enumeration.
recurse	Type: System.Boolean if set to true, the iterator works recursively over all subsymbols.

Reference

[SymbolIterator Class](#) [[▶ 1916](#)]

[SymbolIterator Overload](#) [[▶ 1918](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.24.1.4 SymbolIterator Constructor (IEnumerable.ISymbol., Boolean, Func.ISymbol, Boolean.)

Initializes a new instance of the [SymbolIterator](#) [[▶ 1916](#)] class.

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolIterator(
    IEnumerable<ISymbol> symbols,
    bool recurse,
    Func<ISymbol, bool>? selector
)
```

Parameters

symbols	Type: System.Collections.Generic.IEnumerable. ISymbol [▸ 2691] . The symbol collection.
recurse	Type: System.Boolean if set to true, the iterator works recursively over all subsymbols.
selector	Type: System.Func. ISymbol [▸ 2691] , Boolean. Predicate function indicating that symbols are visible for the iteration. The default value null returns all symbols (of the specified mask).

Reference

[SymbolIterator Class \[▸ 1916\]](#)

[SymbolIterator Overload \[▸ 1918\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.24.1.5 SymbolIterator Constructor (IEnumerable.ISymbol., Boolean, SymbolIterationMask, Func.ISymbol, Boolean., Func.ISymbol, Boolean.)

Initializes a new instance of the [SymbolIterator \[▸ 1916\]](#) class.

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolIterator(
    IEnumerable<ISymbol> symbols,
    bool recurse,
    SymbolIterationMask mask,
    Func<ISymbol, bool>? selector,
    Func<ISymbol, bool>? areChildsIterated
)
```

Parameters

symbols	Type: System.Collections.Generic.IEnumerable. ISymbol [▸ 2691] . The symbol collection.
recurse	Type: System.Boolean if set to true, the iterator works recursively over all subsymbols.
mask	Type: TwinCAT.TypeSystem.SymbolIterationMask [▸ 3019] Specifies a mask over the symbols, that filters out specific symbol categories. The default is All [▸ 3019] and all symbols are shown.

selector	Type: System.Func.ISymbol [▶ 2691], Boolean. Predicate function indicating that symbols are visible for the iteration. The default value null returns all symbols (of the specified mask).
areChildsIterated	Type: System.Func.ISymbol [▶ 2691], Boolean. Predicate function indicating that child of the actual symbol should be iterated (in recurse mode). The default value iterates all child (of the specified mask).

Reference

[SymbolIterator Class \[▶ 1916\]](#)



[SymbolIterator Overload \[▶ 1918\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.24.2 SymbolIterator Properties

The [SymbolIterator \[▶ 1916\]](#) type exposes the following members.

Properties

	Name	Description
	Mask [▶ 3178]	Gets or sets the SymbolIterationMask [▶ 3019] (Inherited from SymbolIterator.T. [▶ 3172].)
	SymbolRecursionDetection [▶ 3178]	Gets or sets a value indicating whether the iterator checks for Symbol recursions (true by default). (Inherited from SymbolIterator.T. [▶ 3172].)

Reference








[SymbolIterator Class \[▶ 1916\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.24.3 SymbolIterator Methods

The [SymbolIterator \[▶ 1916\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 3179]	Gets the enumerator that enumerates through a collection (Inherited from SymbolIterator.T. [▶ 3172].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[SymbolIterator Class \[► 1916\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.25 SymbolLoaderFactory Class

The class SymbolLoaderFactory is used to create a new instance of the AdsSymbolLoader initialized to the parametrized mode (SymbolBrowser V2, new Version)

Inheritance Hierarchy

System.Object

 TwinCAT.Ads.TypeSystem.SymbolLoaderFactory

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229




Syntax

C#

```
public static class SymbolLoaderFactory
```

The SymbolLoaderFactory type exposes the following members.

Methods

	Name	Description
	Create [► 1924]	Creates the specified connection.
		
		

Remarks

The Symbol Loader (V2) supports the following [modes \[► 159\]](#). [Flat \[► 159\]](#)The flat mode organizes the Symbols in a flat list. This mode is available in all .NET versions. [VirtualTree \[► 159\]](#)The virtual tree mode organizes the Symbols hierarchically with parent-child relationships. This mode is available in all .NET Versions. [DynamicTree \[► 159\]](#)The Dynamic tree mode organizes the Symbols hierarchically and (dynamically) creates struct members, array elements and enum fields on the fly. This feature is only available on platforms that support the Dynamic Language Runtime (DLR), actually all .NET Framework Version larger than 4.0. Virtual instances means, that all Symbols are ordered within a tree structure. For that symbol nodes that are not located on a fixed address, a Virtual Symbol will be created. Setting the virtualInstance parameter to 'false' means, that the located symbols will be returned in a flattened list.

Reference




[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

[TwinCAT.SymbolLoaderSettings \[► 148\]](#)

6.7.25.1 SymbolLoaderFactory Methods

The [SymbolLoaderFactory \[► 1923\]](#) type exposes the following members.

Methods

	Name	Description
  	Create [▶ 1924]	Creates the specified connection.

Reference

[SymbolLoaderFactory Class](#) [[▶ 1923](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.25.1 SymbolLoaderFactory.Create Method

Creates the specified connection.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ISymbolLoader Create(
    IConnection connection,
    ISymbolLoaderSettings settings
)
```

Parameters

connection	Type: TwinCAT.IConnection [▶ 79] The connection.
settings	Type: TwinCAT.ISymbolLoaderSettings [▶ 104] The settings.

Return Value

Type: [ISymbolLoader](#) [[▶ 2714](#)]

ISymbolLoader.

Examples

The following sample shows how to create a dynamic version of the SymbolLoader V2. The dynamic symbol loader makes use of the Dynamic Language Runtime (DLR) of the .NET Framework. That means Structures, Arrays and Enumeration types and instances are generated 'on-the-fly' during symbol Browsing. These created dynamic objects are a one to one representation of the Symbol Server target objects (e.g the IEC61131 types on the PLC). Dynamic language features are only available from .NET4 upwards.

Dynamic Tree Mode

```
namespace Sample
{
    using System;
    using System.Diagnostics;
    using System.Threading;
    using TwinCAT;
    using TwinCAT.Ads;
    using TwinCAT.Ads.TypeSystem;
    using TwinCAT.Ads.ValueAccess;
    using TwinCAT.TypeSystem;
    using TwinCAT.TypeSystem.Generic;
    using TwinCAT.ValueAccess;
```

```

class SymbolBrowserProgramV2DynamicTree
{

#region CODE_SAMPLE_SIMPLEDYNAMIC
/// <summary>
/// Defines the entry point of the application.
/// </summary>
/// <param name="args">The arguments.</param>
static async void Main(string[] args)
{
    // Get the AdsAddress from command-line arguments
    AmsAddress address = ArgParser.Parse(args);

    CancellationTokenSource cancelSource = new CancellationTokenSource();
    CancellationToken cancel = cancelSource.Token;

    using (AdsClient client = new AdsClient())
    {
        // Connect to the target device
        client.Connect(address);

        // Usage of "dynamic" Type and Symbols (>= .NET4 only)
        SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.DynamicTree);
        IAdsSymbolLoader dynLoader = (IAdsSymbolLoader)SymbolLoaderFactory.Create(client, settings);

#endregion

        // Set the Default setting for Notifications
        dynLoader.DefaultNotificationSettings = new NotificationSettings(AdsTransMode.OnChange, 200,
2000);

        // Get the Symbols (Dynamic Symbols)
        var resultSymbols = await ((IDynamicSymbolLoader) dynLoader).GetDynamicSymbolsAsync(cancel);

        dynamic dynamicSymbols = resultSymbols.Symbols;
        dynamic adsPort = dynamicSymbols.TwinCAT_SystemInfoVarList._AppInfo.AdsPort;

#region CODE_SAMPLE_SIMPLEDYNAMIC

        // Access Main Symbol with Dynamic Language Runtime support (DLR)
        // Dynamically created property "Main"
        //dynamic symMain = dynamicSymbols.Main;

        // Main is an 'VirtualSymbol' / Organizational unit that doesn't have a value
        // Calling ReadValue is not allowed
        //bool test = symMain.HasValue;
        //dynamic invalid = symMain.ReadValue();

        //Reading TaskInfo Value
        //
        With calling ReadValueAsync() a 'snapshot' of the Symbols Instance is taken (reading async)
        ResultReadValueAccess resultRead = await dynamicSymbols.TwinCAT_SystemInfoVarList._TaskInfo.
ReadValueAsync(cancel);
        dynamic vTaskInfoArray = resultRead.Value;

        // Getting the Snapshot time in UTC format
        DateTimeOffset timeStamp1 = vTaskInfoArray.TimeStamp;

        // Getting TaskInfo Symbol for Task 1
        dynamic symTaskInfo1 = dynamicSymbols.TwinCAT_SystemInfoVarList._TaskInfo[1];

        // Getting CycleCount Symbol
        dynamic symCycleCount = symTaskInfo1.CycleCount;

        // Take Snapshot value of the ApplicationInfo struct
        resultRead = await dynamicSymbols.TwinCAT_SystemInfoVarList._AppInfo.ReadValueAsync(cancel);
        dynamic vAppInfo = resultRead.Value;

        // Get the UTC Timestamp of the snapshot
        DateTimeOffset timeStamp2 = vAppInfo.TimeStamp;

        // Access the ProjectName of the ApplicationInfo Snapshot (type-safe!)
        string projectNameValue = vAppInfo.ProjectName;

        // Reading the CycleCount Value
        resultRead = await symTaskInfo1.CycleCount.ReadValueAsync(cancel); // Taking a Value Sna
pshot
        int cycleCountValue = (int)resultRead.Value;

```

```

        #endregion

        // Registering for dynamic "ValueChanged" events for the Values
        // Using Default Notification settings
        symCycleCount.ValueChanged += new EventHandler<ValueChangedEventArgs>(cycleCount_ValueChanged);

        // Override default notification settings
        symTaskInfo1.NotificationSettings = new NotificationSettings(AdsTransMode.Cyclic, 500, 0);

        // Register for ValueChanged event.
        symTaskInfo1.ValueChanged += new EventHandler<ValueChangedEventArgs>(taskInfo1Value_ValueChanged); // Struct Type

        Thread.Sleep(10000); // Sleep main thread for 10 Seconds
    }
    Console.WriteLine("CycleCount Changed events received: {0}", _cycleCountEvents);
    Console.WriteLine("taskInfo1 Changed events received: {0}", _taskInfo1Events);

    Console.WriteLine("");
    Console.WriteLine("Press [Enter] for leave:");
    Console.ReadLine();
}

static object _notificationSynchronizer = new object();
static int _cycleCountEvents = 0;

/// <summary>
/// Handler function for the CycleCount ValueChanged event.
/// </summary>
/// <param name="sender">The sender.</param>
/// <param name="e">The e.</param>
static void cycleCount_ValueChanged(object sender, ValueChangedEventArgs e)
{
    lock(_notificationSynchronizer)
    {
        Interlocked.Increment(ref _cycleCountEvents);
        // val is a type safe value of int!
        dynamic val = e.Value;
        uint intVal = val;

        DateTimeOffset changedTime = e.DateTime.ToLocalTime(); // Convert UTC to local time
        Console.WriteLine("CycleCount changed to: {0}, TimeStamp: {1}", intVal, changedTime.ToString(
("HH:mm:ss:fff"));
    }
}

static int _taskInfo1Events = 0;

/// <summary>
/// Handler function for the TaskInfo ValueChanged event.
/// </summary>
/// <param name="sender">The sender.</param>
/// <param name="e">The e.</param>
static void taskInfo1Value_ValueChanged(object sender, ValueChangedEventArgs e)
{
    lock (_notificationSynchronizer)
    {
        Interlocked.Increment(ref _taskInfo1Events);
        dynamic val = e.Value;
        DateTimeOffset changedTime = e.DateTime.ToLocalTime(); // Convert to local time

        // Val is a during Runtime created struct type and contains
        // the same Properties as related PLC object.
        int cycleTime = val.CycleTime;
        Console.WriteLine("TaskInfo1Value changed TimeStamp: {0}", changedTime.ToString("HH:mm:ss:fff"));
    }
}
}

```

The following sample shows how to create a static (non dynamic) version of the SymbolLoader V2. The static symbol loader in version 2 is a nearly code compatible version of the Dynamic Loader, only the dynamic creation of objects is not available. The reason for supporting this mode is that .NET Framework Versions lower than Version 4.0 (CLR2) doesn't support the Dynamic Language Runtime (DLR). The SymbolLoader V2 static object is supported from .NET 2.0 on.

Virtual Tree Mode

```

using System;
using System.Threading;
using System.Diagnostics;
using TwinCAT;
using TwinCAT.Ads;
using TwinCAT.TypeSystem;
using TwinCAT.TypeSystem.Generic;
using TwinCAT.Ads.ValueAccess;
using TwinCAT.Ads.TypeSystem;
using TwinCAT.ValueAccess;

namespace Sample
{
    class SymbolBrowserProgramV2VirtualTree
    {
        /// <summary>
        /// Defines the entry point of the application.
        /// </summary>
        /// <param name="args">The arguments.</param>
        static void Main(string[] args)
        {
            ConsoleLogger logger = new ConsoleLogger();

            Console.WriteLine("");
            Console.WriteLine("Press [Enter] for start:");
            Console.ReadLine();

            //logger.Active = false;

            Stopwatch stopper = new Stopwatch();

            // Parse the command-line arguments
            AmsAddress address = ArgParser.Parse(args);

            stopper.Start();

            using (AdsClient client = new AdsClient())
            {
                //client.Synchronize = false;

                // Connect the AdsClient to the device target.
                client.Connect(address);

                // Creates the Symbol Objects as hierarchical tree
                SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.VirtualTree, ValueAccessMode.IndexGroupOffsetPreferred);
                ISymbolLoader symbolLoader = SymbolLoaderFactory.Create(client, settings);

                // Dump Datatypes from Target Device
                Console.WriteLine(string.Format("Dumping '{0}' DataTypes:", symbolLoader.DataTypes.Count));
                foreach (IDataType type in symbolLoader.DataTypes)
                {
                    logger.DumpType(type);
                }
                Console.WriteLine("");

                // Dump Symbols from target device
                Console.WriteLine("Dumping '{0}' Symbols:", symbolLoader.Symbols.Count);
                foreach (ISymbol symbol in symbolLoader.Symbols)
                {
                    logger.DumpSymbol(symbol, 0);
                }
            }
            stopper.Stop();
            TimeSpan elapsed = stopper.Elapsed;

            Console.WriteLine("");
            Console.WriteLine("Browsing complete tree: {0},
            ({1} DataTypes, {2} Symbols)", elapsed, logger.DataTypesCount, logger.SymbolsCount);
            Console.WriteLine("Press [Enter] for leave:");
            Console.ReadLine();
        }
    }
}

```

Examples

The SymbolLoader V2 static object is supported from .NET 2.0 on.

Flat Mode

```

using System;
using System.Diagnostics;
using System.Threading;
using TwinCAT;
using TwinCAT.Ads;
using TwinCAT.Ads.TypeSystem;
using TwinCAT.Ads.ValueAccess;
using TwinCAT.TypeSystem;
using TwinCAT.TypeSystem.Generic;
using TwinCAT.ValueAccess;

namespace Sample
{
    class SymbolBrowserProgramV2Flat
    {
        /// <summary>
        /// Defines the entry point of the application.
        /// </summary>
        /// <param name="args">The arguments.</param>
        static void Main(string[] args)
        {
            ConsoleLogger logger = new ConsoleLogger();

            Console.WriteLine("");
            Console.WriteLine("Press [Enter] for start:");
            Console.ReadLine();

            //logger.Active = false;

            Stopwatch stopper = new Stopwatch();

            // Parse the command line arguments
            AmsAddress address = ArgParser.Parse(args);

            stopper.Start();

            // Create the ADS Client
            using (AdsClient client = new AdsClient())
            {
                //client.Synchronize = false;

                // Connect to Address
                client.Timeout = 30000;
                client.Connect(address);

                // Creates the Symbol Objects in Flat Mode (Flat list)
                SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.Flat, ValueAccessMo
                de.IndexGroupOffsetPreferred);
                ISymbolLoader symbolLoader = SymbolLoaderFactory.Create(client, settings);

                // Dump Datatypes from Target Device
                Console.WriteLine(string.Format("Dumping '{0}' DataTypes:", symbolLoader.DataTypes.Count));
                foreach (IDataType type in symbolLoader.DataTypes)
                {
                    logger.DumpType(type);
                }

                Console.WriteLine("");

                // Dump Symbols from target device
                Console.WriteLine("Dumping '{0}' Symbols:", symbolLoader.Symbols.Count);
                foreach (ISymbol symbol in symbolLoader.Symbols)
                {
                    logger.DumpSymbol(symbol, 0);
                }
            }
            stopper.Stop();
            TimeSpan elapsed = stopper.Elapsed;

            Console.WriteLine("");
            Console.WriteLine("Browsing complete tree: {0},
            ({1} DataTypes, {2} Symbols)", elapsed, logger.DataTypesCount, logger.DataTypesCount);
            Console.WriteLine("Press [Enter] for leave:");
            Console.ReadLine();
        }
    }
}

```


Examples

Argument Parser

```
public static class ArgParser
{
    /// <summary>
    /// Parses the arguments.
    /// </summary>
    /// <param name="args">The arguments.</param>
    /// <returns>AmsAddress.</returns>
    public static AmsAddress Parse(string[] args)
    {
        AmsNetId netId = AmsNetId.Local;
        int port = 851;

        if (args != null)
        {
            if (args.Length > 0 && args[0] != null)
                netId = AmsNetId.Parse(args[0]);

            if (args.Length > 1 && args[1] != null)
                port = int.Parse(args[1]);
        }
        return new AmsAddress(netId, port);
    }
}
```

Dumping Symbols

```
/// <summary>
/// Console logger
/// </summary>
public class ConsoleLogger
{
    public ConsoleLogger()
    {
    }
    bool _active = true;

    /// <summary>
    /// Gets or sets a value indicating whether this ConsoleLogger is active.
    /// </summary>
    /// <value><c>>true</c> if active; otherwise, <c>>false</c>.</value>
    public bool Active
    {
        get { return _active; }
        set
        {
            _active = value;
        }
    }

    int _dataTypes = 0;

    /// <summary>
    /// Gets the number of dumped dataTypes.
    /// </summary>
    /// <value>The data types count.</value>
    public int DataTypesCount
    {
        get { return _dataTypes; }
    }

    int _symbols = 0;

    /// <summary>
    /// Gets the number of dumped symbols
    /// </summary>
    /// <value>The symbols count.</value>
    public int SymbolsCount
    {
        get { return _symbols; }
    }

    /// <summary>
    /// Dumps the data type.
    /// </summary>
}
```

```

    /// <param name="dataType">Data Type.</param>
    public void DumpType(IDataType dataType)
    {
        WriteLine(string.Format("DataType: {0}, Category: {1}, Size: {2}", dataType.Name, dataType.Categ
ory, dataType.Size));

        switch (dataType.Category)
        {
            case DataTypeCategory.Alias:
                IAliasType alias = (IAliasType)dataType;
                WriteLine(GetPrefix(1) + string.Format("Alias BaseType: {0}", alias.BaseTypeName));
                break;

            case DataTypeCategory.Enum:

                //IEnumType<ushort> enumType = (IEnumType<ushort>)dataType;
                IEnumType enumType = (IEnumType)dataType;
                WriteLine(GetPrefix(1) + string.Format("Enum BaseType: {0}", enumType.BaseTypeName));

                foreach (IEnumValue enumValue in enumType.EnumValues)
                {
                    WriteLine(GetPrefix(2) + string.Format("Name: {0}, Value: {1}", enumValue.Name, enumValu
e.Primitive));
                }
                break;
            case DataTypeCategory.Array:

                IArrayType arrayType = (IArrayType)dataType;
                int i = 0;

                foreach (IDimension dim in arrayType.Dimensions)
                {
                    WriteLine(GetPrefix(2) + string.Format("{0}: LowerBound: {1}, Elements: {2}", i+
, dim.LowerBound, dim.ElementCount));
                }
                break;
            case DataTypeCategory.Struct:
                IStructType structType = (IStructType)dataType;

                foreach (IMember member in structType.Members)
                {
                    WriteLine(GetPrefix(2) + string.Format("Offset {0}: Name: {1}, Type: {2}", member.Offset
, member.InstanceName, member.TypeName));
                }
                break;
            default:
                break;
        }

        foreach (ITypeAttribute attribute in dataType.Attributes)
        {
            WriteLine(GetPrefix(1) + string.Format("{0} : {1} }}", attribute.Name, attribute.Value));
        }
        if (!string.IsNullOrEmpty(dataType.Comment))
        {
            WriteLine(GetPrefix(1) + string.Format("Comment: {0}", dataType.Comment));
        }

        IRpcCallableType rpcCallable = dataType as IRpcCallableType;

        if (rpcCallable != null)
        {
            foreach (IRpcMethod rpcMethod in rpcCallable.RpcMethods)
            {
                if (string.IsNullOrEmpty(rpcMethod.Comment))
                    WriteLine(GetPrefix(1) + string.Format("Method: {0}", rpcMethod));
                else
                    WriteLine(GetPrefix(1) + string.Format("Method: {0}, Comment: {1}", rpcMethod, rpcMethod
.Comment));
            }
        }
        _dataTypes++;
    }

    ///// <summary>
    ///// Dumps the Datatype to Console
    ///// </summary>
    ///// <param name="dataType">DataType.</param>
    //public void DumpType(ITcAdsDataType dataType)
    //{

```

```

// // Dump the Attributes (PLC Metadata)
// foreach (ITypeAttribute attribute in dataType.Attributes)
// {
//     WriteLine(GetPrefix(1) + string.Format("{ {0} : {1} }", attribute.Name, attribute.Value)
);
// }

//     WriteLine(string.Format("DataType: {0}, Category: {1}, Size: {2}", dataType.Name, dataType
.Category, dataType.Size));

//     if (dataType.BaseType != null)
//     {
//         WriteLine(GetPrefix(1) + string.Format("BaseType: {0}", dataType.BaseType));
//     }

//     switch (dataType.Category)
//     {
//         case DataTypeCategory.Enum:
//             foreach (IEnumValue enumValue in dataType.EnumValues)
//             {
//                 WriteLine(GetPrefix(2) + string.Format("Name: {0}, Value: {1}", enumValue.Name, enumVa
lue.Primitive));
//             }
//             break;
//         case DataTypeCategory.Array:
//             int i = 0;
//             foreach (IDimension dim in dataType.Dimensions)
//             {
//                 WriteLine(GetPrefix(2) + string.Format("{0}: LowerBound: {1}, Elements: {2}", i+
+, dim.LowerBound, dim.ElementCount));
//             }
//             break;
//         case DataTypeCategory.Struct:
//             foreach (ITcAdsSubItem subItem in dataType.SubItems)
//             {
//                 WriteLine(GetPrefix(2) + string.Format("Offset {0}: Name: {1}, Type: {2}", subItem.Off
set, subItem.SubItemName, subItem.Name));
//             }
//             break;
//         default:
//             break;
//     }
//     _dataTypes++;
// }

/// <summary>
/// Dump Symbol
/// </summary>
/// <param name="symbol">The symbol.</param>
/// <param name="level">Output indentation level</param>
public void DumpSymbol(ISymbol symbol, int level)
{
    IDataTypes type = symbol.DataType as IDataTypes;

    foreach (ITypeAttribute attribute in symbol.Attributes)
    {
        WriteLine(GetPrefix(level) + string.Format("{ {0} : {1} }", attribute.Name, attribute.Valu
e));
    }

    WriteLine(GetPrefix(level) + string.Format("{0} : {1} (IG: 0x{2} IO: 0x{3} size:
{4})", symbol.InstanceName, symbol.TypeName, ((IAdsSymbol)symbol).IndexGroup.ToString("x"), ((IAdsSy
mbol)symbol).IndexOffset.ToString("x"), symbol.Size));

    if (symbol.Category == DataTypeCategory.Array)
    {
        IArrayInstance arrInstance = (IArrayInstance)symbol;
        IArrayType arrType = (IArrayType)symbol.DataType;

        int count = 0;
        level++;

        foreach (ISymbol arrayElement in arrInstance.Elements)
        {
            DumpSymbol(arrayElement, level);
            count++;
        }

        if (count > 20) // Write only the first 20 to limit output
            break;
    }
}

```

```

}
else if (symbol.Category == DataTypeCategory.Struct)
{
    IStructInstance structInstance = (IStructInstance)symbol;
    IStructType structType = (IStructType)symbol.DataType;

    level++;

    foreach (ISymbol member in structInstance.MemberInstances)
    {
        DumpSymbol(member, level);
    }
}
_symbols++;
}

///// <summary>
///// Dumps the specified Symbol to the Console
///// </summary>
///// <param name="symbol">The symbol.</param>
///// <param name="level">The level.</param>
///// public void DumpSymbol(IAdsSymbol2 symbol, int level)
///// {
//    // Dump Attributes of the Symbol
//    foreach (ITypeAttribute attribute in symbol.Attributes)
//    {
//        WriteLine(GetPrefix(level) + string.Format("{0} : {1} }", attribute.Name, attribute.Va
lue));
//    }

//    ITcAdsSymbolBrowser subSymbolProvider = (ITcAdsSymbolBrowser)symbol;

//    // Dump The Symbol
//    WriteLine(GetPrefix(level) + string.Format("{0} : {1} ({2}, IG: 0x{3} IO: 0x{4} size:
{6} subCount:
{5})", symbol.Name, symbol.TypeName, symbol.DataTypeId, symbol.IndexGroup.ToString("x"), symbol.Inde
xOffset.ToString("x"), subSymbolProvider.SubSymbols.Count, symbol.Size));
//    level++;

//    // Dump all SubSymbols with indentation
//    foreach (IAdsSymbol2 subSymbol in ((ITcAdsSymbolBrowser)symbol).SubSymbols)
//    {
//        DumpSymbol(subSymbol, level);
//    }
//    _symbols++;
//}

/// <summary>
/// Dump namespace.
/// </summary>
/// <param name="ns">The namespace.</param>
public void DumpNamespace(INamespace<IDataType> ns)
{
    WriteLine("Namespace: {0}, DataTypes: {1}", ns.Name, ns.DataTypes.Count);

    foreach (IDataType type in ns.DataTypes)
    {
        DumpType(type);
    }
}

/// <summary>
/// Get the indentation prefix
/// </summary>
/// <param name="level">The level.</param>
/// <returns>System.String.</returns>
public string GetPrefix(int level)
{
    return "".PadLeft(level * 3);
}

/// <summary>
/// Writes a line to the Console
/// </summary>
/// <param name="message">The message.</param>
public void WriteLine(string message)
{
    if (Active)
    {
        Console.WriteLine(message);
    }
}

```

```

    }
}

/// <summary>
/// Writes a line to the console
/// </summary>
/// <param name="format">The format.</param>
/// <param name="args">The arguments.</param>
public void WriteLine(string format, params object[] args)
{
    if (Active)
    {
        Console.WriteLine(format, args);
    }
}
}
}

```

Reference

[SymbolLoaderFactory Class \[▶ 1923\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.26 UnionType Class

Represents a union type

Inheritance Hierarchy

System.Object

[TwinCAT.Ads.TypeSystem.DataType \[▶ 1721\]](#)

[TwinCAT.Ads.TypeSystem.UnionType](#)

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#









```








public sealed class UnionType : DataType,
    IUnionType, IDataType, IBitSize

```





The UnionType type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721] .)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721] .)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721] .)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721] .)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721] .)
	Fields [▶ 1936]	Gets a read only collection of the Fields [▶ 2535] of the IUnionType [▶ 2740] .
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721] .)
	Id [▶ 1729]	Gets the ID of the DataType (Inherited from DataType [▶ 1721] .)

	Name	Description
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721] .)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DataType [▶ 1721] .)
	ManagedType [▶ 1733]	Gets the corresponding .NET Type if attached. (Inherited from DataType [▶ 1721] .)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721] .)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721] .)
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721] .)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721] .)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721] .)

Extension Methods





	Name	Description
	IsArrayOfPrimitives. [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021] .)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021] .)
	IsContainer. [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021] .)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021] .)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021] .)
	IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021] .)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021] .)

	Name	Description
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]









Also see about this








-  [DataType.IsContainer Property](#) [[▶ 1731](#)]
-  [DataType.IsPointer Property](#) [[▶ 1732](#)]
-  [DataType.IsPrimitive Property](#) [[▶ 1732](#)]
-  [DataType.IsReference Property](#) [[▶ 1733](#)]

6.7.26.1 UnionType Properties

The [UnionType](#) [[▶ 1933](#)] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	Fields [▶ 1936]	Gets a read only collection of the Fields [▶ 2535] of the IUnionType [▶ 2740].
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the DataType (Inherited from DataType [▶ 1721].)





	Name	Description
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DataType [▶ 1721].)
	ManagedType [▶ 1733]	Gets the corresponding .NET Type if attached. (Inherited from DataType [▶ 1721].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Reference

[UnionType Class](#) [[▶ 1933](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

Also see about this

-  [DataType.IsContainer Property](#) [[▶ 1731](#)]
-  [DataType.IsPointer Property](#) [[▶ 1732](#)]
-  [DataType.IsPrimitive Property](#) [[▶ 1732](#)]
-  [DataType.IsReference Property](#) [[▶ 1733](#)]

6.7.26.1.1 UnionType.Fields Property

Gets a read only collection of the [Fields](#) [[▶ 2535](#)] of the [IUnionType](#) [[▶ 2740](#)].

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IFieldCollection Fields { get; }
```

Property Value

Type: [IFieldCollection](#) [[▶ 2538](#)]

The members as read only collection.

Implements

[IUnionType.Fields](#) [[▶ 2743](#)]

Reference





[UnionType Class](#) [[▶ 1933](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.26.2 UnionType Methods

The [UnionType](#) [[▶ 1933](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721].)

Extension Methods

	Name	Description
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded.

	Name	Description
		Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[UnionType Class](#) [▶ 1933]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.27 WStringType Class

Represents an Unicode string (Wide string)

Inheritance Hierarchy

System.Object

[TwinCAT.Ads.TypeSystem.DataType](#) [▶ 1721]

[TwinCAT.Ads.TypeSystem.WStringType](#)

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229










Syntax









C#

```
public sealed class WStringType : DataType,
    IStringType, IDataType, IBitSize
```





The WStringType type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	Encoding [▶ 1941]	Gets the encoding of the String (Encoding.Default (STRING) or Encoding.UNICODE (WSTRING))
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the DataType (Inherited from DataType [▶ 1721].)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)

	Name	Description
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DataType [▶ 1721].)
	IsFixedLength [▶ 1942]	Gets a value indicating whether the string is of fixed length.
	Length [▶ 1942]	Gets the number of characters within the string.
	ManagedType [▶ 1733]	Gets the corresponding .NET Type if attached. (Inherited from DataType [▶ 1721].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString [▶ 1944]	Returns a String that represents this instance. (Overrides DataType.ToString . [▶ 1738].)

Extension Methods





	Name	Description
	IsArrayOfPrimitives . [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives (Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer . [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer (Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive . [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive (Boolean) [▶ 3034]	Overloaded.

	Name	Description
		Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]







Also see about this












-  [DataType.IsContainer Property](#) [[▶ 1731](#)]
-  [DataType.IsPointer Property](#) [[▶ 1732](#)]
-  [DataType.IsPrimitive Property](#) [[▶ 1732](#)]
-  [DataType.IsReference Property](#) [[▶ 1733](#)]

6.7.27.1 WStringType Properties

The [WStringType](#) [[▶ 1938](#)] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
	BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
	ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
	Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
	Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
	Encoding [▶ 1941]	Gets the encoding of the String (Encoding.Default (STRING) or Encoding.UNICODE (WSTRING))





	Name	Description
	FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
	Id [▶ 1729]	Gets the ID of the DataType (Inherited from DataType [▶ 1721].)
	IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
	IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DataType [▶ 1721].)
	IsFixedLength [▶ 1942]	Gets a value indicating whether the string is of fixed length.
	Length [▶ 1942]	Gets the number of characters within the string.
	ManagedType [▶ 1733]	Gets the corresponding .NET Type if attached. (Inherited from DataType [▶ 1721].)
	Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
	Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
	Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
	TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Reference

[WStringType Class](#) [[▶ 1938](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

Also see about this

-  [DataType.IsContainer Property](#) [[▶ 1731](#)]
-  [DataType.IsPointer Property](#) [[▶ 1732](#)]
-  [DataType.IsPrimitive Property](#) [[▶ 1732](#)]
-  [DataType.IsReference Property](#) [[▶ 1733](#)]

6.7.27.1.1 WStringType.Encoding Property

Gets the encoding of the String (Encoding.Default (STRING) or Encoding.UNICODE (WSTRING))

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Encoding Encoding { get; }
```

Property Value

Type: Encoding
The encoding.

Implements[IStringType.Encoding](#) [▶ 2664]**Reference**[WStringType Class](#) [▶ 1938][TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]**6.7.27.1.2 WStringType.IsFixedLength Property**

Gets a value indicating whether the string is of fixed length.

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public bool IsFixedLength { get; }
```

Property Value

Type: Boolean

true if this instance is fixed length; otherwise, false.

Implements[IStringType.IsFixedLength](#) [▶ 2665]**Exceptions**

Exception	Condition
NotImplementedException	

Reference[WStringType Class](#) [▶ 1938][TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]**6.7.27.1.3 WStringType.Length Property**

Gets the number of characters within the string.

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public int Length { get; }
```

Property Value

Type: Int32

The length.

Implements

[IStringType.Length](#) [[▸ 2665](#)]

Reference





[WStringType Class](#) [[▸ 1938](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▸ 1698](#)]

6.7.27.2 WStringType Methods

The [WStringType](#) [[▸ 1938](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString [▸ 1944]	Returns a String that represents this instance. (Overrides DataType.ToString . [▸ 1738].)

Extension Methods

	Name	Description
	IsArrayOfPrimitives . [▸ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▸ 3021].)
	IsArrayOfPrimitives(Boolean) [▸ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▸ 3021].)
	IsContainer . [▸ 3027]	Overloaded. Gets a value indicating whether this IDataType [▸ 2475] is a container type (Defined by DataTypeExtension [▸ 3021].)
	IsContainer(Boolean) [▸ 3028]	Overloaded. Gets a value indicating whether this IDataType [▸ 2475] is a container type (Defined by DataTypeExtension [▸ 3021].)
	IsPointer [▸ 3031]	Gets a value indicating whether this IDataType [▸ 2475] is a pointer type (Defined by DataTypeExtension [▸ 3021].)
	IsPrimitive . [▸ 3033]	Overloaded. Gets a value indicating whether this IDataType [▸ 2475] is primitive (Defined by DataTypeExtension [▸ 3021].)
	IsPrimitive(Boolean) [▸ 3034]	Overloaded. Gets a value indicating whether this IDataType [▸ 2475] is primitive (Defined by DataTypeExtension [▸ 3021].)
	IsReference [▸ 3036]	Gets a value indicating whether this IDataType [▸ 2475] is a reference type (Defined by DataTypeExtension [▸ 3021].)

	Name	Description
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[WStringType Class](#) [[▶ 1938](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.27.2.1 WStringType.ToString Method

Returns a String that represents this instance.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```

Return Value

Type: String

A String that represents this instance.

Reference

[WStringType Class](#) [[▶ 1938](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.28 AlignmentCalculator Class

Class calculating alignments.

Inheritance Hierarchy

System.Object

 TwinCAT.Ads.TypeSystem.AlignmentCalculator

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public class AlignmentCalculator
```

The AlignmentCalculator type exposes the following members.

Constructors

	Name	Description
	AlignmentCalculator [▶ 1945]	Initializes a new instance of the AlignmentCalculator class.

Properties

	Name	Description
	Pack1 [▶ 1946]	Gets the Pack1 Alignment Calculator

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetNextOffset [▶ 1947]	Gets the next offset.
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [\[▶ 1698\]](#)

6.7.28.1 AlignmentCalculator Constructor

Initializes a new instance of the [AlignmentCalculator](#) [\[▶ 1944\]](#) class.

Namespace: [TwinCAT.Ads.TypeSystem](#) [\[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AlignmentCalculator(
    int packMode
)
```

Parameters

packMode	Type: System.Int32 The pack mode.
----------	--------------------------------------

Reference[AlignmentCalculator Class \[▸ 1944\]](#)[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)**6.7.28.2 AlignmentCalculator Properties**

The [AlignmentCalculator \[▸ 1944\]](#) type exposes the following members.

Properties

	Name	Description
	Pack1 [▸ 1946]	Gets the Pack1 Alignment Calculator

Reference[AlignmentCalculator Class \[▸ 1944\]](#)[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)**6.7.28.2.1 AlignmentCalculator.Pack1 Property**

Gets the Pack1 Alignment Calculator

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static AlignmentCalculator Pack1 { get; }
```

Property ValueType: [AlignmentCalculator \[▸ 1944\]](#)

The pack1.

Reference[AlignmentCalculator Class \[▸ 1944\]](#)[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)**6.7.28.3 AlignmentCalculator Methods**

The [AlignmentCalculator \[▸ 1944\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)

	Name	Description
	GetNextOffset [▶ 1947]	Gets the next offset.
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[AlignmentCalculator Class](#) [▶ 1944]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.28.3.1 AlignmentCalculator.GetNextOffset Method

Gets the next offset.

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int GetNextOffset(
    IInstanceCollection<IMember> instances
)
```

Parameters

instances	Type: TwinCAT.TypeSystem.IInstanceCollection [▶ 2554]. IMember [▶ 2561]. The instances.
-----------	--

Return Value

Type: Int32
System.Int32.

Reference

[AlignmentCalculator Class](#) [▶ 1944]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.29 FluentRpcMethodExtension Class

Class [FluentRpcMethodExtension](#).

Inheritance Hierarchy

System.Object
TwinCAT.Ads.TypeSystem.FlulentRpcMethodExtension

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static class FluentRpcMethodExtension
```

The `FluentRpcMethodExtension` type exposes the following members.

Methods

	Name	Description
	AddParameter(RpcMethod, IRpcMethodParameter) [► 1949]	Adds a parameter to the <code>RpcMethod</code>
	AddParameter(RpcMethod, String, IDataType, MethodParamFlags) [► 1950]	Adds a parameter to the <code>RpcMethod</code>
	AddParameter(RpcMethod, String, IDataType, MethodParamFlags, Int32) [► 1951]	Adds a parameter to the <code>RpcMethod</code>
	SetReturnType [► 1952]	Sets the <code>ReturnType</code> of the <code>RpcMethod</code> .

Remarks

This method extends the [RpcMethod](#) [► 1813] with extension methods to construct methods fluently.

Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.29.1 FluentRpcMethodExtension Methods

The `FluentRpcMethodExtension` [► 1947] type exposes the following members.

Methods

	Name	Description
	AddParameter(RpcMethod, IRpcMethodParameter) [► 1949]	Adds a parameter to the <code>RpcMethod</code>
	AddParameter(RpcMethod, String, IDataType, MethodParamFlags) [► 1950]	Adds a parameter to the <code>RpcMethod</code>
	AddParameter(RpcMethod, String,	Adds a parameter to the <code>RpcMethod</code>

	Name	Description
	IDataType , MethodParamFlags , Int32 [▶ 1951]	
	SetReturnType [▶ 1952]	Sets the ReturnType of the RpcMethod.

Reference

[FluentRpcMethodExtension Class](#) [[▶ 1947](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.29.1.1 FluentRpcMethodExtension.AddParameter Method

Overload List

	Name	Description
	AddParameter(RpcMethod, IRpcMethodParameter) [▶ 1949]	Adds a parameter to the RpcMethod
	AddParameter(RpcMethod, String, IDataType, MethodParamFlags) [▶ 1950]	Adds a parameter to the RpcMethod
	AddParameter(RpcMethod, String, IDataType, MethodParamFlags, Int32) [▶ 1951]	Adds a parameter to the RpcMethod

Reference

[FluentRpcMethodExtension Class](#) [[▶ 1947](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.29.1.1.1 FluentRpcMethodExtension.AddParameter Method (RpcMethod, IRpcMethodParameter)

Adds a parameter to the RpcMethod

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static RpcMethod AddParameter(
    this RpcMethod method,
    IRpcMethodParameter parameter
)
```

Parameters

method	Type: TwinCAT.Ads.TypeSystem.RpcMethod [1813] The method.
parameter	Type: TwinCAT.TypeSystem.IRpcMethodParameter [2635] The Parameter.

Return Value

Type: [RpcMethod](#) [[1813](#)]
RpcMethod.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [RpcMethod](#) [[1813](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[FluentRpcMethodExtension Class](#) [[1947](#)]

[AddParameter Overload](#) [[1949](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[1698](#)]

6.7.29.1.1.2 FluentRpcMethodExtension.AddParameter Method (RpcMethod, String, IData Type, MethodParamFlags)

Adds a parameter to the RpcMethod

Namespace: [TwinCAT.Ads.TypeSystem](#) [[1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static RpcMethod AddParameter(
    this RpcMethod method,
    string name,
    IData Type type,
    MethodParamFlags flags
)
```

Parameters

method	Type: TwinCAT.Ads.TypeSystem.RpcMethod [1813] The method.
name	Type: System.String The name.
type	Type: TwinCAT.TypeSystem.IData Type [2475] The type.

flags	Type: TwinCAT.TypeSystem.MethodParamFlags [▶ 2812] The flags.
-------	--

Return Value

Type: [RpcMethod](#) [[▶ 1813](#)]
RpcMethod.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [RpcMethod](#) [[▶ 1813](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[FluentRpcMethodExtension Class](#) [[▶ 1947](#)]

[AddParameter Overload](#) [[▶ 1949](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.29.1.1.3 FluentRpcMethodExtension.AddParameter Method (RpcMethod, String, IDataTypes, MethodParamFlags, Int32)

Adds a parameter to the RpcMethod

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static RpcMethod AddParameter(
    this RpcMethod method,
    string name,
    IDataTypes type,
    MethodParamFlags flags,
    int lengthIsIndex
)
```

Parameters

method	Type: TwinCAT.Ads.TypeSystem.RpcMethod [▶ 1813] The method.
name	Type: System.String The parameter name.
type	Type: TwinCAT.TypeSystem.IDataTypes [▶ 2475] The type.
flags	Type: TwinCAT.TypeSystem.MethodParamFlags [▶ 2812] Flags of the method parameter.
lengthIsIndex	Type: System.Int32 LengthIs attribute.

Return Value

Type: [RpcMethod](#) [[▶ 1813](#)]
RpcMethod.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [RpcMethod](#) [[1813](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[FluentRpcMethodExtension Class](#) [[1947](#)]

[AddParameter Overload](#) [[1949](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[1698](#)]

6.7.29.1.2 FluentRpcMethodExtension.SetReturnType Method

Sets the ReturnType of the RpcMethod.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static RpcMethod SetReturnType(
    this RpcMethod method,
    IDatatype returnType
)
```

Parameters

method	Type: TwinCAT.Ads.TypeSystem.RpcMethod [1813] The method.
returnType	Type: TwinCAT.TypeSystem.IDatatype [2475] The datatype of the methods return value.

Return Value

Type: [RpcMethod](#) [[1813](#)]
RpcMethod.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [RpcMethod](#) [[1813](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[FluentRpcMethodExtension Class](#) [[1947](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[1698](#)]

6.7.30 FluentRpcStructTypeExtension Class

Class [FluentRpcStructTypeExtension](#).

Inheritance Hierarchy

System.Object

TwinCAT.Ads.TypeSystem.FluentRpcStructTypeExtension

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5





Syntax

C#

```
public static class FluentRpcStructTypeExtension
```

The `FluentRpcStructTypeExtension` type exposes the following members.

Methods

	Name	Description
 	AddAligned [▸ 1953]	Adds a member
 	AddMethod [▸ 1954]	Adds a RpcMethod.

Remarks

Fluent extension class for adding Members and Methods to [RpcStructType \[▸ 1835\]](#)s.





Reference

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.30.1 FluentRpcStructTypeExtension Methods

The `FluentRpcStructTypeExtension [▸ 1952]` type exposes the following members.

Methods

	Name	Description
 	AddAligned [▸ 1953]	Adds a member
 	AddMethod [▸ 1954]	Adds a RpcMethod.

Reference

[FluentRpcStructTypeExtension Class \[▸ 1952\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.30.1.1 FluentRpcStructTypeExtension.AddAligned Method

Adds a member

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public static RpcStructType AddAligned(  
    this RpcStructType str,  
    IMember member  
)
```

Parameters

str Type: [TwinCAT.Ads.TypeSystem.RpcStructType \[▸ 1835\]](#)
The struct.

member Type: [TwinCAT.TypeSystem.IMember \[▸ 2561\]](#)
The member.

Return Value

Type: [RpcStructType \[▸ 1835\]](#)
RpcStructType.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [RpcStructType \[▸ 1835\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[FluentRpcStructTypeExtension Class \[▸ 1952\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.30.1.2 **FluentRpcStructTypeExtension.AddMethod Method**

Adds a RpcMethod.

Namespace: [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public static RpcStructType AddMethod(  
    this RpcStructType str,  
    IRpcMethod method  
)
```

Parameters

str Type: [TwinCAT.Ads.TypeSystem.RpcStructType \[▸ 1835\]](#)
The struct.

method Type: [TwinCAT.TypeSystem.IRpcMethod \[▸ 2625\]](#)
The method.

Return Value

Type: [RpcStructType \[▸ 1835\]](#)
RpcStructType.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [RpcStructType](#) [[▶ 1835](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[FluentRpcStructTypeExtension Class](#) [[▶ 1952](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.31 FluentStructTypeExtension Class

Class [FluentStructTypeExtension](#).

Inheritance Hierarchy

System.Object
 TwinCAT.Ads.TypeSystem.FluentStructTypeExtension

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static class FluentStructTypeExtension
```

The [FluentStructTypeExtension](#) type exposes the following members.

Methods

	Name	Description
	AddAligned [▶ 1956]	Adds a member to the StructType [▶ 1846]
	AddMethod [▶ 1956]	Adds a RpcMethod.

Remarks

Fluent interface for adding members to [StructType](#) [[▶ 1846](#)]s.

Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.31.1 FluentStructTypeExtension Methods

The [FluentStructTypeExtension](#) [[▶ 1955](#)] type exposes the following members.

Methods

	Name	Description
	AddAligned [▶ 1956]	Adds a member to the StructType [▶ 1846]
	AddMethod [▶ 1956]	Adds a RpcMethod.

Reference[FluentStructTypeExtension Class \[► 1955\]](#)[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)**6.7.31.1.1 FluentStructTypeExtension.AddAligned Method**Adds a member to the [StructType \[► 1846\]](#)**Namespace:** [TwinCAT.Ads.TypeSystem \[► 1698\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static StructType AddAligned(
    this StructType str,
    IMember member
)
```

Parameters

str	Type: TwinCAT.Ads.TypeSystem.StructType [► 1846] The string.
member	Type: TwinCAT.TypeSystem.IMember [► 2561] The member.

Return ValueType: [StructType \[► 1846\]](#)

StructType.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [StructType \[► 1846\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference[FluentStructTypeExtension Class \[► 1955\]](#)[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)**6.7.31.1.2 FluentStructTypeExtension.AddMethod Method**

Adds a RpcMethod.

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static StructType AddMethod(
    this StructType str,
    IRpcMethod method
)
```

Parameters

str	Type: TwinCAT.Ads.TypeSystem.StructType [▸ 1846] The struct.
method	Type: TwinCAT.TypeSystem.IRpcMethod [▸ 2625] The method.

Return Value

Type: [StructType \[▸ 1846\]](#)
RpcStructType.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [StructType \[▸ 1846\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

- [FluentStructTypeExtension Class \[▸ 1955\]](#)
- [TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.7.32 InterfaceType Class

Represents an interface type

Inheritance Hierarchy

- System.Object
 - [TwinCAT.Ads.TypeSystem.DataType \[▸ 1721\]](#)
 - [TwinCAT.Ads.TypeSystem.InterfaceType](#)
- Namespace:** [TwinCAT.Ads.TypeSystem \[▸ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public class InterfaceType : DataType,
    IInterfaceType, IDataType, IBitSize, IRpcCallableType
```

The InterfaceType type exposes the following members.

Properties

	Name	Description
	AllMembers [▸ 1962]	Gets all members (down the derivation hierarchy)
	Attributes [▸ 1727]	Gets the attributes of the IDataType [▸ 2475] (Inherited from DataType [▸ 1721] .)
	BaseType [▸ 1963]	Gets the structs Base Type (Null if not derived).
	BaseTypeName [▸ 1963]	Gets the the Name of the Base class (if derived)
	BitSize [▸ 1727]	Gets the size of the DataType [▸ 1721] in bits. (Inherited from DataType [▸ 1721] .)
	ByteSize [▸ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▸ 1721] .)

Name	Description
Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
HasRpcMethods [▶ 1961]	Gets a value indicating whether this instance has RPC Methods.
Id [▶ 1729]	Gets the ID of the DataType (Inherited from DataType [▶ 1721].)
InterfaceImplementationNames [▶ 1961]	Gets the names of the interfaces, this IDataType [▶ 2475] implements.
InterfaceImplementations [▶ 1962]	Gets the resolved interface types, this IDataType [▶ 2475] implments.
IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DataType [▶ 1721].)
IsDerived [▶ 1964]	Gets a value indicating whether this instance is derived.
ManagedType [▶ 1733]	Gets the corresponding .NET Type if attached. (Inherited from DataType [▶ 1721].)
Members [▶ 1964]	Gets a read only collection of the Members [▶ 2561] of the IInterfaceType [▶ 2983].
Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
RpcMethods [▶ 1965]	Gets the Method descriptions for the IRpcCallableType [▶ 2624]
Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Methods

Name	Description
Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
GetHashCode	Serves as the default hash function. (Inherited from Object .)
GetType	Gets the Type of the current instance. (Inherited from Object .)
MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object .)
ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721].)

Extension Methods

	Name	Description
	IsArrayOfPrimitives . [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer . [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive . [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive . [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve . [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

Also see about this

- [DataType.IsContainer Property](#) [▶ 1731]
- [DataType.IsPointer Property](#) [▶ 1732]
- [DataType.IsPrimitive Property](#) [▶ 1732]
- [DataType.IsReference Property](#) [▶ 1733]

6.7.32.1 InterfaceType Properties

The [InterfaceType](#) [[▶ 1957](#)] type exposes the following members.

Properties

Name	Description
AllMembers [▶ 1962]	Gets all members (down the derivation hierarchy)
Attributes [▶ 1727]	Gets the attributes of the IDataType [▶ 2475] (Inherited from DataType [▶ 1721].)
BaseType [▶ 1963]	Gets the structs Base Type (Null if not derived).
BaseTypeName [▶ 1963]	Gets the the Name of the Base class (if derived)
BitSize [▶ 1727]	Gets the size of the DataType [▶ 1721] in bits. (Inherited from DataType [▶ 1721].)
ByteSize [▶ 1728]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DataType [▶ 1721].)
Category [▶ 1728]	Gets the Data Type category (Inherited from DataType [▶ 1721].)
Comment [▶ 1729]	Gets the comment. (Inherited from DataType [▶ 1721].)
FullName [▶ 1729]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from DataType [▶ 1721].)
HasRpcMethods [▶ 1961]	Gets a value indicating whether this instance has RPC Methods.
Id [▶ 1729]	Gets the ID of the DataType (Inherited from DataType [▶ 1721].)
InterfaceImplementationNames [▶ 1961]	Gets the names of the interfaces, this IDataType [▶ 2475] implements.
InterfaceImplementations [▶ 1962]	Gets the resolved interface types, this IDataType [▶ 2475] implments.
IsBitType [▶ 1730]	Gets a value indicating whether this IDataType [▶ 2475] is a bit mapping Type (Inherited from DataType [▶ 1721].)
IsByteAligned [▶ 1730]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DataType [▶ 1721].)
IsDerived [▶ 1964]	Gets a value indicating whether this instance is derived.
ManagedType [▶ 1733]	Gets the corresponding .NET Type if attached. (Inherited from DataType [▶ 1721].)
Members [▶ 1964]	Gets a read only collection of the Members [▶ 2561] of the IInterfaceType [▶ 2983].
Name [▶ 1734]	Gets the name of the Data Type (without namespace) (Inherited from DataType [▶ 1721].)
Namespace [▶ 1734]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from DataType [▶ 1721].)
RpcMethods [▶ 1965]	Gets the Method descriptions for the IRpcCallableType [▶ 2624]
Size [▶ 1735]	Gets the Size of the DataType [▶ 1721] in Bytes or bits. (Inherited from DataType [▶ 1721].)
TypeGuid [▶ 1735]	Gets the Guid of the DataType [▶ 1721] (optional) (Inherited from DataType [▶ 1721].)

Reference

[InterfaceType Class \[► 1957\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

Also see about this

- [DataType.IsContainer Property \[► 1731\]](#)
- [DataType.IsPointer Property \[► 1732\]](#)
- [DataType.IsPrimitive Property \[► 1732\]](#)
- [DataType.IsReference Property \[► 1733\]](#)

6.7.32.1.1 InterfaceType.HasRpcMethods Property

Gets a value indicating whether this instance has RPC Methods.

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool HasRpcMethods { get; }
```

Property Value

Type: Boolean

true if this instance has methods; otherwise, false.

Implements

[IInterfaceType.HasRpcMethods \[► 2986\]](#)

Remarks

The `DataType` (Structure) must be marked with the `PlcAttribute` 'TcRpcEnable' to enable `RpcMethods`, otherwise `RpcMethods` are not passed through to the ADS symbolic information.

Reference

[InterfaceType Class \[► 1957\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.32.1.2 InterfaceType.InterfaceImplementationNames Property

Gets the names of the interfaces, this `IDataType` [► 2475] implements.

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string[] InterfaceImplementationNames { get; }
```

Property Value

Type: [.String](#).
The interface implementations.

Implements

[IInterfaceType.InterfaceImplementationNames](#) [[▶ 2987](#)]

Reference

[InterfaceType Class](#) [[▶ 1957](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.32.1.3 InterfaceType.InterfaceImplementations Property

Gets the resolved interface types, this [IDataType](#) [[▶ 2475](#)] implements.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IInterfaceType?[]? InterfaceImplementations { get; }
```

Property Value

Type: [.IInterfaceType](#) [[▶ 2983](#)].
The implements.

Implements

[IInterfaceType.InterfaceImplementations](#) [[▶ 2987](#)]

Reference

[InterfaceType Class](#) [[▶ 1957](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.32.1.4 InterfaceType.AllMembers Property

Gets all members (down the derivation hierarchy)

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IMemberCollection AllMembers { get; }
```

Property Value

Type: [IMemberCollection](#) [[▶ 2565](#)]
All members.

Implements

[IInterfaceType.AllMembers](#) [[▶ 2987](#)]

Reference

[InterfaceType Class](#) [[▶ 1957](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.32.1.5 InterfaceType.BaseType Property

Gets the structs Base Type (Null if not derived).

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataType? BaseType { get; }
```

Property Value

Type: [IDataType](#) [[▶ 2475](#)]

Implements

[IInterfaceType.BaseType](#) [[▶ 2988](#)]

Reference

[InterfaceType Class](#) [[▶ 1957](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.32.1.6 InterfaceType.BaseTypeName Property

Gets the the Name of the Base class (if derived)

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string BaseTypeName { get; }
```

Property Value

Type: String
Empty if not derived.

Implements

[IInterfaceType.BaseTypeName](#) [[▶ 2988](#)]

Reference

[InterfaceType Class](#) [[▶ 1957](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.32.1.7 InterfaceType.IsDerived Property

Gets a value indicating whether this instance is derived.

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsDerived { get; }
```

Property Value

Type: Boolean

true if this instance is derived; otherwise, false.

Reference

[InterfaceType Class](#) [[▶ 1957](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.32.1.8 InterfaceType.Members Property

Gets a read only collection of the [Members](#) [[▶ 2561](#)] of the [IInterfaceType](#) [[▶ 2983](#)].

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IMemberCollection Members { get; }
```

Property Value

Type: [IMemberCollection](#) [[▶ 2565](#)]

The members as read only collection.

Implements

[IInterfaceType.Members](#) [[▶ 2988](#)]

Remarks

If the [IStructType](#) [[▶ 2671](#)] is derived, only the extended members are returned. To get all supported members down the inheritance chain, use the [AllMembers](#) [[▶ 2987](#)] property.

Reference

[InterfaceType Class](#) [[▶ 1957](#)]

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.32.1.9 InterfaceType.RpcMethods Property

Gets the Method descriptions for the [IRpcCallableType \[▶ 2624\]](#)

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IRpcMethodCollection RpcMethods { get; }
```

Property Value

Type: [IRpcMethodCollection \[▶ 2629\]](#)

The methods.

Implements

[IRpcCallableType.RpcMethods \[▶ 2624\]](#)

Remarks

The DataType (Structure) must be marked with the PlcAttribute 'TcRpcEnable' to enable RpcMethods, otherwise RpcMethods are not passed through to the ADS symbolic information.

Reference

[InterfaceType Class \[▶ 1957\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.32.2 InterfaceType Methods

The [InterfaceType \[▶ 1957\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString [▶ 1738]	Returns a String that represents this instance. (Inherited from DataType [▶ 1721] .)

Extension Methods

	Name	Description
	IsArrayOfPrimitives. [▶ 3025]	Overloaded.

Name	Description
	Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsContainer. [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive. [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
Resolve. [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[InterfaceType Class](#) [▶ 1957]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.33 RpcStructInstance Class

Struct Instance with RPC Methods

Inheritance Hierarchy

System.Object

 TwinCAT.Ads.TypeSystem.Instance [▶ 1765]

 TwinCAT.Ads.TypeSystem.Symbol [▶ 1863]

 StructInstance

 TwinCAT.Ads.TypeSystem.RpcStructInstance

Namespace: TwinCAT.Ads.TypeSystem [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

















Syntax

C#

```
public sealed class RpcStructInstance : StructInstance,
    IRpcStructInstance, IStructInstance, ISymbol, IAttributedInstance, IInstance,
    IBitSize, IRpcCallableInstance
```


The RpcStructInstance type exposes the following members.





















Properties




















	Name	Description
	AccessRights [▶ 1872]	Gets the access rights. (Inherited from Symbol [▶ 1863].)
	Attributes [▶ 1768]	Gets the Type Attributes. (Inherited from Instance [▶ 1765].)
	BitSize [▶ 1768]	Gets the size of this Instance [▶ 1765] in bits. (Inherited from Instance [▶ 1765].)
	ByteSize [▶ 1769]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from Instance [▶ 1765].)
	Category [▶ 1769]	Gets the the DataTypeCategory [▶ 2111] of the Instance. (Inherited from Instance [▶ 1765].)
	Comment [▶ 1770]	Gets the comment. (Inherited from Instance [▶ 1765].)
	Connection [▶ 1873]	Gets the connection that produces values for this ValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	ContextMask [▶ 1770]	Gets the context mask of this instance. (Inherited from Instance [▶ 1765].)
	DataType [▶ 1771]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549]. (Inherited from Instance [▶ 1765].)
	HasValue [▶ 1771]	Gets a value indicating whether this instance has a value. (Inherited from Instance [▶ 1765].)
	ImageBaseAddress [▶ 1873]	Gets the AmsAddress [▶ 752] of the Process Image (Inherited from Symbol [▶ 1863].)
	IndexGroup [▶ 1874]	Gets the index group of the Symbol (Inherited from Symbol [▶ 1863].)
	IndexOffset [▶ 1874]	Gets the index offset of the Symbol (Inherited from Symbol [▶ 1863].)
	InstanceName [▶ 1771]	Gets the name of the instance (without periods (.)) (Inherited from Instance [▶ 1765].)
	InstancePath [▶ 1874]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from Symbol [▶ 1863].)
	IsBitType [▶ 1773]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from Instance [▶ 1765].)



	Name	Description
	IsByteAligned [▶ 1773]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from Instance [▶ 1765].)
	IsDereferencedPointer [▶ 1876]	Gets or sets a value indicating whether an ancestor is a dereferenced Pointer (Inherited from Symbol [▶ 1863].)
	IsDereferencedReference [▶ 1876]	Gets or sets a value indicating whether an ancestor is a dereferenced Reference (Inherited from Symbol [▶ 1863].)
	IsPersistent [▶ 1773]	Indicates that this instance is persistent. (Inherited from Instance [▶ 1765].)
	IsPointer [▶ 1774]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
	IsReadOnly [▶ 1774]	Indicates that this instance is read only. (Inherited from Instance [▶ 1765].)
	IsRecursive [▶ 1877]	Gets a value indicating whether this instance is recursive. (Inherited from Symbol [▶ 1863].)
	IsReference [▶ 1775]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
	IsStatic [▶ 1775]	Gets a value indicating whether this Instance [▶ 2549] is static. (Inherited from Instance [▶ 1765].)
	IsTcComInterfacePointer [▶ 1776]	Indicates that this instance is a TcComInterfacePointer. (Inherited from Instance [▶ 1765].)
	IsTypeGuid [▶ 1776]	Indicates that this instance has set TypeGuid flag. (Inherited from Instance [▶ 1765].)
	IsVirtual [▶ 1878]	Gets a value indicating whether this instance is virtual. (Inherited from Symbol [▶ 1863].)
	Namespace [▶ 1776]	Gets the namespace name. (Inherited from Instance [▶ 1765].)
	NotificationSettings [▶ 1878]	Gets or sets the notification settings. (Inherited from Symbol [▶ 1863].)
	Parent [▶ 1879]	Gets the parent Symbol (Inherited from Symbol [▶ 1863].)
	RpcMethods [▶ 1973]	Gets the Method descriptions for the IRpcCallableType [▶ 2624]
	Size [▶ 1777]	Gets the size of the IDataType [▶ 2475] in bytes or Bits dependant on IsBitType [▶ 1773] (Inherited from Instance [▶ 1765].)
	SubSymbolCount [▶ 1879]	Gets the number of SubSymbols (Inherited from Symbol [▶ 1863].)
	SubSymbols [▶ 1880]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from Symbol [▶ 1863].)
	TypeName [▶ 1777]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from Instance [▶ 1765].)
	ValueEncoding [▶ 1880]	Gets the value encoding. (Inherited from Symbol [▶ 1863].)

Methods



	Name	Description
	Equals [▶ 1885]	Equals (Inherited from Symbol [▶ 1863].)

	Name	Description
	GetHashCode [▶ 1887]	Gets the HashCode of the Address (Inherited from Symbol [▶ 1863].)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
 	InvokeRpcMethod(String, .Object.) [▶ 1976]	Invokes the specified RPC Method
 	InvokeRpcMethod(String, .Object., .Object.) [▶ 1978]	Invokes the specified RPC Method
 	InvokeRpcMethod(String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier., Object.) [▶ 1979]	Invokes the specified RPC Method
 	InvokeRpcMethodAsync(String, .Object., CancellationTokentoken) [▶ 1981]	Invokes the specified RPC Method asynchronously
 	InvokeRpcMethodAsync(String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, CancellationTokentoken) [▶ 1983]	Invokes the specified RPC Method asynchronously
	ReadAnyValue(Type) [▶ 1894]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from Symbol [▶ 1863].)
	ReadAnyValue(Type, Int32) [▶ 1895]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from Symbol [▶ 1863].)
	ReadAnyValueAsync [▶ 1897]	Reads the (AnyType) value asynchronously. (Inherited from Symbol [▶ 1863].)
	ReadRawValue. [▶ 1899]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from Symbol [▶ 1863].)
	ReadRawValue(Int32) [▶ 1900]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from Symbol [▶ 1863].)
	ReadRawValueAsync [▶ 1901]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) asynchronously. (Inherited from Symbol [▶ 1863].)
	ReadValue. [▶ 1902]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	ReadValue(Int32) [▶ 1902]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)

	Name	Description
	ReadValueAsync [▶ 1903]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from Symbol [▶ 1863].)
	SetParent [▶ 1904]	Sets the parent symbol. (Inherited from Symbol [▶ 1863].)
	ToString [▶ 1904]	Returns a String that represents this instance. (Inherited from Symbol [▶ 1863].)
 	TryInvokeRpcMethod(String, .Object., Object.) [▶ 1985]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(String, .Object., .Object., Object.) [▶ 1987]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(IRpcMethod, .Object., .AnyTypeSpecifier, .AnyTypeSpecifier, .Object., Object.) [▶ 1989]	Invokes the the specified RpcMethod of the IRpcCallableInstance [▶ 2606].
 	TryInvokeRpcMethod(String, .Object., .AnyTypeSpecifier, .AnyTypeSpecifier, .Object., Object.) [▶ 1990]	Invokes the the specified RpcMethod of the IRpcCallableInstance [▶ 2606].
	TryReadValue [▶ 1905]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	TryWriteValue [▶ 1905]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	UpdateAnyValue(Object.) [▶ 1907]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from Symbol [▶ 1863].)
	UpdateAnyValue(Object, Int32) [▶ 1907]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from Symbol [▶ 1863].)
	WriteRawValue(Byte.) [▶ 1908]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from Symbol [▶ 1863].)
	WriteRawValue(Byte, Int32) [▶ 1909]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from Symbol [▶ 1863].)
	WriteRawValueAsync [▶ 1910]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from Symbol [▶ 1863].)
	WriteValue(Object) [▶ 1910]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)

	Name	Description
	WriteValue(Object, Int32) [1911]	Writes the Value of the IValueSymbol [2775] (Inherited from Symbol [1863].)
	WriteValueAsync [1912]	Writes the Value of the IValueSymbol [2775] (Inherited from Symbol [1863].)

Events

	Name	Description
	RawValueChanged [1913]	Occurs when the RawValue of the IValueSymbol [2775] has changed. (Inherited from Symbol [1863].)
	ValueChanged [1913]	Occurs when the (Primitive) value of the IValueSymbol [2775] has changed. (Inherited from Symbol [1863].)














Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [[1698](#)]


6.7.33.1 RpcStructInstance Properties

The [RpcStructInstance](#) [[1966](#)] type exposes the following members.

Properties

	Name	Description
	AccessRights [1872]	Gets the access rights. (Inherited from Symbol [1863].)
	Attributes [1768]	Gets the Type Attributes. (Inherited from Instance [1765].)
	BitSize [1768]	Gets the size of this Instance [1765] in bits. (Inherited from Instance [1765].)
	ByteSize [1769]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from Instance [1765].)
	Category [1769]	Gets the the DataTypeCategory [2111] of the Instance. (Inherited from Instance [1765].)
	Comment [1770]	Gets the comment. (Inherited from Instance [1765].)
	Connection [1873]	Gets the connection that produces values for this IValueSymbol [2775] (Inherited from Symbol [1863].)
	ContextMask [1770]	Gets the context mask of this instance. (Inherited from Instance [1765].)
	DataType [1771]	Gets the IDataType [2475] of the Instance [2549]. (Inherited from Instance [1765].)
	HasValue [1771]	Gets a value indicating whether this instance has a value. (Inherited from Instance [1765].)
	ImageBaseAddress [1873]	Gets the AmsAddress [752] of the Process Image (Inherited from Symbol [1863].)
	IndexGroup [1874]	Gets the index group of the Symbol (Inherited from Symbol [1863].)
	IndexOffset [1874]	Gets the index offset of the Symbol (Inherited from Symbol [1863].)

	Name	Description
	InstanceName [▶ 1771]	Gets the name of the instance (without periods (.)) (Inherited from Instance [▶ 1765].)
	InstancePath [▶ 1874]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from Symbol [▶ 1863].)
	IsBitType [▶ 1773]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from Instance [▶ 1765].)
	IsByteAligned [▶ 1773]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from Instance [▶ 1765].)
	IsDereferencedPointer [▶ 1876]	Gets or sets a value indicating whether an ancestor is a dereferenced Pointer (Inherited from Symbol [▶ 1863].)
	IsDereferencedReference [▶ 1876]	Gets or sets a value indicating whether an ancestor is a dereferenced Reference (Inherited from Symbol [▶ 1863].)
	IsPersistent [▶ 1773]	Indicates that this instance is persistent. (Inherited from Instance [▶ 1765].)
	IsPointer [▶ 1774]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
	IsReadOnly [▶ 1774]	Indicates that this instance is read only. (Inherited from Instance [▶ 1765].)
	IsRecursive [▶ 1877]	Gets a value indicating whether this instance is recursive. (Inherited from Symbol [▶ 1863].)
	IsReference [▶ 1775]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
	IsStatic [▶ 1775]	Gets a value indicating whether this Instance [▶ 2549] is static. (Inherited from Instance [▶ 1765].)
	IsTcComInterfacePointer [▶ 1776]	Indicates that this instance is a TcComInterfacePointer . (Inherited from Instance [▶ 1765].)
	IsTypeGuid [▶ 1776]	Indicates that this instance has set TypeGuid flag. (Inherited from Instance [▶ 1765].)
	IsVirtual [▶ 1878]	Gets a value indicating whether this instance is virtual. (Inherited from Symbol [▶ 1863].)
	Namespace [▶ 1776]	Gets the namespace name. (Inherited from Instance [▶ 1765].)
	NotificationSettings [▶ 1878]	Gets or sets the notification settings. (Inherited from Symbol [▶ 1863].)
	Parent [▶ 1879]	Gets the parent Symbol (Inherited from Symbol [▶ 1863].)
	RpcMethods [▶ 1973]	Gets the Method descriptions for the IRpcCallableType [▶ 2624]
	Size [▶ 1777]	Gets the size of the IDataType [▶ 2475] in bytes or Bits dependant on IsBitType [▶ 1773] (Inherited from Instance [▶ 1765].)
	SubSymbolCount [▶ 1879]	Gets the number of SubSymbols (Inherited from Symbol [▶ 1863].)
	SubSymbols [▶ 1880]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from Symbol [▶ 1863].)
	TypeName [▶ 1777]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from Instance [▶ 1765].)

	Name	Description
	ValueEncoding [▶ 1880]	Gets the value encoding. (Inherited from Symbol [▶ 1863].)

Reference

[RpcStructInstance Class](#) [\[▶ 1966\]](#)

[TwinCAT.Ads.TypeSystem Namespace](#) [\[▶ 1698\]](#)

6.7.33.1 RpcStructInstance.RpcMethods Property

Gets the Method descriptions for the [IRpcCallableType](#) [\[▶ 2624\]](#)

Namespace: [TwinCAT.Ads.TypeSystem](#) [\[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public IRpcMethodCollection RpcMethods { get; }
```

Property Value

Type: [IRpcMethodCollection](#) [\[▶ 2629\]](#)

The methods.

Implements

[IRpcCallableInstance.RpcMethods](#) [\[▶ 2607\]](#)

Reference






[RpcStructInstance Class](#) [\[▶ 1966\]](#)




















[TwinCAT.Ads.TypeSystem Namespace](#) [\[▶ 1698\]](#)



















6.7.33.2 RpcStructInstance Methods

The [RpcStructInstance](#) [\[▶ 1966\]](#) type exposes the following members.




Methods

	Name	Description
	Equals [▶ 1885]	Equals (Inherited from Symbol [▶ 1863].)
	GetHashCode [▶ 1887]	Gets the HashCode of the Address (Inherited from Symbol [▶ 1863].)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
 	InvokeRpcMethod(S tring, Object.) [▶ 1976]	Invokes the specified RPC Method

	Name	Description
 	InvokeRpcMethod(String, .Object., .Object.) [▶ 1978]	Invokes the specified RPC Method
 	InvokeRpcMethod(String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier., .Object.) [▶ 1979]	Invokes the specified RPC Method
 	InvokeRpcMethodAsync(String, .Object., CancellationToken) [▶ 1981]	Invokes the specified RPC Method asynchronously
 	InvokeRpcMethodAsync(String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, CancellationToken) [▶ 1983]	Invokes the specified RPC Method asynchronously
	ReadAnyValue(Type) [▶ 1894]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from Symbol [▶ 1863].)
	ReadAnyValue(Type, Int32) [▶ 1895]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from Symbol [▶ 1863].)
	ReadAnyValueAsync [▶ 1897]	Reads the (AnyType) value asynchronously. (Inherited from Symbol [▶ 1863].)
	ReadRawValue. [▶ 1899]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from Symbol [▶ 1863].)
	ReadRawValue(Int32) [▶ 1900]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from Symbol [▶ 1863].)
	ReadRawValueAsync [▶ 1901]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) asynchronously. (Inherited from Symbol [▶ 1863].)
	ReadValue. [▶ 1902]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	ReadValue(Int32) [▶ 1902]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	ReadValueAsync [▶ 1903]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from Symbol [▶ 1863].)
	SetParent [▶ 1904]	Sets the parent symbol. (Inherited from Symbol [▶ 1863].)
	ToString [▶ 1904]	Returns a String that represents this instance. (Inherited from Symbol [▶ 1863].)

	Name	Description
 	TryInvokeRpcMethod(String, Object, Object.) [▶ 1985]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(String, Object, Object, Object.) [▶ 1987]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(IRpcMethod, Object, AnyTypeSpecifier, AnyTypeSpecifier, Object, Object.) [▶ 1989]	Invokes the the specified RpcMethod of the IRpcCallableInstance [▶ 2606].
 	TryInvokeRpcMethod(String, Object, AnyTypeSpecifier, AnyTypeSpecifier, Object, Object.) [▶ 1990]	Invokes the the specified RpcMethod of the IRpcCallableInstance [▶ 2606].
	TryReadValue [▶ 1905]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	TryWriteValue [▶ 1905]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	UpdateAnyValue(Object.) [▶ 1907]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from Symbol [▶ 1863].)
	UpdateAnyValue(Object, Int32) [▶ 1907]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from Symbol [▶ 1863].)
	WriteRawValue(Byte.) [▶ 1908]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from Symbol [▶ 1863].)
	WriteRawValue(Byte, Int32) [▶ 1909]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from Symbol [▶ 1863].)
	WriteRawValueAsync [▶ 1910]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from Symbol [▶ 1863].)
	WriteValue(Object) [▶ 1910]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	WriteValue(Object, Int32) [▶ 1911]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	WriteValueAsync [▶ 1912]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)

Reference[RpcStructInstance Class](#) [► 1966][TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]**6.7.33.2.1 RpcStructInstance.InvokeRpcMethod Method****Overload List**

	Name	Description
	InvokeRpcMethod(String, .Object.) [► 1976]	Invokes the specified RPC Method
	InvokeRpcMethod(String, .Object., .Object.) [► 1978]	Invokes the specified RPC Method
	InvokeRpcMethod(String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier., .Object.) [► 1979]	Invokes the specified RPC Method

Reference[RpcStructInstance Class](#) [► 1966][TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]**6.7.33.2.1.1 RpcStructInstance.InvokeRpcMethod Method (String, .Object.)**

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5**Syntax****C#**

```
public Object InvokeRpcMethod(
    string methodName,
    Object[]? inParameters
)
```

Parameters

methodName Type: [System.String](#)
The method name.

inParameters Type: [.System.Object](#).
The input parameters or NULL

Return Value

Type: [Object](#)
The return value of the Method (as object).

Implements

[IRpcCallableInstance.InvokeRpcMethod\(String, .Object.\)](#) [► 2609]

Remarks

This method only supports primitive data types as inParameters. Any available outparameters will be ignored. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```

Reference

[RpcStructInstance Class](#) [► 1966]

[InvokeRpcMethod Overload](#) [▶ 1976]

[TwinCAT.Ads.TypeSystem Namespace](#) [▶ 1698]

6.7.33.2.1.2 RpcStructInstance.InvokeRpcMethod Method (String, .Object., .Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads.TypeSystem](#) [▶ 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public Object InvokeRpcMethod(  
    string methodName,  
    Object[]? inParameters,  
    out Object[]? outParameters  
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The input parameters or NULL
outParameters	Type: .System.Object . The output parameters.

Return Value

Type: [Object](#)
The return value of the Method (as object).

Implements

[IRpcCallableInstance.InvokeRpcMethod\(String, .Object., .Object.\)](#) [▶ 2610]

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [▶ 2094] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram  
{  
    /// <summary>  
    /// Defines the entry point of the application.  
    /// </summary>  
    /// <param name="args">The arguments.</param>  
    static void Main(string[] args)  
    {  
        // Get the AdsAddress from command-line arguments  
        AmsAddress address = ArgParser.Parse(args);  
  
        using (AdsClient client = new AdsClient())  
        {
```

```

//client.Synchronize = false;

// Connect to the target device
client.Connect(address);

SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

// Get the Symbols (Dynamic Symbols)

IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

// Call a Method that has the following signature (within MAIN Program)
/* {attribute 'TcRpcEnable'}
METHOD PUBLIC M_Add : INT
VAR_INPUT
    i1 : INT := 0;
    i2 : INT := 0;
END_VAR
*/

short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

// Call a Method that has no parameter and returns VOID
main.InvokeRpcMethod("M_Method1", new object[] {});

//Browsing RpcMethods
foreach(IRpcMethod method in main.RpcMethods)
{
    string methodName = method.Name;

    foreach(IRpcMethodParameter parameter in method.Parameters)
    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
}
}

```

Reference

[RpcStructInstance Class](#) [► 1966]

[InvokeRpcMethod Overload](#) [► 1976]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.33.2.1.3 **RpcStructInstance.InvokeRpcMethod Method** (String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object..)

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```

public Object InvokeRpcMethod(
    string methodName,
    Object[]? inParameters,
    AnyTypeSpecifier[]? outSpecifiers,
    AnyTypeSpecifier? retSpecifier,
    out Object[]? outParameters
)

```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094]. The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094]. The ret specifier (specifying the return value) or NULL.
outParameters	Type: .System.Object.. The out parameters.

Return Value

Type: [Object](#)
The return value of the Method (as object).

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});
        }
    }
}
```

```

//Browsing RpcMethods
foreach(IRpcMethod method in main.RpcMethods)
{
    string methodName = method.Name;

    foreach(IRpcMethodParameter parameter in method.Parameters)
    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
}

```

Reference



[RpcStructInstance Class \[▶ 1966\]](#)

[InvokeRpcMethod Overload \[▶ 1976\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.33.2.2 RpcStructInstance.InvokeRpcMethodAsync Method

Overload List

	Name	Description
	InvokeRpcMethodAsync(String, .Object., CancellationTokens) [▶ 1981]	Invokes the specified RPC Method asynchronously
	InvokeRpcMethodAsync(String, .Object., .AnyTypeSpecifier, .AnyTypeSpecifier, CancellationTokens) [▶ 1983]	Invokes the specified RPC Method asynchronously

Reference

[RpcStructInstance Class \[▶ 1966\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.33.2.2.1 RpcStructInstance.InvokeRpcMethodAsync Method (String, .Object., CancellationTokens)

Invokes the specified RPC Method asynchronously

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public Task<ResultRpcMethodAccess> InvokeRpcMethodAsync(
    string methodName,
    Object[]? inParameters,
    CancellationToken cancel
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
cancel	Type: System.Threading.CancellationToken The cancellation token

Return Value

Type: [Task.ResultRpcMethodAccess](#) [► 3213].

A task that represents the asynchronous 'InvokeRpcMethod' operation. The [ResultRpcMethodAccess](#) [► 3213] results contains the return value ([ReturnValue](#) [► 3216]) together with the output parameters. The succeeded communication is indicated by the [ErrorCode](#) property ([ErrorCode](#) [► 3202]) after the communication.

Implements

[IRpcCallableInstance.InvokeRpcMethodAsync\(String, .Object., CancellationToken\)](#) [► 2614]

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [► 2094] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program
        }
    }
}
```

```

// Call a Method that has the following signature (within MAIN Program)
/* {attribute 'TcRpcEnable'}
METHOD PUBLIC M_Add : INT
VAR_INPUT
    i1 : INT := 0;
    i2 : INT := 0;
END_VAR
*/

short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

// Call a Method that has no parameter and returns VOID
main.InvokeRpcMethod("M_Method1", new object[] {});

//Browsing RpcMethods
foreach(IRpcMethod method in main.RpcMethods)
{
    string methodName = method.Name;

    foreach(IRpcMethodParameter parameter in method.Parameters)
    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
}

```

Reference

[RpcStructInstance Class \[► 1966\]](#)

[InvokeRpcMethodAsync Overload \[► 1981\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.33.2.2 RpcStructInstance.InvokeRpcMethodAsync Method (String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, CancellationToken)

Invokes the specified RPC Method asynchronously

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```

public Task<ResultRpcMethodAccess> InvokeRpcMethodAsync(
    string methodName,
    Object[]? inParameters,
    AnyTypeSpecifier[]? outSpecifiers,
    AnyTypeSpecifier? retSpecifier,
    CancellationToken cancel
)

```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [► 2094] . The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [► 2094] The ret specifier (specifying the return value) or NULL.

cancel Type: [System.Threading.CancellationToken](#)
The cancellation token

Return Value

Type: [Task.ResultRpcMethodAccess](#) [▶ 3213].

A task that represents the asynchronous 'InvokeRpcMethod' operation. The [ResultRpcMethodAccess](#) [▶ 3213] results contains the return value ([ReturnValue](#) [▶ 3216]) together with the output parameters. The succeeded communication is indicated by the [ErrorCode](#) property ([ErrorCode](#) [▶ 3202]) after the communication.

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            /* Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```



```

    }
  }
}

```

Reference









[RpcStructInstance Class \[▶ 1966\]](#)

[InvokeRpcMethodAsync Overload \[▶ 1981\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.33.2.3 RpcStructInstance.TryInvokeRpcMethod Method

Overload List

	Name	Description
 	TryInvokeRpcMethod(String, .Object., Object.) [▶ 1985]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(String, .Object., .Object., Object.) [▶ 1987]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(IRpcMethod, .Object., .AnyTypeSpecifier, .AnyTypeSpecifier, .Object., Object.) [▶ 1989]	Invokes the the specified RpcMethod of the IRpcCallableInstance [▶ 2606] .
 	TryInvokeRpcMethod(String, .Object., .AnyTypeSpecifier, .AnyTypeSpecifier, .Object., Object.) [▶ 1990]	Invokes the the specified RpcMethod of the IRpcCallableInstance [▶ 2606] .

Reference

[RpcStructInstance Class \[▶ 1966\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.33.2.3.1 RpcStructInstance.TryInvokeRpcMethod Method (String, .Object., Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public int TryInvokeRpcMethod(
    string methodName,
    Object[]? inParameters,
    out Object?? retValue
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
retValue	Type: System.Object . The return value of the RPC method as object.

Return Value

Type: [Int32](#)
The result value of the call (ErrorCode). 0 means Succeeded.

Implements

[IRpcCallableInstance.TryInvokeRpcMethod\(String, .Object., Object.\)](#) [[▶ 2619](#)]

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [[▶ 2094](#)] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
```

```
        i1 : INT := 0;
        i2 : INT := 0;
    END_VAR
    */

    short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

    // Call a Method that has no parameter and returns VOID
    main.InvokeRpcMethod("M_Method1", new object[] {});

    //Browsing RpcMethods
    foreach(IRpcMethod method in main.RpcMethods)
    {
        string methodName = method.Name;

        foreach(IRpcMethodParameter parameter in method.Parameters)
        {
            string parameterName = parameter.Name;
            string parameterType = parameter.TypeName;
        }
    }
}
```

Reference

[RpcStructInstance Class \[► 1966\]](#)

[TryInvokeRpcMethod Overload \[► 1985\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.33.2.3.2 RpcStructInstance.TryInvokeRpcMethod Method (String, .Object., .Object., Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public int TryInvokeRpcMethod(
    string methodName,
    Object[]? inParameters,
    out Object[]? outParameters,
    out Object?? retValue
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object. The parameters.
outParameters	Type: .System.Object. The out parameters.
retValue	Type: System.Object. The return value of the RPC method as object.

Return Value

Type: [Int32](#)

The result value of the call (ErrorCode). 0 means Succeeded.

Implements

[IRpcCallableInstance.InvokeRpcMethod\(String, Object, Object, Object.\)](#) [► 2620]

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [► 2094] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```

Reference

[RpcStructInstance Class](#) [► 1966]

[TryInvokeRpcMethod Overload \[▶ 1985\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▶ 1698\]](#)

6.7.33.2.3.3 **RpcStructInstance.TryInvokeRpcMethod Method (IRpcMethod, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object., Object.)**

Invokes the the specified RpcMethod of the [IRpcCallableInstance \[▶ 2606\]](#).

Namespace: [TwinCAT.Ads.TypeSystem \[▶ 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public int TryInvokeRpcMethod(
    IRpcMethod method,
    Object[]? inParameters,
    AnyTypeSpecifier[]? outSpecifiers,
    AnyTypeSpecifier? retSpecifier,
    out Object[]? outParameters,
    out Object?? retValue
)
```

Parameters

method	Type: TwinCAT.TypeSystem.IRpcMethod [▶ 2625] The method.
inParameters	Type: .System.Object . The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094] . The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094] The ret specifier (specifying the return value) or NULL.
outParameters	Type: .System.Object . The out parameters.
retValue	Type: System.Object . The return value of the RPC method./>

Return Value

Type: [Int32](#)
[AdsErrorCode](#).

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
```

```

/// Defines the entry point of the application.
/// </summary>
/// <param name="args">The arguments.</param>
static void Main(string[] args)
{
    // Get the AdsAddress from command-line arguments
    AmsAddress address = ArgParser.Parse(args);

    using (AdsClient client = new AdsClient())
    {
        //client.Synchronize = false;

        // Connect to the target device
        client.Connect(address);

        SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
        ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

        // Get the Symbols (Dynamic Symbols)

        IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

        // Call a Method that has the following signature (within MAIN Program)
        /* {attribute 'TcRpcEnable'}
        METHOD PUBLIC M_Add : INT
        VAR_INPUT
            i1 : INT := 0;
            i2 : INT := 0;
        END_VAR
        */

        short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

        // Call a Method that has no parameter and returns VOID
        main.InvokeRpcMethod("M_Method1", new object[] {});

        //Browsing RpcMethods
        foreach(IRpcMethod method in main.RpcMethods)
        {
            string methodName = method.Name;

            foreach(IRpcMethodParameter parameter in method.Parameters)
            {
                string parameterName = parameter.Name;
                string parameterType = parameter.TypeName;
            }
        }
    }
}

```

Reference

[RpcStructInstance Class \[► 1966\]](#)

[TryInvokeRpcMethod Overload \[► 1985\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.33.2.3.4 **RpcStructInstance.TryInvokeRpcMethod Method (String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object., Object.)**

Invokes the the specified RpcMethod of the [IRpcCallableInstance \[► 2606\]](#).

Namespace: [TwinCAT.Ads.TypeSystem \[► 1698\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public int TryInvokeRpcMethod(
    string methodName,
    Object[]? inParameters,
    AnyTypeSpecifier[]? outSpecifiers,
    AnyTypeSpecifier? retSpecifier,
    out Object[]? outParameters,
    out Object?? retValue
)
```

Parameters

methodName	Type: System.String Name of the method.
inParameters	Type: .System.Object . The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094]. The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094] The ret specifier (specifying the return value) or NULL.
outParameters	Type: .System.Object . The out parameters.
retValue	Type: System.Object . The return value of the RPC method./>

Return Value

Type: [Int32](#)
AdsErrorCode.

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)
```

```

    IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

    // Call a Method that has the following signature (within MAIN Program)
    /* {attribute 'TcRpcEnable'}
    METHOD PUBLIC M_Add : INT
    VAR_INPUT
        i1 : INT := 0;
        i2 : INT := 0;
    END_VAR
    */

    short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

    // Call a Method that has no parameter and returns VOID
    main.InvokeRpcMethod("M_Method1", new object[] {});

    //Browsing RpcMethods
    foreach(IRpcMethod method in main.RpcMethods)
    {
        string methodName = method.Name;

        foreach(IRpcMethodParameter parameter in method.Parameters)
        {
            string parameterName = parameter.Name;
            string parameterType = parameter.TypeName;
        }
    }
}

```

Reference

[RpcStructInstance Class \[► 1966\]](#)



[TryInvokeRpcMethod Overload \[► 1985\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.33.3 RpcStructInstance Events

The [RpcStructInstance \[► 1966\]](#) type exposes the following members.

Events

	Name	Description
	RawValueChanged [► 1913]	Occurs when the RawValue of the IValueSymbol [► 2775] has changed. (Inherited from Symbol [► 1863] .)
	ValueChanged [► 1913]	Occurs when the (Primitive) value of the IValueSymbol [► 2775] has changed. (Inherited from Symbol [► 1863] .)

Reference

[RpcStructInstance Class \[► 1966\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[► 1698\]](#)

6.7.34 UnionInstance Class

Class representing a Union Instance

Inheritance Hierarchy

System.Object

 TwinCAT.Ads.TypeSystem.Instance [[▶ 1765](#)]

 TwinCAT.Ads.TypeSystem.Symbol [[▶ 1863](#)]

 TwinCAT.Ads.TypeSystem.UnionInstance

Namespace: TwinCAT.Ads.TypeSystem [[▶ 1698](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public sealed class UnionInstance : Symbol,
    IUnionInstance, ISymbol, IAttributedInstance, IInstance, IBitSize
```

The UnionInstance type exposes the following members.

Properties

Name	Description
AccessRights [▶ 1872]	Gets the access rights. (Inherited from Symbol [▶ 1863].)
Attributes [▶ 1768]	Gets the Type Attributes. (Inherited from Instance [▶ 1765].)
BitSize [▶ 1768]	Gets the size of this Instance [▶ 1765] in bits. (Inherited from Instance [▶ 1765].)
ByteSize [▶ 1769]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from Instance [▶ 1765].)
Category [▶ 1769]	Gets the the DataTypeCategory [▶ 2111] of the Instance. (Inherited from Instance [▶ 1765].)
Comment [▶ 1770]	Gets the comment. (Inherited from Instance [▶ 1765].)
Connection [▶ 1873]	Gets the connection that produces values for this IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
ContextMask [▶ 1770]	Gets the context mask of this instance. (Inherited from Instance [▶ 1765].)
DataType [▶ 1771]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from Instance [▶ 1765].)
FieldInstances [▶ 1998]	Gets the member instances of the Struct Instance [▶ 2666].
HasValue [▶ 1771]	Gets a value indicating whether this instance has a value. (Inherited from Instance [▶ 1765].)
ImageBaseAddress [▶ 1873]	Gets the AmsAddress [▶ 752] of the Process Image (Inherited from Symbol [▶ 1863].)
IndexGroup [▶ 1874]	Gets the index group of the Symbol (Inherited from Symbol [▶ 1863].)
IndexOffset [▶ 1874]	Gets the index offset of the Symbol (Inherited from Symbol [▶ 1863].)
InstanceName [▶ 1771]	Gets the name of the instance (without periods (.)) (Inherited from Instance [▶ 1765].)
InstancePath [▶ 1874]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from Symbol [▶ 1863].)
IsBitType [▶ 1773]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from Instance [▶ 1765].)

Name	Description
IsByteAligned [▶ 1773]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from Instance [▶ 1765].)
IsContainerType [▶ 1999]	Gets a value indicating whether this Symbol is a container/complex type. (Overrides Symbol.IsContainerType [▶ 1875].)
IsDereferencedPointer [▶ 1876]	Gets or sets a value indicating whether an ancestor is a dereferenced Pointer (Inherited from Symbol [▶ 1863].)
IsDereferencedReference [▶ 1876]	Gets or sets a value indicating whether an ancestor is a dereferenced Reference (Inherited from Symbol [▶ 1863].)
IsPersistent [▶ 1773]	Indicates that this instance is persistent. (Inherited from Instance [▶ 1765].)
IsPointer [▶ 1774]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
IsPrimitiveType [▶ 1999]	Gets a value indicating whether this instance is primitive. (Overrides Symbol.IsPrimitiveType [▶ 1877].)
IsReadOnly [▶ 1774]	Indicates that this instance is read only. (Inherited from Instance [▶ 1765].)
IsRecursive [▶ 1877]	Gets a value indicating whether this instance is recursive. (Inherited from Symbol [▶ 1863].)
IsReference [▶ 1775]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
IsStatic [▶ 1775]	Gets a value indicating whether this Instance [▶ 2549] is static. (Inherited from Instance [▶ 1765].)
IsTcComInterfacePointer [▶ 1776]	Indicates that this instance is a TcComInterfacePointer. (Inherited from Instance [▶ 1765].)
IsTypeGuid [▶ 1776]	Indicates that this instance has set TypeGuid flag. (Inherited from Instance [▶ 1765].)
IsVirtual [▶ 1878]	Gets a value indicating whether this instance is virtual. (Inherited from Symbol [▶ 1863].)
Namespace [▶ 1776]	Gets the namespace name. (Inherited from Instance [▶ 1765].)
NotificationSettings [▶ 1878]	Gets or sets the notification settings. (Inherited from Symbol [▶ 1863].)
Parent [▶ 1879]	Gets the parent Symbol (Inherited from Symbol [▶ 1863].)
Size [▶ 1777]	Gets the size of the IDataType [▶ 2475] in bytes or Bits dependant on IsBitType [▶ 1773] (Inherited from Instance [▶ 1765].)
SubSymbolCount [▶ 1879]	Gets the number of SubSymbols (Inherited from Symbol [▶ 1863].)
SubSymbols [▶ 1880]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from Symbol [▶ 1863].)
TypeName [▶ 1777]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from Instance [▶ 1765].)
ValueEncoding [▶ 1880]	Gets the value encoding. (Inherited from Symbol [▶ 1863].)

Methods

Name	Description
Equals(ISymbol) [▶ 1885]	Equalses the specified inst. (Inherited from Symbol [▶ 1863].)

	Name	Description
	Equals(Object) [▶ 1886]	Equals (Inherited from Symbol [▶ 1863].)
	Equals(Symbol) [▶ 1887]	Equalses the specified inst. (Inherited from Symbol [▶ 1863].)
	GetHashCode [▶ 1887]	Gets the GetHashCode of the Address (Inherited from Symbol [▶ 1863].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ReadAnyValue(Type) [▶ 1894]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from Symbol [▶ 1863].)
	ReadAnyValue(Type, Int32) [▶ 1895]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from Symbol [▶ 1863].)
	ReadAnyValue.T. [▶ 1896]	Reads the value of this Value [▶ 2753] into a new created instance of the managed type (Inherited from Symbol [▶ 1863].)
	ReadAnyValue.T. (Int32) [▶ 1896]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from Symbol [▶ 1863].)
	ReadAnyValueAsync(Type, CancellationToken) [▶ 1898]	Reads the (AnyType) value asynchronously. (Inherited from Symbol [▶ 1863].)
	ReadAnyValueAsync.T. (CancellationToken) [▶ 1898]	Read any value as an asynchronous operation. (Inherited from Symbol [▶ 1863].)
	ReadRawValue. [▶ 1899]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from Symbol [▶ 1863].)
	ReadRawValue(Int32) [▶ 1900]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from Symbol [▶ 1863].)
	ReadRawValueAsync [▶ 1901]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) asynchronously. (Inherited from Symbol [▶ 1863].)
	ReadValue. [▶ 1902]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	ReadValue(Int32) [▶ 1902]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	ReadValueAsync [▶ 1903]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from Symbol [▶ 1863].)
	SetParent [▶ 1904]	Sets the parent symbol. (Inherited from Symbol [▶ 1863].)
	ToString [▶ 1904]	Returns a String that represents this instance. (Inherited from Symbol [▶ 1863].)
	TryReadValue [▶ 1905]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	TryWriteValue [▶ 1905]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)

	Name	Description
	UpdateAnyValue(Object) [1907]	Reads the value of this Value [2775] into the specified managed value. (Inherited from Symbol [1863].)
	UpdateAnyValue(Object, Int32) [1907]	Reads the value of this Value [2775] into the specified managed value. (Inherited from Symbol [1863].)
	WriteRawValue(Byte) [1908]	Writes the raw value of the IValueSymbol [2775] (Ads Read / Write) (Inherited from Symbol [1863].)
	WriteRawValue(Byte, Int32) [1909]	Writes the raw value of the IValueSymbol [2775] (Ads Read / Write) (Inherited from Symbol [1863].)
	WriteRawValueAsync [1910]	Writes the raw value of the IValueSymbol [2775] (Ads Read / Write) (Inherited from Symbol [1863].)
	WriteValue(Object) [1910]	Writes the Value of the IValueSymbol [2775] (Inherited from Symbol [1863].)
	WriteValue(Object, Int32) [1911]	Writes the Value of the IValueSymbol [2775] (Inherited from Symbol [1863].)
	WriteValueAsync [1912]	Writes the Value of the IValueSymbol [2775] (Inherited from Symbol [1863].)

Events

	Name	Description
	RawValueChanged [1913]	Occurs when the RawValue of the IValueSymbol [2775] has changed. (Inherited from Symbol [1863].)
	ValueChanged [1913]	Occurs when the (Primitive) value of the IValueSymbol [2775] has changed. (Inherited from Symbol [1863].)

Extension Methods

	Name	Description
	ReferencesExternalData [3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [3021].)

Reference

[TwinCAT.Ads.TypeSystem Namespace](#) [[1698](#)]

Also see about this

- [Symbol.Equals Method](#) [[1885](#)]
- [Symbol.ReadAnyValueAsync Method](#) [[1897](#)]

6.7.34.1 UnionInstance Properties

The [UnionInstance](#) [[1992](#)] type exposes the following members.

Properties

Name	Description
AccessRights [▶ 1872]	Gets the access rights. (Inherited from Symbol [▶ 1863] .)
Attributes [▶ 1768]	Gets the Type Attributes. (Inherited from Instance [▶ 1765] .)
BitSize [▶ 1768]	Gets the size of this Instance [▶ 1765] in bits. (Inherited from Instance [▶ 1765] .)
ByteSize [▶ 1769]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from Instance [▶ 1765] .)
Category [▶ 1769]	Gets the the DataTypeCategory [▶ 2111] of the Instance. (Inherited from Instance [▶ 1765] .)
Comment [▶ 1770]	Gets the comment. (Inherited from Instance [▶ 1765] .)
Connection [▶ 1873]	Gets the connection that produces values for this ValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863] .)
ContextMask [▶ 1770]	Gets the context mask of this instance. (Inherited from Instance [▶ 1765] .)
DataType [▶ 1771]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549] . (Inherited from Instance [▶ 1765] .)
FieldInstances [▶ 1998]	Gets the member instances of the Struct Instance [▶ 2666] .
HasValue [▶ 1771]	Gets a value indicating whether this instance has a value. (Inherited from Instance [▶ 1765] .)
ImageBaseAddress [▶ 1873]	Gets the AmsAddress [▶ 752] of the Process Image (Inherited from Symbol [▶ 1863] .)
IndexGroup [▶ 1874]	Gets the index group of the Symbol (Inherited from Symbol [▶ 1863] .)
IndexOffset [▶ 1874]	Gets the index offset of the Symbol (Inherited from Symbol [▶ 1863] .)
InstanceName [▶ 1771]	Gets the name of the instance (without periods (.)) (Inherited from Instance [▶ 1765] .)
InstancePath [▶ 1874]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from Symbol [▶ 1863] .)
IsBitType [▶ 1773]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from Instance [▶ 1765] .)
IsByteAligned [▶ 1773]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from Instance [▶ 1765] .)
IsContainerType [▶ 1999]	Gets a value indicating whether this Symbol is a container/complex type. (Overrides Symbol.IsContainerType [▶ 1875] .)
IsDereferencedPointer [▶ 1876]	Gets or sets a value indicating whether an ancestor is a dereferenced Pointer (Inherited from Symbol [▶ 1863] .)
IsDereferencedReference [▶ 1876]	Gets or sets a value indicating whether an ancestor is a dereferenced Reference (Inherited from Symbol [▶ 1863] .)
IsPersistent [▶ 1773]	Indicates that this instance is persistent. (Inherited from Instance [▶ 1765] .)
IsPointer [▶ 1774]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765] .)
IsPrimitiveType [▶ 1999]	Gets a value indicating whether this instance is primitive. (Overrides Symbol.IsPrimitiveType [▶ 1877] .)

Name	Description
IsReadOnly [▶ 1774]	Indicates that this instance is read only. (Inherited from Instance [▶ 1765].)
IsRecursive [▶ 1877]	Gets a value indicating whether this instance is recursive. (Inherited from Symbol [▶ 1863].)
IsReference [▶ 1775]	Gets a value indicating whether this instance is reference. (Inherited from Instance [▶ 1765].)
IsStatic [▶ 1775]	Gets a value indicating whether this Instance [▶ 2549] is static. (Inherited from Instance [▶ 1765].)
IsTcComInterfacePointer [▶ 1776]	Indicates that this instance is a TcComInterfacePointer . (Inherited from Instance [▶ 1765].)
IsTypeGuid [▶ 1776]	Indicates that this instance has set TypeGuid flag. (Inherited from Instance [▶ 1765].)
IsVirtual [▶ 1878]	Gets a value indicating whether this instance is virtual. (Inherited from Symbol [▶ 1863].)
Namespace [▶ 1776]	Gets the namespace name. (Inherited from Instance [▶ 1765].)
NotificationSettings [▶ 1878]	Gets or sets the notification settings. (Inherited from Symbol [▶ 1863].)
Parent [▶ 1879]	Gets the parent Symbol (Inherited from Symbol [▶ 1863].)
Size [▶ 1777]	Gets the size of the IDataType [▶ 2475] in bytes or Bits dependant on IsBitType [▶ 1773] (Inherited from Instance [▶ 1765].)
SubSymbolCount [▶ 1879]	Gets the number of SubSymbols (Inherited from Symbol [▶ 1863].)
SubSymbols [▶ 1880]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from Symbol [▶ 1863].)
TypeName [▶ 1777]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from Instance [▶ 1765].)
ValueEncoding [▶ 1880]	Gets the value encoding. (Inherited from Symbol [▶ 1863].)

Reference

[UnionInstance Class](#) [[▶ 1992](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.34.1 UnionInstance.FieldInstances Property

Gets the member instances of the [Struct Instance](#) [[▶ 2666](#)].

Namespace: [TwinCAT.Ads.TypeSystem](#) [[▶ 1698](#)]

Assembly: [TwinCAT.Ads](#) (in [TwinCAT.Ads.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public ISymbolCollection<ISymbol> FieldInstances { get; }
```

Property Value

Type: [ISymbolCollection](#) [[▶ 2700](#)].[ISymbol](#) [[▶ 2691](#)].
The member instances.

Implements

[IUnionInstance.FieldInstances](#) [► 2740]

Reference

[UnionInstance Class](#) [► 1992]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.34.1.2 UnionInstance.IsContainerType Property

Gets a value indicating whether this Symbol is a container/complex type.

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool IsContainerType { get; }
```

Property Value

Type: Boolean

true if this instance is container type; otherwise, false.

Implements

[ISymbol.IsContainerType](#) [► 2694]

[ISymbol.IsContainerType](#) [► 2694]

Reference

[UnionInstance Class](#) [► 1992]

[TwinCAT.Ads.TypeSystem Namespace](#) [► 1698]

6.7.34.1.3 UnionInstance.IsPrimitiveType Property

Gets a value indicating whether this instance is primitive.

Namespace: [TwinCAT.Ads.TypeSystem](#) [► 1698]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool IsPrimitiveType { get; }
```

Property Value

Type: Boolean

true if this instance is primitive; otherwise, false.

Implements

[ISymbol.IsPrimitiveType](#) [[▶ 2695](#)]

[ISymbol.IsPrimitiveType](#) [[▶ 2695](#)]

Reference

[UnionInstance Class](#) [[▶ 1992](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

6.7.34.2 UnionInstance Methods

The [UnionInstance](#) [[▶ 1992](#)] type exposes the following members.

Methods

	Name	Description
	Equals(ISymbol) [▶ 1885]	Equalses the specified inst. (Inherited from Symbol [▶ 1863].)
	Equals(Object) [▶ 1886]	Equals (Inherited from Symbol [▶ 1863].)
	Equals(Symbol) [▶ 1887]	Equalses the specified inst. (Inherited from Symbol [▶ 1863].)
	GetHashCode [▶ 1887]	Gets the GetHashCode of the Address (Inherited from Symbol [▶ 1863].)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ReadAnyValue(Type) [▶ 1894]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from Symbol [▶ 1863].)
	ReadAnyValue(Type, Int32) [▶ 1895]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from Symbol [▶ 1863].)
	ReadAnyValue.T. [▶ 1896]	Reads the value of this Value [▶ 2753] into a new created instance of the managed type (Inherited from Symbol [▶ 1863].)
	ReadAnyValue.T. (Int32) [▶ 1896]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from Symbol [▶ 1863].)
	ReadAnyValueAsync (Type, CancellationToken) [▶ 1898]	Reads the (AnyType) value asynchronously. (Inherited from Symbol [▶ 1863].)
	ReadAnyValueAsync.T. (CancellationToken) [▶ 1898]	Read any value as an asynchronous operation. (Inherited from Symbol [▶ 1863].)
	ReadRawValue. [▶ 1899]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from Symbol [▶ 1863].)
	ReadRawValue(Int32) [▶ 1900]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from Symbol [▶ 1863].)

	Name	Description
	ReadRawValueAsync [▶ 1901]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) asynchronously. (Inherited from Symbol [▶ 1863].)
	ReadValue. [▶ 1902]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	ReadValue(Int32) [▶ 1902]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	ReadValueAsync [▶ 1903]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from Symbol [▶ 1863].)
	SetParent [▶ 1904]	Sets the parent symbol. (Inherited from Symbol [▶ 1863].)
	ToString [▶ 1904]	Returns a String that represents this instance. (Inherited from Symbol [▶ 1863].)
	TryReadValue [▶ 1905]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	TryWriteValue [▶ 1905]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	UpdateAnyValue(Object) [▶ 1907]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from Symbol [▶ 1863].)
	UpdateAnyValue(Object, Int32) [▶ 1907]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from Symbol [▶ 1863].)
	WriteRawValue(Byte) [▶ 1908]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from Symbol [▶ 1863].)
	WriteRawValue(Byte, Int32) [▶ 1909]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from Symbol [▶ 1863].)
	WriteRawValueAsync [▶ 1910]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from Symbol [▶ 1863].)
	WriteValue(Object) [▶ 1910]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	WriteValue(Object, Int32) [▶ 1911]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)
	WriteValueAsync [▶ 1912]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from Symbol [▶ 1863].)

Extension Methods



	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[UnionInstance Class](#) [[▶ 1992](#)]

[TwinCAT.Ads.TypeSystem Namespace](#) [[▶ 1698](#)]

Also see about this

-  [Symbol.Equals Method \[▸ 1885\]](#)
-  [Symbol.ReadAnyValueAsync Method \[▸ 1897\]](#)

6.7.34.3 UnionInstance Events

The [UnionInstance \[▸ 1992\]](#) type exposes the following members.

Events

	Name	Description
	RawValueChanged [▸ 1913]	Occurs when the RawValue of the IValueSymbol [▸ 2775] has changed. (Inherited from Symbol [▸ 1863] .)
	ValueChanged [▸ 1913]	Occurs when the (Primitive) value of the IValueSymbol [▸ 2775] has changed. (Inherited from Symbol [▸ 1863] .)

Reference


[UnionInstance Class \[▸ 1992\]](#)

[TwinCAT.Ads.TypeSystem Namespace \[▸ 1698\]](#)

6.8 TwinCAT.Ads.ValueAccess Namespace

Root namespace for ADS value access.

Enumerations

	Enumeration	Description
	ValueAccessMode [▸ 2002]	Enum ValueAccessMethod

6.8.1 ValueAccessMode Enumeration

Enum ValueAccessMethod

Namespace: [TwinCAT.Ads.ValueAccess \[▸ 2002\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.116+a71ced3

Syntax**C#**

```
public enum ValueAccessMode
```

Members

	Member name	Value	Description
	None	0	None / Uninitialized
	IndexGroupOffset	1	Value access via Index Group and Offset Only
	Symbolic	2	Symbolic access via Instance Path only.
	IndexGroupOffsetPreferred	3	Uses IndexGroup IndexOffset Preferred (and Symbolic for Dereferenced Pointers / References)

	Member name	Value	Description
	Default	2	The Default access mode (Symbolic)

Remarks




Mode	Description
None	None/Uninitialized. No Valid mode.
IndexGroupOffset	Communicates over IndexGroup/IndexOffset only. This is the most direct/ efficient access into the Process image. The advantage is that, the symbol access is done via 1 ADS round trip. Disadvantages are that not all Symbols can be accessed via IG/IO (e.g. References) and IndexOffsets could be invalid after online changes / PlcProgram downloads. Detection of these events and following invalidation of all changed symbols need to be done within the user application.
Symbolic	The Symbolic-only mode is the most safe mode to use but needs more time than the IndexGroupOffset. It could need up to 3 ADS round trips (create handle, access value, close handle) but is not influenced by online changes or / plcProgram downloads.
IndexGroupOffsetPreferred	This is a mixed access mode. For symbols, where it is possible it uses the IndexGroup/IndexOffset. For others it chooses the Symbolic access.
Default	The Default-Mode setting if no other ValueAccessMode is specified. This is set to Symbolic.

Reference

[TwinCAT.Ads.ValueAccess Namespace \[► 2002\]](#)

6.9 TwinCAT.Ams Namespace


Classes

	Class	Description
	AmsConfiguration [► 2004]	Static configuration of the Ams Router system.
	AmsPortNotAvailableException [► 2013]	Class AmsPortNotAvailableException. Cannot register to the Router Port. Implements the AmsServerException [► 2009]
	AmsServerException [► 2009]	Ams Server Exception class Implements the Exception

Interfaces

	Interface	Description
	IAdsHeaderDump [► 2012]	Interface IAdsHeaderDump

Enumerations

	Enumeration	Description
	AmsServerErrorCode [► 2008]	Ams Server Error Codes.

6.9.1 AmsConfiguration Class

Static configuration of the Ams Router system.

Inheritance Hierarchy

System.Object

TwinCAT.Ams.AmsConfiguration

Namespace: [TwinCAT.Ams](#) [[▶ 2003](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229









Syntax

C#





```
public static class AmsConfiguration
```

The AmsConfiguration type exposes the following members.

Properties

	Name	Description
 	DefaultRouterEndPo int [▶ 2005]	Gets the default router end point (IPAddress.Loopback, Port 0xBF02)
 	RouterAddress [▶ 2005]	Gets the actually configured router address.
 	RouterEndPoint [▶ 2006]	Gets or sets the default router IPEndPoint.
 	RouterPort [▶ 2007]	Gets actually configured router port.

Fields

	Name	Description
 	DEFAULT_TCP_PORT [▶ 2007]	The default TCP port (0xBF02, 48898), unsecured
 	DEFAULT_TCP_PORT ADSSECURE [▶ 2008]	The default ADSSecure Port (0x1f50, 8016), secured

Remarks

If the router is intended to run separately of the (virtual) system that instantiates the AdsClient or AdsServer (e.g. in UnitTests), some global static settings must be adopted to enable different network scenarios. In the default case the AdsServers and AdsClients use the Loopback TCP port **0xBF02** to communicate internally. If the systems of AdsServer/AdsClient are split from the router (e.g. running the router isolated in a virtual machine), the Router Endpoint must be set via [RouterEndPoint](#) [[▶ 2006](#)] setter.





Reference

[TwinCAT.Ams Namespace](#) [[▶ 2003](#)]

6.9.1.1 AmsConfiguration Properties

The [AmsConfiguration](#) [▸ 2004] type exposes the following members.

Properties

	Name	Description
 S	DefaultRouterEndPoint [▸ 2005]	Gets the default router end point (IPAddress.Loopback, Port 0xBF02)
 S	RouterAddress [▸ 2005]	Gets the actually configured router address.
 S	RouterEndPoint [▸ 2006]	Gets or sets the default router IPEndPoint.
 S	RouterPort [▸ 2007]	Gets actually configured router port.

Reference

[AmsConfiguration Class](#) [▸ 2004]

[TwinCAT.Ads Namespace](#) [▸ 2003]

6.9.1.1.1 AmsConfiguration.DefaultRouterEndPoint Property

Gets the default router end point (IPAddress.Loopback, Port 0xBF02)

Namespace: [TwinCAT.Ads](#) [▸ 2003]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IPEndPoint DefaultRouterEndPoint { get; }
```

Property Value

Type: IPEndPoint

The default router end point.

Reference

[AmsConfiguration Class](#) [▸ 2004]

[TwinCAT.Ads Namespace](#) [▸ 2003]

6.9.1.1.2 AmsConfiguration.RouterAddress Property

Gets the actually configured router address.

Namespace: [TwinCAT.Ads](#) [▸ 2003]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IPAddress RouterAddress { get; }
```

Property Value

Type: IPAddress
The router address.

Reference

[AmsConfiguration Class \[► 2004\]](#)

[TwinCAT.Ads Namespace \[► 2003\]](#)

[AmsConfiguration.RouterEndPoint \[► 2006\]](#)

6.9.1.1.3 AmsConfiguration.RouterEndPoint Property

Gets or sets the default router IPEndPoint.

Namespace: [TwinCAT.Ads \[► 2003\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IPEndPoint RouterEndPoint { get; set; }
```

Property Value

Type: IPEndPoint
The IPEndPoint used by Ams Clients/Servers.

Remarks

This setting allows to change the Router Endpoint - which listens by default on **IPAddress 127.0.0.1, Port 0xBF02**. An application for this could be to move the router to a different (virtual) system, separated from the system running the **AdsServer** or **AdsClient**. This endpoint can only be changed process wide and should be done before the first access to the Router occurs (e.g. getting the local NetId).

ATTENTION: Bending this Endpoint is not possible with the standard TwinCAT Router. As security feature the TwinCAT Router only accepts Loopback connections to 127.0.0.1, connection requests from the outside will be closed immediately.

The simplistic .NET Core Router implementation in class 'TwinCAT.Ads.AdsRouter.AmsTcpIpRouter' from nuget package '[Beckhoff.TwinCAT.Ads.TcpRouter](#)' or the '[Beckhoff.TwinCAT.Ads.AdsRouterConsole](#)' doesn't have that restriction and can be used more flexible for use with virtual environments like VirtualMachines or Docker. flexible.

The following sample demonstrates how to configure the (internal) RouterEndPoint to IPAddress 1.2.3.4:42 . This must be done before AdsServer or AdsClient are instantiated.

```
AmsConfiguration.RouterEndPoint = new IPEndPoint(IPAddress.Parse("1.2.3.4"), 42);
```

Reference

[AmsConfiguration Class \[► 2004\]](#)

[TwinCAT.Ads Namespace \[► 2003\]](#)

6.9.1.1.4 AmsConfiguration.RouterPort Property

Gets actually configured router port.

Namespace: [TwinCAT.Ams \[▸ 2003\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static int RouterPort { get; }
```

Property Value

Type: Int32

The router port.

Reference

[AmsConfiguration Class \[▸ 2004\]](#)





[TwinCAT.Ams Namespace \[▸ 2003\]](#)

[AmsConfiguration.RouterEndPoint \[▸ 2006\]](#)

6.9.1.2 AmsConfiguration Fields

The [AmsConfiguration \[▸ 2004\]](#) type exposes the following members.

Fields

	Name	Description
 	DEFAULT_TCP_PORT [▸ 2007]	The default TCP port (0xBF02, 48898), unsecured
 	DEFAULT_TCP_PORT_ADSSECURE [▸ 2008]	The default ADSSecure Port (0x1f50, 8016), secured

Reference

[AmsConfiguration Class \[▸ 2004\]](#)

[TwinCAT.Ams Namespace \[▸ 2003\]](#)

6.9.1.2.1 AmsConfiguration.DEFAULT_TCP_PORT Field

The default TCP port (0xBF02, 48898), unsecured

Namespace: [TwinCAT.Ams \[▸ 2003\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public const int DEFAULT_TCP_PORT = 48898
```

Field Value

Type: Int32

Reference[AmsConfiguration Class \[► 2004\]](#)[TwinCAT.Ams Namespace \[► 2003\]](#)**6.9.1.2.2 AmsConfiguration.DEFAULT_TCP_PORT_ADSSecure Field**

The default ADSSecure Port (0x1f50, 8016), secured

Namespace: [TwinCAT.Ams \[► 2003\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public const int DEFAULT_TCP_PORT_ADSSecure = 8016
```

Field Value

Type: Int32

Reference[AmsConfiguration Class \[► 2004\]](#)[TwinCAT.Ams Namespace \[► 2003\]](#)**6.9.2 AmsServerErrorCode Enumeration**

Ams Server Error Codes.

Namespace: [TwinCAT.Ams \[► 2003\]](#)**Assembly:** TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public enum AmsServerErrorCode
```

Members

	Member name	Value	Description
	None	0	None / Uninitialized
	ConnectPortFailed	1	Connecting port failed.
	DisconnectPortFailed	2	Disconnecting port failed.
	ReceiveQueueOverflow	3	ReceiveQueue overflow
	ReceiveNotificationQueueOverflow	4	Receive Notification Queue overflow.

Reference[TwinCAT.Ams Namespace \[► 2003\]](#)

6.9.3 AmsServerException Class

Ams Server Exception class Implements the Exception

Inheritance Hierarchy

System.Object
 System.Exception
 TwinCAT.Ads.AmsServerException
 TwinCAT.Ads.AmsPortNotAvailableException [[▶ 2013](#)]

Namespace: [TwinCAT.Ads](#) [[▶ 2003](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229


Syntax

C#









```
[SerializableAttribute]
public class AmsServerException : Exception
```

The AmsServerException type exposes the following members.




Constructors






	Name	Description
	AmsServerException [▶ 2010]	Initializes a new instance of the AmsServerException class.

Properties


	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)

	Name	Description
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)

Reference

[TwinCAT.Ams Namespace \[► 2003\]](#)

System.Exception

6.9.3.1 AmsServerException Constructor

Initializes a new instance of the [AmsServerException \[► 2009\]](#) class.

Namespace: [TwinCAT.Ams \[► 2003\]](#)

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected AmsServerException(
    SerializationInfo serializationInfo,
    StreamingContext streamingContext
)
```

Parameters

serializationInfo	Type: System.Runtime.Serialization.SerializationInfo The serialization information.
streamingContext	Type: System.Runtime.Serialization.StreamingContext The streaming context.

Exceptions

Exception	Condition
ArgumentNullException	serializationInfo

Reference









[AmsServerException Class \[► 2009\]](#)

[TwinCAT.Ams Namespace \[► 2003\]](#)

6.9.3.2 AmsServerException Properties

The [AmsServerException](#) [► 2009] type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Reference









[AmsServerException Class](#) [► 2009]

[TwinCAT.Ads Namespace](#) [► 2003]

6.9.3.3 AmsServerException Methods

The [AmsServerException](#) [► 2009] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Reference


[AmsServerException Class](#) [► 2009]

[TwinCAT.Ams Namespace \[► 2003\]](#)

6.9.3.4 AmsServerException Events

The [AmsServerException \[► 2009\]](#) type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)

Reference

[AmsServerException Class \[► 2009\]](#)

[TwinCAT.Ams Namespace \[► 2003\]](#)

6.9.4 IAdsHeaderDump Interface

Interface IAdsHeaderDump

Namespace: [TwinCAT.Ams \[► 2003\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public interface IAdsHeaderDump
```

The IAdsHeaderDump type exposes the following members.

Methods

	Name	Description
	Dump [► 2013]	Dumps the Header (only for internal debug purposes)

Reference

[TwinCAT.Ams Namespace \[► 2003\]](#)

6.9.4.1 IAdsHeaderDump Methods

The [IAdsHeaderDump \[► 2012\]](#) type exposes the following members.

Methods

	Name	Description
	Dump [► 2013]	Dumps the Header (only for internal debug purposes)

Reference

[IAdsHeaderDump Interface \[► 2012\]](#)

[TwinCAT.Ams Namespace \[► 2003\]](#)

6.9.4.1.1 IAdsHeaderDump.Dump Method

Dumps the Header (only for internal debug purposes)

Namespace: [TwinCAT.Ams](#) [[▶ 2003](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string Dump()
```

Return Value

Type: String
System.String.

Reference

[IAdsHeaderDump Interface](#) [[▶ 2012](#)]

[TwinCAT.Ams Namespace](#) [[▶ 2003](#)]

6.9.5 AmsPortNotAvailableException Class

Class AmsPortNotAvailableException. Cannot register to the Router Port. Implements the [AmsServerException](#) [[▶ 2009](#)]

Inheritance Hierarchy

System.Object
System.Exception
[TwinCAT.Ams.AmsServerException](#) [[▶ 2009](#)]
TwinCAT.Ams.AmsPortNotAvailableException

Namespace: [TwinCAT.Ams](#) [[▶ 2003](#)]

Assembly: TwinCAT.Ads.Server (in TwinCAT.Ads.Server.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public class AmsPortNotAvailableException : AmsServerException
```

The AmsPortNotAvailableException type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)

	Name	Description
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)

Remarks

No router port available. Usually this error occurs when the local TwinCAT Router is not started or if it doesn't support the TcpIp Loopback channel

Reference

[TwinCAT.Ads Namespace \[► 2003\]](#)

[TwinCAT.Ads.AmsServerException \[► 2009\]](#)

6.9.5.1 AmsPortNotAvailableException Properties

The [AmsPortNotAvailableException \[► 2013\]](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	Name	Description
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Reference

[AmsPortNotAvailableException Class \[► 2013\]](#)

[TwinCAT.Ams Namespace \[► 2003\]](#)

6.9.5.2 AmsPortNotAvailableException Methods

The [AmsPortNotAvailableException \[► 2013\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Reference

[AmsPortNotAvailableException Class \[► 2013\]](#)

[TwinCAT.Ams Namespace \[► 2003\]](#)

6.9.5.3 AmsPortNotAvailableException Events

The [AmsPortNotAvailableException \[► 2013\]](#) type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)









Reference

[AmsPortNotAvailableException Class \[► 2013\]](#)



[TwinCAT.Ads Namespace \[► 2003\]](#)

6.10 TwinCAT.PlcOpen Namespace

Classes

	Class	Description
	DATE [► 2016]	PlcOpen DATE class (32-bit)
	DateBase [► 2023]	PlcOpen Date base class (32-Bit)
	DT [► 2034]	PlcOpen DT (DATE_AND_TIME) datatype.
	LTIME [► 2045]	PlcOpen LTIME class
	LTimeBase [► 2053]	Time base class
	TIME [► 2061]	PlcOpen TIME class
	TimeBase [► 2068]	Base class for PlcOpen Time types.
	TOD [► 2077]	PLCOpen TimeOfDay class (32-Bit)

Interfaces

	Interface	Description
	IPlcOpenTimeBase [► 2041]	Interface IPlcOpenType
	IPlcOpenTimeBase.T1, T2. [► 2043]	Interface IPlcOpenType

6.10.1 DATE Class

PlcOpen DATE class (32-bit)

Inheritance Hierarchy

System.Object

[TwinCAT.PlcOpen.DateBase \[► 2023\]](#)

TwinCAT.PlcOpen.DATE

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax

C#



```
public sealed class DATE : DateBase
```

The DATE type exposes the following members.








Constructors

	Name	Description
	DATE. [▶ 2018]	Initializes a new instance of the DATE class.
	DATE(DateTime) [▶ 2020]	Initializes a new instance of the DATE class.
	DATE(DateTimeOffset) [▶ 2018]	Initializes a new instance of the DATE class.
	DATE(UInt32) [▶ 2019]	Initializes a new instance of the DATE class.
	DATE(Int32, Int32, Int32) [▶ 2019]	Initializes a new instance of the DATE class.

Properties

	Name	Description
	Date [▶ 2021]	Gets or the date value (only the Date part of the DateTime)
	Ticks [▶ 2028]	Returns the number of ticks that represent the value of this DateBase [▶ 2023]. (Inherited from DateBase [▶ 2023].)

Methods



	Name	Description
	Equals [▶ 2030]	Determines whether the specified Object is equal to this instance. (Inherited from DateBase [▶ 2023].)
	GetHashCode [▶ 2031]	Gets the GetHashCode of the Address (Inherited from DateBase [▶ 2023].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	Parse [▶ 2022]	Parses the specified string to a DATE object.
	ToString [▶ 2022]	Returns a String that represents this instance. (Overrides Object.ToString..)
	TryParse [▶ 2023]	Tries to parse the specified string to a DATE object.
		




Reference

[TwinCAT.PlcOpen Namespace](#) [[▶ 2016](#)]

6.10.1.1 DATE Constructor

Overload List

	Name	Description
	DATE. [▶ 2018]	Initializes a new instance of the DATE [▶ 2016] class.
	DATE(DateTime) [▶ 2020]	Initializes a new instance of the DATE [▶ 2016] class.

	Name	Description
	DATE(DateTimeOffset) [2018]	Initializes a new instance of the DATE [2016] class.
	DATE(UInt32) [2019]	Initializes a new instance of the DATE [2016] class.
	DATE(Int32, Int32, Int32) [2019]	Initializes a new instance of the DATE [2016] class.

Reference

[DATE Class](#) [[2016](#)]

[TwinCAT.PlcOpen Namespace](#) [[2016](#)]

6.10.1.1.1 DATE Constructor

Initializes a new instance of the [DATE](#) [[2016](#)] class.

Namespace: [TwinCAT.PlcOpen](#) [[2016](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DATE()
```

Reference

[DATE Class](#) [[2016](#)]

[DATE Overload](#) [[2017](#)]

[TwinCAT.PlcOpen Namespace](#) [[2016](#)]

6.10.1.1.2 DATE Constructor (DateTimeOffset)

Initializes a new instance of the [DATE](#) [[2016](#)] class.

Namespace: [TwinCAT.PlcOpen](#) [[2016](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DATE(
    DateTimeOffset date
)
```

Parameters

date	Type: System.DateTimeOffset The date.
------	--

Reference

[DATE Class](#) [[2016](#)]

[DATE Overload](#) [[2017](#)]

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.1.1.3 DATE Constructor (Int64)

Initializes a new instance of the [DATE \[► 2016\]](#) class.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public DATE(  
    long dateValue  
)
```

Parameters

dateValue	Type: System.Int64 The date value in PlcOpen Ticks.
-----------	--

Reference

[DATE Class \[► 2016\]](#)

[DATE Overload \[► 2017\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.1.1.4 DATE Constructor (UInt32)

Initializes a new instance of the [DATE \[► 2016\]](#) class.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DATE(  
    uint dateValue  
)
```

Parameters

dateValue	Type: System.UInt32 The date value in PlcOpen Ticks.
-----------	---

Reference

[DATE Class \[► 2016\]](#)

[DATE Overload \[► 2017\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.1.1.5 DATE Constructor (Int32, Int32, Int32)

Initializes a new instance of the [DATE \[► 2016\]](#) class.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DATE(  
    int year,  
    int month,  
    int day  
)
```

Parameters

year	Type: System.Int32 The year.
month	Type: System.Int32 The month.
day	Type: System.Int32 The day.

Reference

[DATE Class \[► 2016\]](#)

[DATE Overload \[► 2017\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.1.1.6 DATE Constructor (DateTime)

Initializes a new instance of the [DATE \[► 2016\]](#) class.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DATE(  
    DateTime date  
)
```

Parameters

date	Type: System.DateTime The date.
------	------------------------------------

Reference

[DATE Class \[► 2016\]](#)



[DATE Overload \[► 2017\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.1.2 DATE Properties

The [DATE \[► 2016\]](#) type exposes the following members.

Properties

	Name	Description
	Date [▶ 2021]	Gets or the date value (only the Date part of the DateTime)
	Ticks [▶ 2028]	Returns the number of ticks that represent the value of this DateBase [▶ 2023]. (Inherited from DateBase [▶ 2023].)

Reference

[DATE Class](#) [[▶ 2016](#)]

[TwinCAT.PlcOpen Namespace](#) [[▶ 2016](#)]

6.10.1.2.1 DATE.Date Property

Gets or the date value (only the Date part of the DateTime)

Namespace: [TwinCAT.PlcOpen](#) [[▶ 2016](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTime Date { get; }
```

Property Value

Type: DateTime

The date.

Reference







[DATE Class](#) [[▶ 2016](#)]

[TwinCAT.PlcOpen Namespace](#) [[▶ 2016](#)]

6.10.1.3 DATE Methods

The [DATE](#) [[▶ 2016](#)] type exposes the following members.

Methods

	Name	Description
	Equals [▶ 2030]	Determines whether the specified Object is equal to this instance. (Inherited from DateBase [▶ 2023].)
	GetHashCode [▶ 2031]	Gets the GetHashCode of the Address (Inherited from DateBase [▶ 2023].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	Parse [▶ 2022]	Parses the specified string to a DATE [▶ 2016] object.
	ToString [▶ 2022]	Returns a String that represents this instance. (Overrides Object.ToString..)
	TryParse [▶ 2023]	Tries to parse the specified string to a DATE [▶ 2016] object.

Reference[DATE Class \[► 2016\]](#)[TwinCAT.PlcOpen Namespace \[► 2016\]](#)**6.10.1.3.1 DATE.Parse Method**

Parses the specified string to a [DATE \[► 2016\]](#) object.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static DATE Parse(
    string s
)
```

Parameters

s	Type: System.String The s.
---	-------------------------------

Return ValueType: [DATE \[► 2016\]](#)

DATE.

Exceptions

Exception	Condition
FormatException	Cannot parse DATE object!

Reference[DATE Class \[► 2016\]](#)[TwinCAT.PlcOpen Namespace \[► 2016\]](#)**6.10.1.3.2 DATE.ToString Method**

Returns a String that represents this instance.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public override string ToString()
```

Return Value

Type: String

A String that represents this instance.

Reference[DATE Class \[► 2016\]](#)[TwinCAT.PlcOpen Namespace \[► 2016\]](#)**6.10.1.3.3 DATE.TryParse Method**

Tries to parse the specified string to a [DATE \[► 2016\]](#) object.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static bool TryParse(
    string s,
    out DATE?? date
)
```

Parameters

s	Type: System.String The s.
date	Type: TwinCAT.PlcOpen.DATE [► 2016] . The date.

Return Value

Type: Boolean

true if the string could be parsed to [DATE \[► 2016\]](#), false otherwise.**Reference**[DATE Class \[► 2016\]](#)[TwinCAT.PlcOpen Namespace \[► 2016\]](#)**6.10.2 DateBase Class**

PlcOpen Date base class (32-Bit)

Inheritance Hierarchy

System.Object




TwinCAT.PlcOpen.DateBase

[TwinCAT.PlcOpen.DATE \[► 2016\]](#)[TwinCAT.PlcOpen.DT \[► 2034\]](#)**Namespace:** [TwinCAT.PlcOpen \[► 2016\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**




```
public abstract class DataBase : IPlcOpenTimeBase<DateTime, uint>,
    IPlcOpenTimeBase
```

The DataBase type exposes the following members.








Constructors

	Name	Description
	DateBase. [▶ 2025]	Initializes a new instance of the DateBase class.
	DateBase(DateTime) [▶ 2027]	Initializes a new instance of the DateBase class.
	DateBase(UInt32) [▶ 2026]	Initializes a new instance of the DateBase class.


Properties

	Name	Description
	IsDate [▶ 2029]	Gets a value indicating whether this instance is representig a date (despite date and time)
 	MarshalSize [▶ 2028]	Gets the marshal size in bytes.
	Ticks [▶ 2028]	Returns the number of ticks that represent the value of this DateBase.

Methods

	Name	Description
	Equals [▶ 2030]	Determines whether the specified Object is equal to this instance. (Overrides Object.Equals(Object).)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode [▶ 2031]	Gets the GetHashCode of the Address (Overrides Object.GetHashCode..)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ParseToTicks [▶ 2031]	Parses the specified PlcOpen Date string to PlcOpen ticks.
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Fields


	Name	Description
	plcTimeSeconds [▶ 2033]	PlcTime Seconds (Ticks)



Reference

[TwinCAT.PlcOpen Namespace](#) [[▶ 2016](#)]

6.10.2.1 DateBase Constructor

Overload List

	Name	Description
	DateBase. [▶ 2025]	Initializes a new instance of the DateBase [▶ 2023] class.

	Name	Description
	DateBase(DateTime) [► 2027]	Initializes a new instance of the DateBase [► 2023] class.
	DateBase(UInt32) [► 2026]	Initializes a new instance of the DateBase [► 2023] class.

Reference

[DateBase Class](#) [\[► 2023\]](#)

[TwinCAT.PlcOpen Namespace](#) [\[► 2016\]](#)

6.10.2.1.1 DateBase Constructor

Initializes a new instance of the [DateBase](#) [\[► 2023\]](#) class.

Namespace: [TwinCAT.PlcOpen](#) [\[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected DateBase()
```

Reference

[DateBase Class](#) [\[► 2023\]](#)

[DateBase Overload](#) [\[► 2024\]](#)

[TwinCAT.PlcOpen Namespace](#) [\[► 2016\]](#)

6.10.2.1.2 DateBase Constructor (DateTimeOffset)

Initializes a new instance of the [DateBase](#) [\[► 2023\]](#) class.

Namespace: [TwinCAT.PlcOpen](#) [\[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
protected DateBase(  
    DateTimeOffset date  
)
```

Parameters

date Type: [System.DateTimeOffset](#)
The date.

Reference

[DateBase Class](#) [\[► 2023\]](#)

[DateBase Overload](#) [\[► 2024\]](#)

[TwinCAT.PlcOpen Namespace](#) [\[► 2016\]](#)

6.10.2.1.3 DateBase Constructor (Int64)

Initializes a new instance of the [DateBase](#) [► 2023] class.

Namespace: [TwinCAT.PlcOpen](#) [► 2016]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
protected DateBase(
    long dateValue
)
```

Parameters

dateValue Type: [System.Int64](#)
The date value in PlcOpen Ticks.

Exceptions

Exception	Condition
ArgumentOutOfRangeException	

Reference

[DateBase Class](#) [► 2023]

[DateBase Overload](#) [► 2024]

[TwinCAT.PlcOpen Namespace](#) [► 2016]

6.10.2.1.4 DateBase Constructor (UInt32)

Initializes a new instance of the [DateBase](#) [► 2023] class.

Namespace: [TwinCAT.PlcOpen](#) [► 2016]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected DateBase(
    uint plcTicks
)
```

Parameters

plcTicks	Type: System.UInt32 The date value in PlcOpen Ticks / Seconds from 1/1190 in Local Time Zone.
----------	--

Exceptions

Exception	Condition
ArgumentOutOfRangeException	

Reference

[DateBase Class \[▸ 2023\]](#)

[DateBase Overload \[▸ 2024\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.2.1.5 DateBase Constructor (DateTime)

Initializes a new instance of the [DateBase \[▸ 2023\]](#) class.

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected DateBase (
    DateTime date
)
```

Parameters

date	Type: System.DateTime The date.
------	------------------------------------

Reference

[DateBase Class \[▸ 2023\]](#)



[DateBase Overload \[▸ 2024\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.2.2 DateBase Properties

The [DateBase \[▸ 2023\]](#) type exposes the following members.

Properties

	Name	Description
	IsDate [▸ 2029]	Gets a value indicating whether this instance is representig a date (despite date and time)
	MarshalSize [▸ 2028]	Gets the marshal size in bytes.
	Ticks [▸ 2028]	Returns the number of ticks that represent the value of this DateBase [▸ 2023] .

Reference

[DateBase Class \[▸ 2023\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.2.2.1 DateBase.Date Property

Gets or sthe date value.

Namespace: [TwinCAT.PlcOpen](#) [[► 2016](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public DateTimeOffset Date { get; }
```

Property Value

Type: [DateTimeOffset](#)
The date.

Reference

[DateBase Class](#) [[► 2023](#)]

[TwinCAT.PlcOpen Namespace](#) [[► 2016](#)]

6.10.2.2 DateBase.MarshalSize Property

Gets the marshal size in bytes.

Namespace: [TwinCAT.PlcOpen](#) [[► 2016](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static int MarshalSize { get; }
```

Property Value

Type: [Int32](#)
Marshal size in bytes.

Reference

[DateBase Class](#) [[► 2023](#)]

[TwinCAT.PlcOpen Namespace](#) [[► 2016](#)]

6.10.2.3 DateBase.Ticks Property

Returns the number of ticks that represent the value of this [DateBase](#) [[► 2023](#)].

Namespace: [TwinCAT.PlcOpen](#) [[► 2016](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint Ticks { get; }
```

Property Value

Type: [UInt32](#)
The ticks.

Implements

[IPlcOpenTimeBase.T1, T2..Ticks \[▸ 2044\]](#)

Reference

[DateBase Class \[▸ 2023\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.2.2.4 DateBase.IsDate Property

Gets a value indicating whether this instance is representig a date (despite date and time)

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected abstract bool IsDate { get; }
```

Property Value

Type: Boolean
true if this instance is date; otherwise, false.

Reference








[DateBase Class \[▸ 2023\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.2.3 DateBase Methods

The [DateBase \[▸ 2023\]](#) type exposes the following members.

Methods

	Name	Description
	Equals [▸ 2030]	Determines whether the specified Object is equal to this instance. (Overrides Object.Equals(Object).)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode [▸ 2031]	Gets the GetHashCode of the Address (Overrides Object.GetHashCode..)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ParseToTicks [▸ 2031]	Parses the specified PlcOpen Date string to PlcOpen ticks.
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[DateBase Class \[▸ 2023\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.2.3.1 DateBase.DateToValue Method

Converts the specified DateTime value to PlcOpen Ticks.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public static long DateToValue(  
    DateTimeOffset date  
)
```

Parameters

date	Type: System.DateTimeOffset The date.
------	--

Return Value

Type: [Int64](#)

Reference

[DateBase Class \[► 2023\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.2.3.2 DateBase.Equals Method

Determines whether the specified Object is equal to this instance.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool Equals(  
    Object? obj  
)
```

Parameters

obj	Type: System.Object The object to compare with the current object.
-----	---

Return Value

Type: [Boolean](#)
true if the specified Object is equal to this instance; otherwise, false.

Reference

[DateBase Class \[► 2023\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.2.3.3 DateBase.GetHashCode Method

Gets the HashCode of the Address

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override int GetHashCode()
```

Return Value

Type: Int32

Reference

[DateBase Class \[► 2023\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.2.3.4 DateBase.ParseToTicks Method

Parses the specified PlcOpen Date string to PlcOpen ticks.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected abstract long ParseToTicks(  
    string s  
)
```

Parameters

s	Type: System.String The s.
---	-------------------------------

Return Value

Type: Int64





Reference

[DateBase Class \[► 2023\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.2.3.5 DateBase.ValueToDate Method

Overload List

	Name	Description
 	ValueToDate(Int64) [► 2032]	Converts the specified PlcOpen dateValue in ticks to a DateTime Object
 	ValueToDate(UInt32) [► 2032]	Converts the specified PlcOpen dateValue in ticks to a DateTime Object

Reference

[DateBase Class \[► 2023\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.2.3.5.1 DateBase.ValueToDate Method (Int64)

Converts the specified PlcOpen dateValue in ticks to a DateTime Object

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public static DateTimeOffset ValueToDate(
    long dateValue
)
```

Parameters

dateValue Type: [System.Int64](#)
The date value.

Return Value

Type: [DateTimeOffset](#)

Reference

[DateBase Class \[► 2023\]](#)

[ValueToDate Overload \[► 2032\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.2.3.5.2 DateBase.ValueToDate Method (UInt32)

Converts the specified PlcOpen dateValue in ticks to a DateTime Object

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public static DateTimeOffset ValueToDate(  
    uint dateValue  
)
```

Parameters

dateValue Type: [System.UInt32](#)
The date value.

Return Value

Type: [DateTimeOffset](#)

Reference

[DateBase Class](#) [[▶ 2023](#)]

[ValueToDate Overload](#) [[▶ 2032](#)]

[TwinCAT.PlcOpen Namespace](#) [[▶ 2016](#)]

6.10.2.4 DateBase Fields

The [DateBase](#) [[▶ 2023](#)] type exposes the following members.

Fields

	Name	Description
	plcTimeSeconds [▶ 2033]	PlcTime Seconds (Ticks)

Reference

[DateBase Class](#) [[▶ 2023](#)]

[TwinCAT.PlcOpen Namespace](#) [[▶ 2016](#)]

6.10.2.4.1 DateBase.plcTimeSeconds Field

PlcTime Seconds (Ticks)

Namespace: [TwinCAT.PlcOpen](#) [[▶ 2016](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected uint plcTimeSeconds
```

Field Value

Type: UInt32

Remarks

Seconds from 1/1/1970 in Local Time Zone (different from UnixTimeSeconds which is UTC!). The is exactly the same value and layout, like it is stored in PlcControl for the DATE and DT type!.

Reference

[DateBase Class \[► 2023\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.3 DT Class

PlcOpen DT (DATE_AND_TIME) datatype.

Inheritance Hierarchy

System.Object

[TwinCAT.PlcOpen.DateBase \[► 2023\]](#)

TwinCAT.PlcOpen.DT

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229






Syntax

C#



```
public sealed class DT : DateBase
```

The DT type exposes the following members.


Constructors






	Name	Description
	DT. [► 2035]	Initializes a new instance of the DT class.
	DT(DateTime) [► 2038]	Initializes a new instance of the DT class.
	DT(DateTimeOffset) [► 2036]	Initializes a new instance of the DT class.
	DT(UInt32) [► 2037]	Initializes a new instance of the DT class.
	DT(Int32, Int32, Int32, Int32, Int32) [► 2037]	Initializes a new instance of the DT class.

Properties

	Name	Description
	DateTime [► 2038]	Gets or the date value (only the Date part of the DateTime)
	Ticks [► 2028]	Returns the number of ticks that represent the value of this DateBase [► 2023] . (Inherited from DateBase [► 2023] .)

Methods

	Name	Description
	Equals [► 2030]	Determines whether the specified Object is equal to this instance. (Inherited from DateBase [► 2023] .)






	Name	Description
	GetHashCode [▶ 2031]	Gets the HashCode of the Address (Inherited from DateBase [▶ 2023] .)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	Parse [▶ 2039]	Parses the specified string to the DT object.
	ToString [▶ 2040]	Returns a String that represents this instance. (Overrides Object.ToString ..)
	TryParse [▶ 2040]	Tries to parse the specified string to a DT object.

Reference

[TwinCAT.PlcOpen Namespace](#) [\[▶ 2016\]](#)

6.10.3.1 DT Constructor

Overload List

	Name	Description
	DT [▶ 2035]	Initializes a new instance of the DT [▶ 2034] class.
	DT(DateTime) [▶ 2038]	Initializes a new instance of the DT [▶ 2034] class.
	DT(DateTimeOffset) [▶ 2036]	Initializes a new instance of the DT [▶ 2034] class.
	DT(UInt32) [▶ 2037]	Initializes a new instance of the DT [▶ 2034] class.
	DT(Int32, Int32, Int32, Int32, Int32) [▶ 2037]	Initializes a new instance of the DT [▶ 2034] class.

Reference

[DT Class](#) [\[▶ 2034\]](#)

[TwinCAT.PlcOpen Namespace](#) [\[▶ 2016\]](#)

6.10.3.1.1 DT Constructor

Initializes a new instance of the [DT](#) [\[▶ 2034\]](#) class.

Namespace: [TwinCAT.PlcOpen](#) [\[▶ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DT()
```

Reference

[DT Class](#) [\[▶ 2034\]](#)

[DT Overload \[► 2035\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.3.1.2 DT Constructor (DateTimeOffset)

Initializes a new instance of the [DT \[► 2034\]](#) class.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DT(  
    DateTimeOffset date  
)
```

Parameters

date	Type: System.DateTimeOffset The date.
------	--

Reference

[DT Class \[► 2034\]](#)

[DT Overload \[► 2035\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.3.1.3 DT Constructor (Int64)

Initializes a new instance of the [DT \[► 2034\]](#) class.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public DT(  
    long dateValue  
)
```

Parameters

dateValue	Type: System.Int64 The date value in PlcOpen Ticks.
-----------	--

Reference

[DT Class \[► 2034\]](#)

[DT Overload \[► 2035\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.3.1.4 DT Constructor (UInt32)

Initializes a new instance of the [DT \[▶ 2034\]](#) class.

Namespace: [TwinCAT.PlcOpen \[▶ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DT(
    uint dateValue
)
```

Parameters

dateValue	Type: System.UInt32 The date value in PlcOpen Ticks.
-----------	---

Reference

[DT Class \[▶ 2034\]](#)

[DT Overload \[▶ 2035\]](#)

[TwinCAT.PlcOpen Namespace \[▶ 2016\]](#)

6.10.3.1.5 DT Constructor (Int32, Int32, Int32, Int32, Int32, Int32)

Initializes a new instance of the [DT \[▶ 2034\]](#) class.

Namespace: [TwinCAT.PlcOpen \[▶ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DT(
    int year,
    int month,
    int day,
    int hour,
    int minute,
    int second
)
```

Parameters

year	Type: System.Int32 The year.
month	Type: System.Int32 The month.
day	Type: System.Int32 The day.
hour	Type: System.Int32 The hour.
minute	Type: System.Int32 The minute.
second	Type: System.Int32 The second.

Reference[DT Class \[► 2034\]](#)[DT Overload \[► 2035\]](#)[TwinCAT.PlcOpen Namespace \[► 2016\]](#)**6.10.3.1.6 DT Constructor (DateTime)**

Initializes a new instance of the [DT \[► 2034\]](#) class.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public DT(
    DateTime date
)
```



Parameters

date	Type: System.DateTime The date.
------	------------------------------------

Reference[DT Class \[► 2034\]](#)[DT Overload \[► 2035\]](#)[TwinCAT.PlcOpen Namespace \[► 2016\]](#)**6.10.3.2 DT Properties**

The [DT \[► 2034\]](#) type exposes the following members.

Properties

	Name	Description
	DateTime [► 2038]	Gets or the date value (only the Date part of the DateTime)
	Ticks [► 2028]	Returns the number of ticks that represent the value of this DateBase [► 2023] . (Inherited from DateBase [► 2023] .)

Reference[DT Class \[► 2034\]](#)[TwinCAT.PlcOpen Namespace \[► 2016\]](#)**6.10.3.2.1 DT.DateTime Property**

Gets or the date value (only the Date part of the DateTime)

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTime DateTime { get; }
```

Property Value

Type: DateTime
The date.

Reference








[DT Class \[► 2034\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.3.3 DT Methods

The [DT \[► 2034\]](#) type exposes the following members.

Methods

	Name	Description
	Equals [► 2030]	Determines whether the specified Object is equal to this instance. (Inherited from DateBase [► 2023] .)
	GetHashCode [► 2031]	Gets the GetHashCode of the Address (Inherited from DateBase [► 2023] .)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	Parse [► 2039]	Parses the specified string to the DT [► 2034] object.
	ToString [► 2040]	Returns a String that represents this instance. (Overrides Object.ToString..)
	TryParse [► 2040]	Tries to parse the specified string to a DT [► 2034] object.
		

Reference

[DT Class \[► 2034\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.3.3.1 DT.Parse Method

Parses the specified string to the [DT \[► 2034\]](#) object.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static DT Parse(
    string s
)
```

Parameters

s	Type: System.String The s.
---	-------------------------------

Return Value

Type: [DT \[▸ 2034\]](#)
DT.

Exceptions

Exception	Condition
FormatException	Cannot parse DT object!

Reference

[DT Class \[▸ 2034\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.3.3.2 DT.ToString Method

Returns a String that represents this instance.

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public override string ToString()
```

Return Value

Type: String
A String that represents this instance.

Reference

[DT Class \[▸ 2034\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.3.3.3 DT.TryParse Method

Tries to parse the specified string to a [DT \[▸ 2034\]](#) object.

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static bool TryParse(
    string s,
    out DT?? dt
)
```


Parameters

s	Type: System.String The s.
dt	Type: TwinCAT.PlcOpen.DT [▶ 2034]. The dt.

Return Value

Type: Boolean
true if the string could be parsed to [DT \[\[▶ 2034\]\(#\)\]](#), false otherwise.

Reference

[DT Class \[\[▶ 2034\]\(#\)\]](#)

[TwinCAT.PlcOpen Namespace \[\[▶ 2016\]\(#\)\]](#)

6.10.4 IPlcOpenTimeBase Interface

Interface IPlcOpenType

Namespace: [TwinCAT.PlcOpen \[\[▶ 2016\]\(#\)\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229




Syntax

C#

```
public interface IPlcOpenTimeBase
```

The IPlcOpenTimeBase type exposes the following members.

Properties

	Name	Description
	ManagedValueType [▶ 2042]	Gets the type of the underlying human readable type (DateTime or Timespan)
	TicksValueType [▶ 2042]	Gets the type of the underlying ticks resolution (uint32 or uint64)
	UntypedValue [▶ 2042]	Returns the 'Value' as object type.



Reference


[TwinCAT.PlcOpen Namespace \[\[▶ 2016\]\(#\)\]](#)

6.10.4.1 IPlcOpenTimeBase Properties

The [IPlcOpenTimeBase \[\[▶ 2041\]\(#\)\]](#) type exposes the following members.

Properties

	Name	Description
	ManagedValueType [▶ 2042]	Gets the type of the underlying human readable type (DateTime or Timespan)
	TicksValueType [▶ 2042]	Gets the type of the underlying ticks resolution (uint32 or uint64)

	Name	Description
	UntypedValue [▶ 2042]	Returns the 'Value' as object type.

Reference

[IPlcOpenTimeBase Interface](#) [▶ 2041]

[TwinCAT.PlcOpen Namespace](#) [▶ 2016]

6.10.4.1.1 IPlcOpenTimeBase.ManagedValueType Property

Gets the type of the underlying human readable type (DateTime or Timespan)

Namespace: [TwinCAT.PlcOpen](#) [▶ 2016]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Type ManagedValueType { get; }
```

Property Value

Type: Type

The type of the managed value.

Reference

[IPlcOpenTimeBase Interface](#) [▶ 2041]

[TwinCAT.PlcOpen Namespace](#) [▶ 2016]

6.10.4.1.2 IPlcOpenTimeBase.TicksValueType Property

Gets the type of the underlying ticks resolution (uint32 or uint64)

Namespace: [TwinCAT.PlcOpen](#) [▶ 2016]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Type TicksValueType { get; }
```

Property Value

Type: Type

The type of the ticks value.

Reference

[IPlcOpenTimeBase Interface](#) [▶ 2041]

[TwinCAT.PlcOpen Namespace](#) [▶ 2016]

6.10.4.1.3 IPlcOpenTimeBase.UntypedValue Property

Returns the 'Value' as object type.

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object UntypedValue { get; }
```

Property Value

Type: Object
The untyped value.

Reference

[IPlcOpenTimeBase Interface \[▸ 2041\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.5 IPlcOpenTimeBase.T1, T2. Interface

Interface IPlcOpenType

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#






```
public interface IPlcOpenTimeBase<T1, T2> : IPlcOpenTimeBase
```

Type Parameters

T1	The type of the t1.
T2	The type of the t2.

The IPlcOpenTimeBase.T1, T2. type exposes the following members.

Properties

	Name	Description
	ManagedValueType [▸ 2042]	Gets the type of the underlying human readable type (DateTime or Timespan) (Inherited from IPlcOpenTimeBase [▸ 2041] .)
	Ticks [▸ 2044]	Returns the number of ticks that represent the value of this IPlcOpenTimeBase.T1, T2. (uint32 or uint64).
	TicksValueType [▸ 2042]	Gets the type of the underlying ticks resolution (uint32 or uint64) (Inherited from IPlcOpenTimeBase [▸ 2041] .)
	UntypedValue [▸ 2042]	Returns the 'Value' as object type. (Inherited from IPlcOpenTimeBase [▸ 2041] .)
	Value [▸ 2044]	Returns the value of this IPlcOpenTimeBase.T1, T2. as Managed base type (DateTime or Timespan)

Reference






[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

[TwinCAT.PlcOpen.IPlcOpenTimeBase \[▸ 2041\]](#)

6.10.5.1 IPlcOpenTimeBase.T1, T2. Properties

The [IPlcOpenTimeBase.T1, T2. \[▸ 2043\]](#) generic type exposes the following members.

Properties

	Name	Description
	ManagedValueType [▸ 2042]	Gets the type of the underlying human readable type (DateTime or Timespan) (Inherited from IPlcOpenTimeBase [▸ 2041].)
	Ticks [▸ 2044]	Returns the number of ticks that represent the value of this IPlcOpenTimeBase.T1, T2. [▸ 2043] (uint32 or uint64).
	TicksValueType [▸ 2042]	Gets the type of the underlying ticks resolution (uint32 or uint64) (Inherited from IPlcOpenTimeBase [▸ 2041].)
	UntypedValue [▸ 2042]	Returns the 'Value' as object type. (Inherited from IPlcOpenTimeBase [▸ 2041].)
	Value [▸ 2044]	Returns the value of this IPlcOpenTimeBase.T1, T2. [▸ 2043] as Managed base type (DateTime or Timespan)

Reference

[IPlcOpenTimeBase.T1, T2. Interface \[▸ 2043\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.5.1.1 IPlcOpenTimeBase.T1, T2..Ticks Property

Returns the number of ticks that represent the value of this [IPlcOpenTimeBase.T1, T2. \[▸ 2043\]](#) (uint32 or uint64).

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
T2 Ticks { get; }
```

Property Value

Type: [T2 \[▸ 2043\]](#)

The ticks.

Reference

[IPlcOpenTimeBase.T1, T2. Interface \[▸ 2043\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.5.1.2 IPlcOpenTimeBase.T1, T2..Value Property

Returns the value of this [IPlcOpenTimeBase.T1, T2. \[▸ 2043\]](#) as Managed base type (DateTime or Timespan)

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
T1 Value { get; }
```

Property Value

Type: [T1](#) [[▶ 2043](#)]
 The value.

Reference

[IPlcOpenTimeBase.T1, T2. Interface](#) [[▶ 2043](#)]

[TwinCAT.PlcOpen Namespace](#) [[▶ 2016](#)]

6.10.6 LTIME Class

PlcOpen LTIME class

Inheritance Hierarchy

System.Object
[TwinCAT.PlcOpen.LTimeBase](#) [[▶ 2053](#)]
 TwinCAT.PlcOpen.LTIME

Namespace: [TwinCAT.PlcOpen](#) [[▶ 2016](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229








Syntax

C#



```
public sealed class LTIME : LTimeBase
```

The LTIME type exposes the following members.







Constructors

	Name	Description
	LTIME . [▶ 2047]	Initializes a new instance of the TIME [▶ 2061] class.
	LTIME(Int64) [▶ 2047]	Initializes a new instance of the TIME [▶ 2061] class.
	LTIME(TimeSpan) [▶ 2047]	Initializes a new instance of the TIME [▶ 2061] class.
	LTIME(UInt64) [▶ 2048]	Initializes a new instance of the TIME [▶ 2061] class.
	LTIME(Int32, Int32, Int32) [▶ 2048]	Initializes a new instance of the LTIME class.
	LTIME(Int32, Int32, Int32, Int32) [▶ 2049]	Initializes a new instance of the LTIME class.
	LTIME(Int32, Int32, Int32, Int32, Int32, Int32) [▶ 2050]	Initializes a new instance of the LTIME class.

Properties

	Name	Description
	Ticks [▶ 2056]	Returns the number of ticks that represent the value of this LTimeBase [▶ 2053]. (Inherited from LTimeBase [▶ 2053].)
	Time [▶ 2056]	Gets or the time value. (Inherited from LTimeBase [▶ 2053].)

Methods

	Name	Description
	Equals [▶ 2057]	Determines whether the specified Object is equal to this instance. (Inherited from LTimeBase [▶ 2053].)
	GetHashCode [▶ 2058]	Gets the GetHashCode of the Address (Inherited from LTimeBase [▶ 2053].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	Parse [▶ 2051]	Parses the specified string to a LTIME object.
	ToString [▶ 2052]	Returns a String that represents this instance. (Overrides Object.ToString..)
	TryParse [▶ 2052]	Tries to parse the string to a LTIME object.








Remarks

This class has an internal representation of an UINT64 (8 Bytes)

Reference

[TwinCAT.PlcOpen Namespace](#) [[▶ 2016](#)]

6.10.6.1 LTIME Constructor**Overload List**

	Name	Description
	LTIME. [▶ 2047]	Initializes a new instance of the TIME [▶ 2061] class.
	LTIME(Int64) [▶ 2047]	Initializes a new instance of the TIME [▶ 2061] class.
	LTIME(TimeSpan) [▶ 2047]	Initializes a new instance of the TIME [▶ 2061] class.
	LTIME(UInt64) [▶ 2048]	Initializes a new instance of the TIME [▶ 2061] class.
	LTIME(Int32, Int32, Int32) [▶ 2048]	Initializes a new instance of the LTIME [▶ 2045] class.
	LTIME(Int32, Int32, Int32, Int32) [▶ 2049]	Initializes a new instance of the LTIME [▶ 2045] class.
	LTIME(Int32, Int32, Int32, Int32, Int32, Int32) [▶ 2050]	Initializes a new instance of the LTIME [▶ 2045] class.

Reference

[LTIME Class \[► 2045\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.6.1.1 LTIME Constructor

Initializes a new instance of the [TIME \[► 2061\]](#) class.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public LTIME()
```

Reference

[LTIME Class \[► 2045\]](#)

[LTIME Overload \[► 2046\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.6.1.2 LTIME Constructor (Int64)

Initializes a new instance of the [TIME \[► 2061\]](#) class.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public LTIME(  
    long timeValue  
)
```

Parameters

timeValue	Type: System.Int64 The time value.
-----------	---------------------------------------

Reference

[LTIME Class \[► 2045\]](#)

[LTIME Overload \[► 2046\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.6.1.3 LTIME Constructor (TimeSpan)

Initializes a new instance of the [TIME \[► 2061\]](#) class.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public LTIME(
    TimeSpan time
)
```

Parameters

time	Type: System.TimeSpan The time.
------	------------------------------------

Reference

[LTIME Class \[► 2045\]](#)

[LTIME Overload \[► 2046\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.6.1.4 LTIME Constructor (UInt64)

Initializes a new instance of the [LTIME \[► 2061\]](#) class.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public LTIME(
    ulong timeValue
)
```

Parameters

timeValue	Type: System.UInt64 The time value.
-----------	--

Reference

[LTIME Class \[► 2045\]](#)

[LTIME Overload \[► 2046\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.6.1.5 LTIME Constructor (Int32, Int32, Int32)

Initializes a new instance of the [LTIME \[► 2045\]](#) class.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public LTIME(
    int seconds,
    int milliseconds,
    int microseconds
)
```

Parameters

seconds	Type: System.Int32 The seconds.
milliseconds	Type: System.Int32 The milliseconds.
microseconds	Type: System.Int32 The microseconds.

Reference

[LTIME Class \[► 2045\]](#)

[LTIME Overload \[► 2046\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.6.1.6 LTIME Constructor (Int32, Int32, Int32, Int32)

Initializes a new instance of the [LTIME \[► 2045\]](#) class.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public LTIME(
    int seconds,
    int milliseconds,
    int microseconds,
    int nanoseconds
)
```

Parameters

seconds	Type: System.Int32 The seconds.
milliseconds	Type: System.Int32 The milliseconds.
microseconds	Type: System.Int32 The microseconds.
nanoseconds	Type: System.Int32 The nanoseconds.

Reference

[LTIME Class \[► 2045\]](#)

[LTIME Overload \[► 2046\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.6.1.7 LTIME Constructor (Int32, Int32, Int32, Int32, Int32, Int32, Int32)

Initializes a new instance of the [LTIME](#) [[▶ 2045](#)] class.

Namespace: [TwinCAT.PlcOpen](#) [[▶ 2016](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public LTIME(
    int days,
    int hours,
    int minutes,
    int seconds,
    int milliseconds,
    int microseconds,
    int nanoseconds
)
```

Parameters

days	Type: System.Int32 The days.
hours	Type: System.Int32 The hours.
minutes	Type: System.Int32 The minutes.
seconds	Type: System.Int32 The seconds.
milliseconds	Type: System.Int32 The milliseconds.
microseconds	Type: System.Int32 The microseconds.
nanoseconds	Type: System.Int32 The nanoseconds.

Reference

[LTIME Class](#) [[▶ 2045](#)]



[LTIME Overload](#) [[▶ 2046](#)]

[TwinCAT.PlcOpen Namespace](#) [[▶ 2016](#)]

6.10.6.2 LTIME Properties

The [LTIME](#) [[▶ 2045](#)] type exposes the following members.

Properties

	Name	Description
	Ticks [▶ 2056]	Returns the number of ticks that represent the value of this LTimeBase [▶ 2053]. (Inherited from LTimeBase [▶ 2053].)
	Time [▶ 2056]	Gets or the time value. (Inherited from LTimeBase [▶ 2053].)

Reference









[LTIME Class](#) [[▶ 2045](#)]

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.6.3 LTIME Methods

The [LTIME \[▸ 2045\]](#) type exposes the following members.

Methods

	Name	Description
	Equals [▸ 2057]	Determines whether the specified Object is equal to this instance. (Inherited from LTimeBase [▸ 2053] .)
	GetHashCode [▸ 2058]	Gets the GetHashCode of the Address (Inherited from LTimeBase [▸ 2053] .)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	Parse [▸ 2051]	Parses the specified string to a LTIME [▸ 2045] object.
		
	ToString [▸ 2052]	Returns a String that represents this instance. (Overrides Object.ToString..)
	TryParse [▸ 2052]	Tries to parse the string to a LTIME [▸ 2045] object.
		

Reference

[LTIME Class \[▸ 2045\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.6.3.1 LTIME.Parse Method

Parses the specified string to a [LTIME \[▸ 2045\]](#) object.

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static LTIME Parse(
    string str
)
```

Parameters

str	Type: System.String The string.
-----	------------------------------------

Return Value

Type: [LTIME \[▸ 2045\]](#)
LTIME.

Exceptions

Exception	Condition
FormatException	Cannot create TIME DataType!

Reference[LTIME Class \[► 2045\]](#)[TwinCAT.PlcOpen Namespace \[► 2016\]](#)**6.10.6.3.2 LTIME.ToString Method**

Returns a String that represents this instance.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public override string ToString()
```

Return Value

Type: String

A String that represents this instance.

Reference[LTIME Class \[► 2045\]](#)[TwinCAT.PlcOpen Namespace \[► 2016\]](#)**6.10.6.3.3 LTIME.TryParse Method**

Tries to parse the string to a [LTIME \[► 2045\]](#) object.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static bool TryParse(
    string str,
    out LTIME?? ret
)
```

Parameters

str	Type: System.String The string.
ret	Type: TwinCAT.PlcOpen.LTIME [► 2045] . The ret.

Return Value

Type: Boolean

true if the [LTIME \[► 2045\]](#) object could be parsed, false otherwise.**Reference**[LTIME Class \[► 2045\]](#)[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.7 LTimeBase Class

Time base class

Inheritance Hierarchy

System.Object
 TwinCAT.PlcOpen.LTimeBase
 TwinCAT.PlcOpen.LTIME [|> 2045](#)

Namespace: [TwinCAT.PlcOpen |> 2016](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax

C#





```
public abstract class LTimeBase : IPlcOpenTimeBase<TimeSpan, ulong>,
    IPlcOpenTimeBase
```

The LTimeBase type exposes the following members.









Constructors





	Name	Description
	LTimeBase. > 2054	Initializes a new instance of the TimeBase > 2068 class.
	LTimeBase(UInt64) > 2054	Initializes a new instance of the TimeBase > 2068 class.

Properties


	Name	Description
	MarshalSize > 2055	Gets the marshal size in bytes.
		
	Ticks > 2056	Returns the number of ticks that represent the value of this LTimeBase.
	Time > 2056	Gets or the time value.

Methods

	Name	Description
	Equals > 2057	Determines whether the specified Object is equal to this instance. (Overrides Object.Equals(Object).)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode > 2058	Gets the GetHashCode of the Address (Overrides Object.GetHashCode..)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	TimeToValue > 2058	Converts the Timespan to PlcOpen ticks.
		
	ToString	Returns a string that represents the current object. (Inherited from Object.)

	Name	Description
 	ValueToTime(Int64) [▶ 2059]	Converts the timeValue (PlcOpen ticks) to TimeSpan
 	ValueToTime(UInt64) [▶ 2060]	Converts the timeValue (PlcOpen ticks) to TimeSpan

Fields





	Name	Description
	internalTimeValue [▶ 2060]	The internal time value

Reference

[TwinCAT.PlcOpen Namespace](#) [▶ 2016]

6.10.7.1 LTimeBase Constructor

Overload List

	Name	Description
 	LTimeBase. [▶ 2054]	Initializes a new instance of the TimeBase [▶ 2068] class.
 	LTimeBase(UInt64) [▶ 2054]	Initializes a new instance of the TimeBase [▶ 2068] class.

Reference

[LTimeBase Class](#) [▶ 2053]

[TwinCAT.PlcOpen Namespace](#) [▶ 2016]

6.10.7.1.1 LTimeBase Constructor

Initializes a new instance of the [TimeBase](#) [▶ 2068] class.

Namespace: [TwinCAT.PlcOpen](#) [▶ 2016]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected LTimeBase()
```

Reference

[LTimeBase Class](#) [▶ 2053]

[LTimeBase Overload](#) [▶ 2054]

[TwinCAT.PlcOpen Namespace](#) [▶ 2016]

6.10.7.1.2 LTimeBase Constructor (UInt64)

Initializes a new instance of the [TimeBase](#) [▶ 2068] class.

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected LTimeBase(
    ulong timeValue
)
```

Parameters

timeValue	Type: System.UInt64 The time value.
-----------	--

Exceptions

Exception	Condition
ArgumentOutOfRangeException	

Reference

[LTimeBase Class \[▸ 2053\]](#)





[LTimeBase Overload \[▸ 2054\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.7.2 LTimeBase Properties

The [LTimeBase \[▸ 2053\]](#) type exposes the following members.

Properties

	Name	Description
	MarshalSize [▸ 2055]	Gets the marshal size in bytes.
		
	Ticks [▸ 2056]	Returns the number of ticks that represent the value of this LTimeBase [▸ 2053] .
	Time [▸ 2056]	Gets or the time value.

Reference

[LTimeBase Class \[▸ 2053\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.7.2.1 LTimeBase.MarshalSize Property

Gets the marshal size in bytes.

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static int MarshalSize { get; }
```

Property Value

Type: Int32
Marshal size in bytes.

Reference

[LTimeBase Class](#) [[▶ 2053](#)]

[TwinCAT.PlcOpen Namespace](#) [[▶ 2016](#)]

6.10.7.2.2 LTimeBase.Ticks Property

Returns the number of ticks that represent the value of this [LTimeBase](#) [[▶ 2053](#)].

Namespace: [TwinCAT.PlcOpen](#) [[▶ 2016](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ulong Ticks { get; }
```

Property Value

Type: UInt64
The ticks.

Implements

[IPlcOpenTimeBase.T1, T2..Ticks](#) [[▶ 2044](#)]

Reference

[LTimeBase Class](#) [[▶ 2053](#)]

[TwinCAT.PlcOpen Namespace](#) [[▶ 2016](#)]

6.10.7.2.3 LTimeBase.Time Property

Gets or the time value.

Namespace: [TwinCAT.PlcOpen](#) [[▶ 2016](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public TimeSpan Time { get; }
```

Property Value

Type: TimeSpan
The time.

Reference













[LTimeBase Class \[▸ 2053\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.7.3 LTimeBase Methods

The [LTimeBase \[▸ 2053\]](#) type exposes the following members.

Methods

	Name	Description
	Equals [▸ 2057]	Determines whether the specified Object is equal to this instance. (Overrides Object.Equals(Object).)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode [▸ 2058]	Gets the GetHashCode of the Address (Overrides Object.GetHashCode..)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
 	TimeToValue [▸ 2058]	Converts the Timespan to PlcOpen ticks.
	ToString	Returns a string that represents the current object. (Inherited from Object.)
 	ValueToTime(Int64) [▸ 2059]	Converts the timeValue (PlcOpen ticks) to TimeSpan
 	ValueToTime(UInt64) [▸ 2060]	Converts the timeValue (PlcOpen ticks) to TimeSpan

Reference

[LTimeBase Class \[▸ 2053\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.7.3.1 LTimeBase.Equals Method

Determines whether the specified Object is equal to this instance.

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool Equals(
    Object? obj
)
```

Parameters

obj	Type: System.Object The object to compare with the current object.
-----	---

Return Value

Type: Boolean

true if the specified Object is equal to this instance; otherwise, false.

Reference

[LTimeBase Class \[► 2053\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.7.3.2 LTimeBase.GetHashCode Method

Gets the HashCode of the Address

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override int GetHashCode()
```

Return Value

Type: Int32

Reference

[LTimeBase Class \[► 2053\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.7.3.3 LTimeBase.TimeToValue Method

Converts the Timespan to PlcOpen ticks.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ulong TimeToValue(  
    TimeSpan time  
)
```

Parameters

time	Type: System.TimeSpan The time.
------	------------------------------------

Return Value

Type: UInt64





Reference

[LTimeBase Class \[▸ 2053\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.7.3.4 LTimeBase.ValueToTime Method

Overload List

	Name	Description
 	ValueToTime(Int64) [▸ 2059]	Converts the timeValue (PlcOpen ticks) to TimeSpan
 	ValueToTime(UInt64) [▸ 2060]	Converts the timeValue (PlcOpen ticks) to TimeSpan

Reference

[LTimeBase Class \[▸ 2053\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.7.3.4.1 LTimeBase.ValueToTime Method (Int64)

Converts the timeValue (PlcOpen ticks) to TimeSpan

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static TimeSpan ValueToTime(
    long nanoseconds
)
```

Parameters

nanoseconds	Type: System.Int64 The time value.
-------------	---------------------------------------

Return Value

Type: TimeSpan

Reference

[LTimeBase Class \[▸ 2053\]](#)

[ValueToTime Overload \[▸ 2059\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.7.3.4.2 LTimeBase.ValueToTime Method (UInt64)

Converts the timeValue (PlcOpen ticks) to TimeSpan

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static TimeSpan ValueToTime(
    ulong nanoseconds
)
```

Parameters

nanoseconds	Type: System.UInt64 The time value.
-------------	--

Return Value

Type: TimeSpan

Reference

[LTimeBase Class \[▸ 2053\]](#)


[ValueToTime Overload \[▸ 2059\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.7.4 LTimeBase Fields

The [LTimeBase \[▸ 2053\]](#) type exposes the following members.

Fields

	Name	Description
	internalTimeValue [▸ 2060]	The internal time value

Reference

[LTimeBase Class \[▸ 2053\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.7.4.1 LTimeBase.internalTimeValue Field

The internal time value

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected ulong internalTimeValue
```

Field Value

Type: UInt64

Reference

[LTimeBase Class \[▶ 2053\]](#)

[TwinCAT.PlcOpen Namespace \[▶ 2016\]](#)

6.10.8 TIME Class

PlcOpen TIME class

Inheritance Hierarchy

System.Object

[TwinCAT.PlcOpen.TimeBase \[▶ 2068\]](#)

TwinCAT.PlcOpen.TIME

Namespace: [TwinCAT.PlcOpen \[▶ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229







Syntax

C#




```
public sealed class TIME : TimeBase
```

The TIME type exposes the following members.









Constructors

	Name	Description
	TIME. [▶ 2062]	Initializes a new instance of the TIME class.
	TIME(UInt64) [▶ 2063]	Initializes a new instance of the TIME class.
	TIME(TimeSpan) [▶ 2063]	Initializes a new instance of the TIME class.
	TIME(UInt32) [▶ 2064]	Initializes a new instance of the TIME class.
	TIME(Int32, Int32) [▶ 2064]	Initializes a new instance of the TIME class.
	TIME(Int32, Int32, Int32, Int32, Int32, Int32) [▶ 2065]	Initializes a new instance of the TIME class.

Properties

	Name	Description
	InternalTimeValue [▶ 2071]	Gets the internal time value. (Inherited from TimeBase [▶ 2068].)
	Ticks [▶ 2072]	Returns the number of ticks that represent the value of this TimeBase [▶ 2068] (uint32 or uint64). (Inherited from TimeBase [▶ 2068].)
	Time [▶ 2072]	Gets the time value. (Inherited from TimeBase [▶ 2068].)

Methods

	Name	Description
	Equals [▸ 2073]	Determines whether the specified Object is equal to this instance. (Inherited from TimeBase [▸ 2068].)
	GetHashCode [▸ 2074]	Gets the GetHashCode of the Address (Inherited from TimeBase [▸ 2068].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	Parse [▸ 2066]	Parses the specified string to a TIME object.
		
	ToString [▸ 2067]	Returns a String that represents this instance. (Overrides Object.ToString ..)
	TryParse [▸ 2067]	Tries to parse the TIME object from string.
		







Remarks

This class has an internal representation of an UINT32 (4 bytes).

Reference

[TwinCAT.PlcOpen Namespace](#) [[▸ 2016](#)]

6.10.8.1 TIME Constructor**Overload List**

	Name	Description
	TIME . [▸ 2062]	Initializes a new instance of the TIME [▸ 2061] class.
	TIME(Int64) [▸ 2063]	Initializes a new instance of the TIME [▸ 2061] class.
	TIME(TimeSpan) [▸ 2063]	Initializes a new instance of the TIME [▸ 2061] class.
	TIME(UInt32) [▸ 2064]	Initializes a new instance of the TIME [▸ 2061] class.
	TIME(Int32, Int32) [▸ 2064]	Initializes a new instance of the TIME [▸ 2061] class.
	TIME(Int32, Int32, Int32, Int32, Int32) [▸ 2065]	Initializes a new instance of the TIME [▸ 2061] class.

Reference

[TIME Class](#) [[▸ 2061](#)]

[TwinCAT.PlcOpen Namespace](#) [[▸ 2016](#)]

6.10.8.1.1 TIME Constructor

Initializes a new instance of the [TIME](#) [[▸ 2061](#)] class.

Namespace: [TwinCAT.PlcOpen](#) [[▸ 2016](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public TIME()
```

Reference

[TIME Class \[▸ 2061\]](#)

[TIME Overload \[▸ 2062\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.8.1.2 TIME Constructor (Int64)

Initializes a new instance of the [TIME \[▸ 2061\]](#) class.

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public TIME(  
    long timeValue  
)
```

Parameters

timeValue	Type: System.Int64 The time value.
-----------	---------------------------------------

Reference

[TIME Class \[▸ 2061\]](#)

[TIME Overload \[▸ 2062\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.8.1.3 TIME Constructor (TimeSpan)

Initializes a new instance of the [TIME \[▸ 2061\]](#) class.

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public TIME(  
    TimeSpan time  
)
```

Parameters

time	Type: System.TimeSpan The time.
------	------------------------------------

Reference[TIME Class \[► 2061\]](#)[TIME Overload \[► 2062\]](#)[TwinCAT.PlcOpen Namespace \[► 2016\]](#)**6.10.8.1.4 TIME Constructor (UInt32)**Initializes a new instance of the [TIME \[► 2061\]](#) class.**Namespace:** [TwinCAT.PlcOpen \[► 2016\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public TIME(
    uint timeValue
)
```

Parameters

timeValue	Type: System.UInt32 The time value.
-----------	--

Reference[TIME Class \[► 2061\]](#)[TIME Overload \[► 2062\]](#)[TwinCAT.PlcOpen Namespace \[► 2016\]](#)**6.10.8.1.5 TIME Constructor (Int32, Int32)**Initializes a new instance of the [TIME \[► 2061\]](#) class.**Namespace:** [TwinCAT.PlcOpen \[► 2016\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public TIME(
    int seconds,
    int milliseconds
)
```

Parameters

seconds	Type: System.Int32 The seconds.
milliseconds	Type: System.Int32 The milliseconds.

Reference[TIME Class \[► 2061\]](#)

[TIME Overload \[▸ 2062\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.8.1.6 TIME Constructor (Int32, Int32, Int32, Int32, Int32)

Initializes a new instance of the [TIME \[▸ 2061\]](#) class.

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public TIME(
    int days,
    int hours,
    int minutes,
    int seconds,
    int milliseconds
)
```

Parameters

days	Type: System.Int32 The days.
hours	Type: System.Int32 The hours.
minutes	Type: System.Int32 The minutes.
seconds	Type: System.Int32 The seconds.
milliseconds	Type: System.Int32 The milliseconds.

Reference

[TIME Class \[▸ 2061\]](#)




[TIME Overload \[▸ 2062\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.8.2 TIME Properties

The [TIME \[▸ 2061\]](#) type exposes the following members.








Properties

	Name	Description
	InternalTimeValue [▸ 2071]	Gets the internal time value. (Inherited from TimeBase [▸ 2068].)
	Ticks [▸ 2072]	Returns the number of ticks that represent the value of this TimeBase [▸ 2068] (uint32 or uint64). (Inherited from TimeBase [▸ 2068].)
	Time [▸ 2072]	Gets the time value. (Inherited from TimeBase [▸ 2068].)

Reference[TIME Class \[▸ 2061\]](#)[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)**6.10.8.3 TIME Methods**

The [TIME \[▸ 2061\]](#) type exposes the following members.

Methods

	Name	Description
	Equals [▸ 2073]	Determines whether the specified Object is equal to this instance. (Inherited from TimeBase [▸ 2068] .)
	GetHashCode [▸ 2074]	Gets the GetHashCode of the Address (Inherited from TimeBase [▸ 2068] .)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	Parse [▸ 2066]	Parses the specified string to a TIME [▸ 2061] object.
	ToString [▸ 2067]	Returns a String that represents this instance. (Overrides Object.ToString..)
	TryParse [▸ 2067]	Tries to parse the TIME [▸ 2061] object from string.
		

Reference[TIME Class \[▸ 2061\]](#)[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)**6.10.8.3.1 TIME.Parse Method**

Parses the specified string to a [TIME \[▸ 2061\]](#) object.

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static TIME Parse(
    string str
)
```

Parameters

str	Type: System.String The string.
-----	------------------------------------

Return Value

Type: [TIME \[▸ 2061\]](#)

TIME.

Exceptions

Exception	Condition
FormatException	Cannot create TIME DataType!

Reference

[TIME Class \[► 2061\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.8.3.2 TIME.ToString Method

Returns a String that represents this instance.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public override string ToString()
```

Return Value

Type: String

A String that represents this instance.

Reference

[TIME Class \[► 2061\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.8.3.3 TIME.TryParse Method

Tries to parse the [TIME \[► 2061\]](#) object from string.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static bool TryParse(
    string str,
    out TIME?? ret
)
```

Parameters

str	Type: System.String The string.
ret	Type: TwinCAT.PlcOpen.TIME [► 2061] . The ret.

Return Value

Type: Boolean

true if the [TIME \[► 2061\]](#) object could be parsed, false otherwise.

Reference

[TIME Class \[▸ 2061\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.9 TimeBase Class

Base class for PlcOpen Time types.

Inheritance Hierarchy

System.Object

TwinCAT.PlcOpen.TimeBase

[TwinCAT.PlcOpen.TIME \[▸ 2061\]](#)

[TwinCAT.PlcOpen.TOD \[▸ 2077\]](#)

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229




Syntax

C#






```
public abstract class TimeBase : IPlcOpenTimeBase<TimeSpan, uint>,
    IPlcOpenTimeBase
```

The TimeBase type exposes the following members.


Constructors












	Name	Description
	TimeBase. [▸ 2069]	Initializes a new instance of the TimeBase class.
	TimeBase(Int64) [▸ 2070]	Initializes a new instance of the TimeBase class.
	TimeBase(UInt32) [▸ 2070]	Initializes a new instance of the TimeBase class.

Properties


	Name	Description
	InternalTimeValue [▸ 2071]	Gets the internal time value.
	MarshalSize [▸ 2072]	Gets the marshal size in bytes.
		
	Ticks [▸ 2072]	Returns the number of ticks that represent the value of this TimeBase (uint32 or uint64).
	Time [▸ 2072]	Gets the time value.

Methods

	Name	Description
	Equals [▸ 2073]	Determines whether the specified Object is equal to this instance. (Overrides Object.Equals(Object).)

	Name	Description
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	<u>GetHashCode</u> [▶ 2074]	Gets the GetHashCode of the Address (Overrides Object.GetHashCode..)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	<u>TimeToValue</u> [▶ 2074]	Converts the Timespan to PlcOpen ticks.
		
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	<u>ValueToTime(Int64)</u> [▶ 2075]	Converts the timeValue (PlcOpen ticks) to TimeSpan
		
	<u>ValueToTime(UInt32)</u> [▶ 2076]	Converts the timeValue (PlcOpen ticks) to TimeSpan
		

Fields




	Name	Description
	<u>internalTimeValue</u> [▶ 2077]	The internal time value

Reference

[TwinCAT.PlcOpen Namespace \[▶ 2016\]](#)

6.10.9.1 TimeBase Constructor

Overload List

	Name	Description
	<u>TimeBase.</u> [▶ 2069]	Initializes a new instance of the TimeBase [▶ 2068] class.
	<u>TimeBase(Int64)</u> [▶ 2070]	Initializes a new instance of the TimeBase [▶ 2068] class.
	<u>TimeBase(UInt32)</u> [▶ 2070]	Initializes a new instance of the TimeBase [▶ 2068] class.

Reference

[TimeBase Class \[▶ 2068\]](#)

[TwinCAT.PlcOpen Namespace \[▶ 2016\]](#)

6.10.9.1.1 TimeBase Constructor

Initializes a new instance of the [TimeBase \[▶ 2068\]](#) class.

Namespace: [TwinCAT.PlcOpen \[▶ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected TimeBase()
```

Reference

[TimeBase Class \[► 2068\]](#)

[TimeBase Overload \[► 2069\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.9.1.2 TimeBase Constructor (Int64)

Initializes a new instance of the [TimeBase \[► 2068\]](#) class.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected TimeBase(
    long timeValue
)
```

Parameters

timeValue	Type: System.Int64 The time value.
-----------	---------------------------------------

Exceptions

Exception	Condition
ArgumentOutOfRangeException	

Reference

[TimeBase Class \[► 2068\]](#)

[TimeBase Overload \[► 2069\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.9.1.3 TimeBase Constructor (UInt32)

Initializes a new instance of the [TimeBase \[► 2068\]](#) class.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected TimeBase(
    uint timeValue
)
```

Parameters

timeValue	Type: System.UInt32 The time value.
-----------	--

Exceptions

Exception	Condition
ArgumentOutOfRangeException	

Reference

[TimeBase Class \[▸ 2068\]](#)






[TimeBase Overload \[▸ 2069\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.9.2 TimeBase Properties

The [TimeBase \[▸ 2068\]](#) type exposes the following members.

Properties

	Name	Description
	InternalTimeValue [▸ 2071]	Gets the internal time value.
 	MarshalSize [▸ 2072]	Gets the marshal size in bytes.
	Ticks [▸ 2072]	Returns the number of ticks that represent the value of this TimeBase [▸ 2068] (uint32 or uint64).
	Time [▸ 2072]	Gets the time value.

Reference

[TimeBase Class \[▸ 2068\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.9.2.1 TimeBase.InternalTimeValue Property

Gets the internal time value.

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint InternalTimeValue { get; }
```

Property Value

Type: UInt32
The internal time value.

Reference

[TimeBase Class \[► 2068\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.9.2.2 TimeBase.MarshalSize Property

Gets the marshal size in bytes.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static int MarshalSize { get; }
```

Property Value

Type: Int32

Marshal size in bytes.

Reference

[TimeBase Class \[► 2068\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.9.2.3 TimeBase.Ticks Property

Returns the number of ticks that represent the value of this [TimeBase \[► 2068\]](#) (uint32 or uint64).

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint Ticks { get; }
```

Property Value

Type: UInt32

The ticks (in 100ns).

Implements

[IPlcOpenTimeBase.T1, T2..Ticks \[► 2044\]](#)

Reference

[TimeBase Class \[► 2068\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.9.2.4 TimeBase.Time Property

Gets the time value.

Namespace: [TwinCAT.PlcOpen \[▶ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public virtual TimeSpan Time { get; }
```

Property Value

Type: TimeSpan
The time.

Reference













[TimeBase Class \[▶ 2068\]](#)

[TwinCAT.PlcOpen Namespace \[▶ 2016\]](#)

6.10.9.3 TimeBase Methods

The [TimeBase \[▶ 2068\]](#) type exposes the following members.

Methods

	Name	Description
	Equals [▶ 2073]	Determines whether the specified Object is equal to this instance. (Overrides Object.Equals(Object).)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode [▶ 2074]	Gets the GetHashCode of the Address (Overrides Object.GetHashCode..)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	TimeToValue [▶ 2074]	Converts the Timespan to PlcOpen ticks.
		
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	ValueToTime(Int64) [▶ 2075]	Converts the timeValue (PlcOpen ticks) to TimeSpan
		
	ValueToTime(UInt32) [▶ 2076]	Converts the timeValue (PlcOpen ticks) to TimeSpan
		

Reference

[TimeBase Class \[▶ 2068\]](#)

[TwinCAT.PlcOpen Namespace \[▶ 2016\]](#)

6.10.9.3.1 TimeBase.Equals Method

Determines whether the specified Object is equal to this instance.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool Equals(  
    Object? obj  
)
```

Parameters

obj	Type: System.Object The object to compare with the current object.
-----	---

Return Value

Type: Boolean

true if the specified Object is equal to this instance; otherwise, false.

Reference

[TimeBase Class \[► 2068\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.9.3.2 TimeBase.GetHashCode Method

Gets the GetHashCode of the Address

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override int GetHashCode()
```

Return Value

Type: Int32

Reference

[TimeBase Class \[► 2068\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.9.3.3 TimeBase.TimeToValue Method

Converts the Timespan to PlcOpen ticks.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static long TimeToValue(
    TimeSpan time
)
```

Parameters

time	Type: System.TimeSpan The time.
------	------------------------------------

Return Value

Type: Int64



Reference

[TimeBase Class \[► 2068\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.9.3.4 TimeBase.ValueToTime Method

Overload List

	Name	Description
	ValueToTime(Int64) [► 2075]	Converts the timeValue (PlcOpen ticks) to TimeSpan
	ValueToTime(UInt32) [► 2076]	Converts the timeValue (PlcOpen ticks) to TimeSpan

Reference

[TimeBase Class \[► 2068\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.9.3.4.1 TimeBase.ValueToTime Method (Int64)

Converts the timeValue (PlcOpen ticks) to TimeSpan

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static TimeSpan ValueToTime(
    long timeValue
)
```

Parameters

timeValue	Type: System.Int64 The time value.
-----------	---------------------------------------

Return Value

Type: TimeSpan

Reference[TimeBase Class \[► 2068\]](#)[ValueToTime Overload \[► 2075\]](#)[TwinCAT.PlcOpen Namespace \[► 2016\]](#)**6.10.9.3.4.2 TimeBase.ValueToTime Method (UInt32)**

Converts the timeValue (PlcOpen ticks) to TimeSpan

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static TimeSpan ValueToTime(
    uint timeValue
)
```


Parameters

timeValue	Type: System.UInt32 The time value.
-----------	--

Return Value

Type: TimeSpan

Reference[TimeBase Class \[► 2068\]](#)[ValueToTime Overload \[► 2075\]](#)[TwinCAT.PlcOpen Namespace \[► 2016\]](#)**6.10.9.4 TimeBase Fields**The [TimeBase \[► 2068\]](#) type exposes the following members.**Fields**

	Name	Description
	internalTimeValue [► 2077]	The internal time value

Reference[TimeBase Class \[► 2068\]](#)[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.9.4.1 TimeBase.internalTimeValue Field

The internal time value

Namespace: [TwinCAT.PlcOpen](#) [[▶ 2016](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected uint internalTimeValue
```

Field Value

Type: UInt32

Reference

[TimeBase Class](#) [[▶ 2068](#)]

[TwinCAT.PlcOpen Namespace](#) [[▶ 2016](#)]

6.10.10 TOD Class

PLCOpen TimeOfDay class (32-Bit)

Inheritance Hierarchy

System.Object

[TwinCAT.PlcOpen.TimeBase](#) [[▶ 2068](#)]

TwinCAT.PlcOpen.TOD

Namespace: [TwinCAT.PlcOpen](#) [[▶ 2016](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229






Syntax

C#




```
public sealed class TOD : TimeBase
```

The TOD type exposes the following members.







Constructors

	Name	Description
	TOD. [▶ 2079]	Initializes a new instance of the TOD class.
	TOD(Int64) [▶ 2079]	Initializes a new instance of the TOD class.
	TOD(TimeSpan) [▶ 2079]	Initializes a new instance of the TOD class.
	TOD(UInt32) [▶ 2080]	Initializes a new instance of the TOD class.
	TOD(Int32, Int32, Int32, Int32, Int32) [▶ 2080]	Initializes a new instance of the TOD class.

Properties

	Name	Description
	InternalTimeValue [▶ 2071]	Gets the internal time value. (Inherited from TimeBase [▶ 2068] .)
	Ticks [▶ 2072]	Returns the number of ticks that represent the value of this TimeBase [▶ 2068] (uint32 or uint64). (Inherited from TimeBase [▶ 2068] .)
	Time [▶ 2072]	Gets the time value. (Inherited from TimeBase [▶ 2068] .)






Methods

	Name	Description
	Equals [▶ 2073]	Determines whether the specified Object is equal to this instance. (Inherited from TimeBase [▶ 2068] .)
	GetHashCode [▶ 2074]	Gets the GetHashCode of the Address (Inherited from TimeBase [▶ 2068] .)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	Parse [▶ 2082]	Parses the specified string to a TOD object.
	ToString [▶ 2082]	Returns a string that represents the current object. (Overrides Object.ToString..)
	TryParse [▶ 2083]	Tries to parse the string to a TOD object.

Reference

[TwinCAT.PlcOpen Namespace](#) [\[▶ 2016\]](#)

6.10.10.1 TOD Constructor**Overload List**

	Name	Description
	TOD . [▶ 2079]	Initializes a new instance of the TOD [▶ 2077] class.
	TOD(Int64) [▶ 2079]	Initializes a new instance of the TOD [▶ 2077] class.
	TOD(TimeSpan) [▶ 2079]	Initializes a new instance of the TOD [▶ 2077] class.
	TOD(UInt32) [▶ 2080]	Initializes a new instance of the TOD [▶ 2077] class.
	TOD(Int32, Int32, Int32, Int32, Int32) [▶ 2080]	Initializes a new instance of the TOD [▶ 2077] class.

Reference

[TOD Class](#) [\[▶ 2077\]](#)

[TwinCAT.PlcOpen Namespace](#) [\[▶ 2016\]](#)

6.10.10.1.1 TOD Constructor

Initializes a new instance of the [TOD \[▸ 2077\]](#) class.

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public TOD()
```

Reference

[TOD Class \[▸ 2077\]](#)

[TOD Overload \[▸ 2078\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.10.1.2 TOD Constructor (Int64)

Initializes a new instance of the [TOD \[▸ 2077\]](#) class.

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public TOD(  
    long time  
)
```

Parameters

time	Type: System.Int64 The time.
------	---------------------------------

Reference

[TOD Class \[▸ 2077\]](#)

[TOD Overload \[▸ 2078\]](#)

[TwinCAT.PlcOpen Namespace \[▸ 2016\]](#)

6.10.10.1.3 TOD Constructor (TimeSpan)

Initializes a new instance of the [TOD \[▸ 2077\]](#) class.

Namespace: [TwinCAT.PlcOpen \[▸ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public TOD(  
    TimeSpan timeSpan  
)
```

Parameters

timeSpan	Type: System.TimeSpan The time span.
----------	---

Reference

[TOD Class \[► 2077\]](#)

[TOD Overload \[► 2078\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.10.1.4 TOD Constructor (UInt32)

Initializes a new instance of the [TOD \[► 2077\]](#) class.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public TOD(
    uint time
)
```

Parameters

time	Type: System.UInt32 The time.
------	----------------------------------

Reference

[TOD Class \[► 2077\]](#)

[TOD Overload \[► 2078\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.10.1.5 TOD Constructor (Int32, Int32, Int32, Int32, Int32)

Initializes a new instance of the [TOD \[► 2077\]](#) class.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public TOD(
    int days,
    int hours,
    int minutes,
    int seconds,
    int milliseconds
)
```

Parameters

days	Type: System.Int32 The days.
------	---------------------------------

hours	Type: System.Int32 The hours.
minutes	Type: System.Int32 The minutes.
seconds	Type: System.Int32 The seconds.
milliseconds	Type: System.Int32 The milliseconds.

Reference

[TOD Class \[► 2077\]](#)




[TOD Overload \[► 2078\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.10.2 TOD Properties

The [TOD \[► 2077\]](#) type exposes the following members.

Properties

	Name	Description
	InternalTimeValue [► 2071]	Gets the internal time value. (Inherited from TimeBase [► 2068].)
	Ticks [► 2072]	Returns the number of ticks that represent the value of this TimeBase [► 2068] (uint32 or uint64). (Inherited from TimeBase [► 2068].)
	Time [► 2072]	Gets the time value. (Inherited from TimeBase [► 2068].)

Reference







[TOD Class \[► 2077\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.10.3 TOD Methods

The [TOD \[► 2077\]](#) type exposes the following members.

Methods

	Name	Description
	Equals [► 2073]	Determines whether the specified Object is equal to this instance. (Inherited from TimeBase [► 2068].)
	GetHashCode [► 2074]	Gets the GetHashCode of the Address (Inherited from TimeBase [► 2068].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	Parse [► 2082]	Parses the specified string to a TOD [► 2077] object.
	ToString [► 2082]	Returns a string that represents the current object. (Overrides Object.ToString..)
	TryParse [► 2083]	Tries to parse the string to a TOD [► 2077] object.

	Name	Description
S		

Reference

[TOD Class \[► 2077\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.10.3.1 TOD.Parse Method

Parses the specified string to a [TOD \[► 2077\]](#) object.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static TOD Parse(
    string str
)
```

Parameters

str	Type: System.String The string.
-----	------------------------------------

Return Value

Type: [TOD \[► 2077\]](#)
TOD.

Exceptions

Exception	Condition
FormatException	Cannot parse TOD object!

Reference

[TOD Class \[► 2077\]](#)

[TwinCAT.PlcOpen Namespace \[► 2016\]](#)

6.10.10.3.2 TOD.ToString Method

Returns a string that represents the current object.

Namespace: [TwinCAT.PlcOpen \[► 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```

Return Value

Type: String
 A string that represents the current object.

Reference

[TOD Class \[▶ 2077\]](#)

[TwinCAT.PlcOpen Namespace \[▶ 2016\]](#)

6.10.10.3.3 TOD.TryParse Method

Tries to parse the string to a [TOD \[▶ 2077\]](#) object.

Namespace: [TwinCAT.PlcOpen \[▶ 2016\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool TryParse(
    string str,
    out TOD?? ret
)
```

Parameters

str	Type: System.String The string.
ret	Type: TwinCAT.PlcOpen.TOD [▶ 2077] . The ret.

Return Value

Type: Boolean
 true if the [TOD \[▶ 2077\]](#) object could be parsed, false otherwise.

Reference




[TOD Class \[▶ 2077\]](#)



















[TwinCAT.PlcOpen Namespace \[▶ 2016\]](#)

6.11 TwinCAT.TypeSystem Namespace















Namespace for the common (non ADS dependent) type system.










Classes

	Class	Description
	AlignedMemberCollection [▶ 2973]	Class AlignedMemberCollection. Implements the MemberCollection [▶ 2802]
	AnySymbolSpecifier [▶ 2091]	Class AnySymbolSpecifier.
	AnyTypeSpecifier [▶ 2094]	Class AnyTypeSpecifier.

	Class	Description
	AnyTypeValueCreator [▶ 3020]	Class AnyTypeValueCreator.
	CannotAccessVirtualSymbolException [▶ 2103]	Cannot access virtual Symbol
	CannotResolveDataTypeException [▶ 2106]	Class CannotResolveDataTypeException. Implements the DataTypeException [▶ 2121]
	DataTypeCollection [▶ 2112]	Collection of DataTypes . [▶ 2475]
	DataTypeEventArgs [▶ 2118]	Class DataTypeEventArgs.
	DataTypeException [▶ 2121]	Data Type Exception
	DataTypeExtension [▶ 3021]	Extends the IDataType [▶ 2475] interface with more sophisticated functions.
	DataTypeNameEventArgs [▶ 2127]	Class DataTypeNameEventArgs.
	Dimension [▶ 2130]	Represents a single dimension of an IArrayType [▶ 2459]
	DimensionCollection [▶ 2133]	Collection class for Array Dimensions
	DynamicAliasInstance [▶ 2149]	Class DynamicAliasInstance. This class cannot be inherited.
	DynamicArrayInstance [▶ 2160]	Dynamic Array Instance
	DynamicArrayValue [▶ 3042]	Class DynamicArrayValue. Implements the DynamicValue [▶ 2347]
	DynamicEnumValue [▶ 3048]	Class DynamicEnumValue. Implements the DynamicValue [▶ 2347]
	DynamicInterfaceInstance [▶ 2995]	Dynamic struct instance
	DynamicOversamplingArrayInstance [▶ 2176]	Dynamic Array Instance
	DynamicPointerInstance [▶ 2186]	Dynamic Pointer Instance
	DynamicPointerValue [▶ 2196]	Class DynamicPointerValue.

	Class	Description
	DynamicReferenceInstance [▶ 2202]	Dynamic Reference Instance
	DynamicReferenceValue [▶ 2217]	Class DynamicReferenceValue.
	DynamicStructInstance [▶ 2251]	Dynamic struct instance
 	DynamicSymbol [▶ 2266]	Dynamic Symbol [▶ 2691] object.
	DynamicSymbolsCollection [▶ 2327]	Dynamic (Expandable) Symbols collection.
	DynamicUnionInstance [▶ 2336]	Dynamic union instance
 	DynamicValue [▶ 2347]	Dynamic value (uses RuntimeBinding for ISymbol [▶ 2691] value reading / writing).
	DynamicVirtualStructInstance [▶ 2374]	Dynamic struct instance
	EnumValue.T. [▶ 2384]	Enum Value
	EnumValueCollection [▶ 2391]	Class EnumValueCollection.
	EnumValueCollection.T. [▶ 2407]	Collection of EnumValues [▶ 2384]
	FieldCollection [▶ 2427]	Collection of IField [▶ 2535] objects.
	FluentAlignedMemberCollectionExtension [▶ 2980]	Class FluentStructTypeExtension.
	IDimensionCollectionExtension [▶ 2981]	Class IDimensionCollectionExtension.
	InsufficientAccessRightsException [▶ 2573]	Insufficient rights for access
	MarshalException [▶ 2795]	Common Marshalling Exception
	MemberCollection [▶ 2802]	Collection of IMember [▶ 2561] objects.









	Class	Description
	RawValueChangedEventArgs [▶ 2813]	Event args for the RawValueChanged [▶ 2774] event.
	ReadOnlyDataTypeCollection [▶ 2815]	ReadOnly Collection of IDataType [▶ 2475] objects.
	ReadOnlyDimensionCollection [▶ 2818]	ReadOnly version of the DimensionCollection [▶ 2133]
	ReadOnlyEnumValueCollection [▶ 2824]	Read only version of the EnumValueCollection.T. [▶ 2407]
	ReadOnlyEnumValueCollection.T. [▶ 2833]	Read only version of the EnumValueCollection.T. [▶ 2407]
	ReadOnlyFieldCollection [▶ 2841]	Read only collection of IField [▶ 2535] objects
	ReadOnlyMemberCollection [▶ 2845]	Read only collection of IMember [▶ 2561] objects
	ReadOnlyMethodParameterCollection [▶ 2852]	Read only RpcMethodParameterCollection [▶ 2915].
	ReadOnlyRpcMethodCollection [▶ 2855]	Read only RpcMethodCollection [▶ 2894]
	ReadOnlySymbolCollection [▶ 2861]	ReadOnly collection containing ISymbol [▶ 2691] objects.
	ReadOnlyTypeAttributeCollection [▶ 2866]	Read only version of the TypeAttributeCollection [▶ 2952]
	ResultDataTypes [▶ 2872]	Class representing the asynchronous result of reading a IDataTypeCollection [▶ 2484] via ADS. Implements the ResultValue.TValue. [▶ 1181]
	ResultDynamicSymbols [▶ 2876]	Class representing the asynchronous result of reading a dynamic symbol collection via ADS. Implements the ResultValue.TValue. [▶ 1181]
	ResultSymbols [▶ 2880]	Class representing the asynchronous result of reading an symbol collection of type ISymbolCollection.T. [▶ 2700] via ADS. Implements the ResultValue.TValue. [▶ 1181]
	ResultSymbols.T. [▶ 2884]	Class representing the asynchronous result of reading a symbol enumeration of type IEnumerable.T. via ADS. Implements the ResultValue.TValue. [▶ 1181]
	RpcInvokeException [▶ 2887]	Class RpcInvokeException . Implements the SymbolException [▶ 2933]

	Class	Description
	RpcMethodCollection [▶ 2894]	Collection of RpcMethods . [▶ 2625]
	RpcMethodNotSupportedException [▶ 2910]	Symbol Exception
	RpcMethodParameterCollection [▶ 2915]	Collection of RPC method parameters
	SymbolCollection [▶ 2928]	Interface represents a collection of ISymbol [▶ 2691] objects.
	SymbolException [▶ 2933]	Symbol bound exceptions
	TypeAttribute [▶ 2946]	ADS Attribute
	TypeAttributeCollection [▶ 2952]	Collection of AdsAttributes [▶ 2725]
	ValueChangedBaseEventArgs [▶ 2968]	Event args for the RawValueChanged [▶ 2774] event.
	ValueChangedEventArgs [▶ 2971]	Event args for the ValueChanged [▶ 2790] event.





















Structures
















	Structure	Description
	UInt32Ptr [▶ 3054]	Represents an 32-Bit Pointer (Process independant)
	UInt64Ptr [▶ 3070]	Represents an 64-Bit Pointer (Process independant)





Interfaces

	Interface	Description
	IAliasInstance [▶ 2433]	Interface representing an instance of an IAliasType [▶ 2436].
	IAliasType [▶ 2436]	Interface representing an Alias Type
	IAnyTypeMarshaler [▶ 2441]	Interface IAnyTypeMarshaler Implements the IGenericTypeMarshaler [▶ 2542]
	IArrayInstance [▶ 2453]	Interface representing an array instance
	IArrayType [▶ 2459]	Interface representing an array DataType [▶ 2475].
	IArrayValue [▶ 2465]	Interface IArrayValue
	IAttributedInstance [▶ 2469]	Interface IAttributedInstance
	IBitSize [▶ 2472]	Interface IBitSize









	Interface	Description
	IDataType [▶ 2475]	Base interface for objects representing data types
	IDataTypeCollection [▶ 2484]	Interface IDataValueCollection Implements the ICollection.T.
	IDataTypeCollection.T. [▶ 2486]	Data Type container interface
	IDimension [▶ 2490]	Interface representing a single Dimension [▶ 2490] of an ArrayType [▶ 2459].
	IDimensionCollection [▶ 2492]	Interface IDimensionCollection
	IDynamicSymbol [▶ 2496]	Interface IDynamicSymbol
	IDynamicSymbolLoader [▶ 2500]	Dynamic symbol loader interface
	IDynamicSymbolsCollection [▶ 2503]	Interface IDynamicSymbolsContainer Implements the IDynamicMetaObjectProvider
	IDynamicValue [▶ 2504]	Interface IDynamicValue Implements the IDynamicMetaObjectProvider Implements the IValue [▶ 2745] Implements the IStructValue [▶ 2678] Implements the IArrayValue [▶ 2465]
	IEnumType [▶ 2507]	Common Enum type interface
	IEnumType.T. [▶ 2516]	Interface representing an enum type
	IEnumValue [▶ 2524]	Generic interface for EnumValues
	IEnumValueCollection [▶ 2527]	Interface IEnumValueCollection
	IEnumValueCollection.TEnumValue, TValue. [▶ 2529]	Interface for collections of IEnumValues [▶ 2524]. Implements the ICollection.T.
	IField [▶ 2535]	Specifies a single field/member of a Struct DataType [▶ 2671].
	IFieldCollection [▶ 2538]	Interface IFieldCollection Implements the IInstanceCollection.T. [▶ 2554]
	IGenericTypeMarshaller [▶ 2542]	Interface IGenericTypeMarshaller Implements the ITypeMarshaller [▶ 2731]
	IHierarchicalSymbol [▶ 2545]	Bindable Symbol interface (for internal use only)
	IInstance [▶ 2549]	Interface specifying instance objects.
	IInstanceCollection.T. [▶ 2554]	Generic InstanceCollection interface.
	IInterfaceInstance [▶ 3014]	Interface representing an instance of a IInterfaceType [▶ 2983]

	Interface	Description
	IInterfaceType [▶ 2983]	Interface representing an Interface Data type. Implements the IDataType [▶ 2475]
	IMember [▶ 2561]	Specifies a single field/member of a Struct DataType [▶ 2671].
	IMemberCollection [▶ 2565]	Interface IMemberCollection Implements the IInstanceCollection.T. [▶ 2554]
	INamespaceCollecti on [▶ 2570]	Interface INamespaceCollection
	INamespaceCollecti on.T. [▶ 2571]	Interface INamespaceCollection
	IOversamplingArrayI nstance [▶ 2576]	Interface IOversamplingArrayInstance
	IPointerType [▶ 2580]	Interface representing an instance of an IPointerType [▶ 2584]
	IPointerType [▶ 2584]	Interface representing a pointer type
	IPrimitiveType [▶ 2588]	Interface IPrimitiveType
	IProcessImageAddr ess [▶ 2593]	Interface describing a Process Image Address
	IReferenceInstance [▶ 2595]	Interface representing an instance of an IReferenceType [▶ 2600]
	IReferenceType [▶ 2600]	Interface representing a reference type
	IRpcCallableInstanc e [▶ 2606]	Interface for an RPC callable PLC Method (Remote procedure call)
	IRpcCallableType [▶ 2624]	Interface representing an RPC callable IStructType [▶ 2671]
	IRpcMethod [▶ 2625]	Interface describes an RPC Method
	IRpcMethodCollecti on [▶ 2629]	Interface for RPC Method collections.
	IRpcMethodParame ter [▶ 2635]	Interface IRpcMethodParameter
	IRpcMethodParame terCollection [▶ 2639]	Interface IRpcMethodParameterCollection
	IRpcStructInstance [▶ 2642]	Obsolete. Interface IRpcStructInstance
	IRpcStructType [▶ 2990]	Obsolete. Interface IRpcStructType Implements the IRpcCallableType [▶ 2624] Implements the IStructType [▶ 2671]

	Interface	Description
	IStringInstance [▶ 2646]	Interface IStringInstance
	IStringMarshaler [▶ 2650]	Common interface for marshalling ADS string values.
	IStringType [▶ 2661]	Interface representing a string IDataType [▶ 2475]
	IStructInstance [▶ 2666]	Interface representing an instance of a IStructType [▶ 2671]
	IStructType [▶ 2671]	Interface representing Struct data types
	IStructValue [▶ 2678]	Interface IStructValue
	ISubRangeType [▶ 2681]	Interface representing a SubRange type
	ISubRangeType.T. [▶ 2686]	Interface representing a SubRange type
	ISymbol [▶ 2691]	Interface specifying Symbols (
	ISymbolCollection [▶ 2697]	Interface ISymbolCollection Implements the IInstanceCollection.T. [▶ 2554]
	ISymbolCollection.T. [▶ 2700]	Interface ISymbolCollection
	ISymbolFactory [▶ 2702]	Symbol Factory Interface
	ISymbolFactoryServicesProvider [▶ 2711]	Symbol Value Access interface
	ISymbolInfo [▶ 2712]	Interface ISymbolInfo
	ISymbolLoader [▶ 2714]	Symbol Loader interface
	ISymbolProvider [▶ 2717]	Symbol Provider interface.
	ISymbolServer [▶ 2719]	Symbol Server Interface
	ITypeAttribute [▶ 2725]	Interface for ADS attributes
	ITypeAttributeCollection [▶ 2726]	Interface ITypeAttributeCollection
	ITypeMarshaler [▶ 2731]	Interface ITypeMarshaler
	IUnionInstance [▶ 2737]	Interface for an Instance of the IUnionType [▶ 2740].
	IUnionType [▶ 2740]	Interface for an union data type.
	IValue [▶ 2745]	Symbol Value Interface
	IValueAccessorProvider [▶ 2752]	Interface IValueAccessorProvider

	Interface	Description
	IValueAnySymbol [▶ 2753]	Interface IValueAnySymbol
	IValueRawSymbol [▶ 2765]	Interface IValueRawSymbol
	IValueSymbol [▶ 2775]	Interface for a ISymbol [▶ 2691] that supports values.
	IVirtualStructInstance [▶ 2790]	Virtual Struct instance interface.

Enumerations

	Enumeration	Description
	DataTypeCategory [▶ 2111]	Category of a DataType / Instance
	DataTypeResolveStrategy [▶ 3042]	Enum DataTypeResolveStrategy
	InstanceCollectionMode [▶ 2573]	Enum InstanceCollectionMode
	MethodParamFlags [▶ 2812]	Flag set specifying the MethodParameter context
	PrimitiveTypeFlags [▶ 2812]	Enum PrimitiveTypeFlags
	StringConvertMode [▶ 2927]	Enum StringConvertMode
	SymbolAccessRights [▶ 2928]	Enum specifying Access Rights to symbols
	SymbolIterationMask [▶ 3019]	Mask Flagset to specify filters for SymbolIterator.T. [▶ 3172].

6.11.1 AnySymbolSpecifier Class

Class AnySymbolSpecifier.

Inheritance Hierarchy

System.Object
 TwinCAT.TypeSystem.AnySymbolSpecifier

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#



```
public class AnySymbolSpecifier
```

The AnySymbolSpecifier type exposes the following members.







Constructors

	Name	Description
	AnySymbolSpecifier [▶ 2092]	Initializes a new instance of the AnySymbolSpecifier class.

Properties

	Name	Description
	InstancePath [▶ 2093]	Gets the instance path.
	TypeSpecifier [▶ 2093]	Gets the type specifier.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Remarks

Specifies an Symbol path together with an [AnyTypeSpecifier](#) [▶ 2094] to address a symbol for an Read/Write operation

Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.1.1 AnySymbolSpecifier Constructor

Initializes a new instance of the [AnySymbolSpecifier](#) [▶ 2091] class.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public AnySymbolSpecifier(
    string instancePath,
    AnyTypeSpecifier spec
)
```

Parameters

instancePath	Type: System.String The instance path.
--------------	---

spec	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [▸ 2094] The spec.
------	---

Reference



[AnySymbolSpecifier Class \[▸ 2091\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.1.2 AnySymbolSpecifier Properties

The [AnySymbolSpecifier \[▸ 2091\]](#) type exposes the following members.

Properties

	Name	Description
	InstancePath [▸ 2093]	Gets the instance path.
	TypeSpecifier [▸ 2093]	Gets the type specifier.

Reference

[AnySymbolSpecifier Class \[▸ 2091\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.1.2.1 AnySymbolSpecifier.InstancePath Property

Gets the instance path.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string InstancePath { get; }
```

Property Value

Type: String

The instance path.

Reference

[AnySymbolSpecifier Class \[▸ 2091\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.1.2.2 AnySymbolSpecifier.TypeSpecifier Property

Gets the type specifier.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AnyTypeSpecifier TypeSpecifier { get; }
```

Property Value

Type: [AnyTypeSpecifier](#) [[▶ 2094](#)]

The type specifier.

Reference







[AnySymbolSpecifier Class](#) [[▶ 2091](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.1.3 AnySymbolSpecifier Methods

The [AnySymbolSpecifier](#) [[▶ 2091](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[AnySymbolSpecifier Class](#) [[▶ 2091](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.2 AnyTypeSpecifier Class

Class AnyTypeSpecifier.

Inheritance Hierarchy

System.Object

 TwinCAT.TypeSystem.AnyTypeSpecifier

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229







Syntax

C#








```
public class AnyTypeSpecifier
```

The AnyTypeSpecifier type exposes the following members.

Properties

	Name	Description
	Category [▶ 2099]	Category of the AnyTypeSpecifier
	Dimensions [▶ 2101]	Gets the dimensions of an array.
	Element [▶ 2101]	The element type (Arrays)
	Encoding [▶ 2102]	Gets the encoding
	StrLen [▶ 2100]	The String length (only for String [▶ 2111])
	Type [▶ 2101]	Managed type

Methods

	Name	Description
	Create [▶ 2103]	Creates the AnyTypeSpecifier.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)



Remarks

The AnyTypeSpecifier is used to specify out/return parameters for ReadAny, InvokeRpc-Methods or typed AdsNotifications. The Args parameter: String: Index-0 Length of String String[] Index-0 Length of String Index-1 Length of First Dimension Index-2 Length of Second Dimension Arrays Index-0 Length of Second Dimension Index-2 Length of Second Dimension

Reference


[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]





Also see about this

-  [AnyTypeSpecifier.DimLengths Property](#) [[▶ 2100](#)]
-  [AnyTypeSpecifier.ElementType Property](#) [[▶ 2100](#)]

6.11.2.1 AnyTypeSpecifier Constructor

Overload List

	Name	Description
	AnyTypeSpecifier(Object) [▶ 2096]	Initializes a new instance of the AnyTypeSpecifier [▶ 2094] class from the specified prototype

	Name	Description
	AnyTypeSpecifier(Type) [▸ 2096]	Initializes a new instance of the AnyTypeSpecifier [▸ 2094] class.
	AnyTypeSpecifier(Type, IList.IDimensionCollection.) [▸ 2097]	Initializes a new instance of the AnyTypeSpecifier [▸ 2094] class.
	AnyTypeSpecifier(Type, Int32) [▸ 2097]	Initializes a new instance of the AnyTypeSpecifier [▸ 2094] class.
	AnyTypeSpecifier(Type, .Int32.) [▸ 2098]	Initializes a new instance of the AnyTypeSpecifier [▸ 2094] class.

Reference

[AnyTypeSpecifier Class](#) [[▸ 2094](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.2.1.1 AnyTypeSpecifier Constructor (Object)

Initializes a new instance of the [AnyTypeSpecifier](#) [[▸ 2094](#)] class from the specified prototype

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public AnyTypeSpecifier(
    Object prototype
)
```

Parameters

prototype Type: [System.Object](#)
The prototype.

Reference

[AnyTypeSpecifier Class](#) [[▸ 2094](#)]

[AnyTypeSpecifier Overload](#) [[▸ 2095](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.2.1.2 AnyTypeSpecifier Constructor (Type)

Initializes a new instance of the [AnyTypeSpecifier](#) [[▸ 2094](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public AnyTypeSpecifier(  
    Type type  
)
```

Parameters

type Type: [System.Type](#)
The type.

Reference

[AnyTypeSpecifier Class \[▸ 2094\]](#)

[AnyTypeSpecifier Overload \[▸ 2095\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.2.1.3 **AnyTypeSpecifier Constructor (Type, IList.IDimensionCollection.)**

Initializes a new instance of the [AnyTypeSpecifier \[▸ 2094\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public AnyTypeSpecifier(  
    Type type,  
    IList<IDimensionCollection> dimLengths  
)
```

Parameters

type Type: [System.Type](#)
The type.

dimLengths Type: [System.Collections.Generic.IList.IDimensionCollection \[▸ 2492\]](#).
The dim lengths.

Reference

[AnyTypeSpecifier Class \[▸ 2094\]](#)

[AnyTypeSpecifier Overload \[▸ 2095\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.2.1.4 **AnyTypeSpecifier Constructor (Type, Int32)**

Initializes a new instance of the [AnyTypeSpecifier \[▸ 2094\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public AnyTypeSpecifier(
    Type type,
    int strLen
)
```

Parameters

type Type: [System.Type](#)
The type.

strLen Type: [System.Int32](#)
Length of the string.

Exceptions

Exception	Condition
ArgumentOutOfRangeException	type

Reference

[AnyTypeSpecifier Class](#) [▸ 2094]

[AnyTypeSpecifier Overload](#) [▸ 2095]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.2.1.5 AnyTypeSpecifier Constructor (Type, .Int32.)

Initializes a new instance of the [AnyTypeSpecifier](#) [▸ 2094] class.

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public AnyTypeSpecifier(
    Type type,
    int[] dimLengths
)
```

Parameters

type Type: [System.Type](#)
The type.

dimLengths Type: [.System.Int32](#).
The dim lengths.

Exceptions

Exception	Condition
ArgumentOutOfRangeException	type

Reference

[AnyTypeSpecifier Class \[▸ 2094\]](#)







[AnyTypeSpecifier Overload \[▸ 2095\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.2.2 AnyTypeSpecifier Properties

The [AnyTypeSpecifier \[▸ 2094\]](#) type exposes the following members.

Properties



	Name	Description
	Category [▸ 2099]	Category of the AnyTypeSpecifier [▸ 2094]
	Dimensions [▸ 2101]	Gets the dimensions of an array.
	Element [▸ 2101]	The element type (Arrays)
	Encoding [▸ 2102]	Gets the encoding
	StrLen [▸ 2100]	The String length (only for String [▸ 2111])
	Type [▸ 2101]	Managed type

Reference

[AnyTypeSpecifier Class \[▸ 2094\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

Also see about this

-  [AnyTypeSpecifier.DimLengths Property \[▸ 2100\]](#)
-  [AnyTypeSpecifier.ElementType Property \[▸ 2100\]](#)

6.11.2.2.1 AnyTypeSpecifier.Category Property

Category of the [AnyTypeSpecifier \[▸ 2094\]](#)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DataTypeCategory Category { get; }
```

Property Value

Type: [DataTypeCategory \[▸ 2111\]](#)

Reference

[AnyTypeSpecifier Class \[▸ 2094\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.2.2 AnyTypeSpecifier.DimLengths Property

List of jagged Dimensions (Arrays and jagged arrays)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public IList<IDimensionCollection> DimLengths { get; }
```

Property Value

Type: [IList.IDimensionCollection](#) [[▶ 2492](#)].

Reference

[AnyTypeSpecifier Class](#) [[▶ 2094](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.2.3 AnyTypeSpecifier.ElementType Property

The element type (Arrays)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public AnyTypeSpecifier? ElementType { get; }
```

Property Value

Type: [AnyTypeSpecifier](#) [[▶ 2094](#)]

Reference

[AnyTypeSpecifier Class](#) [[▶ 2094](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.2.4 AnyTypeSpecifier.StrLen Property

The String length (only for [String](#) [[▶ 2111](#)])

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int StrLen { get; }
```

Property Value

Type: Int32

Reference

[AnyTypeSpecifier Class \[▸ 2094\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.2.2.5 AnyTypeSpecifier.Type Property

Managed type

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Type Type { get; }
```

Property Value

Type: Type

Reference

[AnyTypeSpecifier Class \[▸ 2094\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.2.2.6 AnyTypeSpecifier.Dimensions Property

Gets the dimensions of an array.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDimensionCollection? Dimensions { get; }
```

Property Value

Type: [IDimensionCollection \[▸ 2492\]](#)

The dimensions.

Reference

[AnyTypeSpecifier Class \[▸ 2094\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.2.2.7 AnyTypeSpecifier.Element Property

The element type (Arrays)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AnyTypeSpecifier? Element { get; }
```

Property Value

Type: [AnyTypeSpecifier](#) [[▶ 2094](#)]

Reference

[AnyTypeSpecifier Class](#) [[▶ 2094](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.2.2.8 AnyTypeSpecifier.Encoding Property

Gets the encoding

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Encoding Encoding { get; }
```

Property Value

Type: [Encoding](#)
The encoding.

Reference







[AnyTypeSpecifier Class](#) [[▶ 2094](#)]


[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.2.3 AnyTypeSpecifier Methods

The [AnyTypeSpecifier](#) [[▶ 2094](#)] type exposes the following members.

Methods

	Name	Description
	Create [▶ 2103]	Creates the AnyTypeSpecifier [▶ 2094].
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)

	Name	Description
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[AnyTypeSpecifier Class \[▸ 2094\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.2.3.1 AnyTypeSpecifier.Create Method

Creates the [AnyTypeSpecifier \[▸ 2094\]](#).

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AnyTypeSpecifier Create(
    Type type,
    int[]? args,
    Encoding? encoding
)
```

Parameters

type	Type: System.Type The data type.
args	Type: .System.Int32. The AnyType arguments.
encoding	Type: System.Text.Encoding The encoding used by the type.

Return Value

Type: [AnyTypeSpecifier \[▸ 2094\]](#)
AnyTypeSpecifier.

Reference

[AnyTypeSpecifier Class \[▸ 2094\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.3 CannotAccessVirtualSymbolException Class

Cannot access virtual Symbol

Inheritance Hierarchy

System.Object
 System.Exception
 [TwinCAT.AdsException \[▸ 61\]](#)
 [TwinCAT.TypeSystem.SymbolException \[▸ 2933\]](#)
 TwinCAT.TypeSystem.CannotAccessVirtualSymbolException

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229


Syntax

C#











```
[SerializableAttribute]
public sealed class CannotAccessVirtualSymbolException : SymbolException
```

The CannotAccessVirtualSymbolException type exposes the following members.







Constructors

	Name	Description
	CannotAccessVirtualSymbolException [▶ 2105]	Initializes a new instance of the CannotAccessVirtualSymbolException class.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	InstancePath [▶ 2944]	Gets the instance path. (Inherited from SymbolException [▶ 2933].)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	Symbol [▶ 2944]	Gets the symbol. (Inherited from SymbolException [▶ 2933].)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData [▶ 2945]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from SymbolException [▶ 2933].)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.3.1 CannotAccessVirtualSymbolException Constructor

Initializes a new instance of the [CannotAccessVirtualSymbolException \[▶ 2103\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public CannotAccessVirtualSymbolException(
    ISymbol symbol
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol.
--------	--

Reference











[CannotAccessVirtualSymbolException Class \[▶ 2103\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.3.2 CannotAccessVirtualSymbolException Properties

The [CannotAccessVirtualSymbolException \[▶ 2103\]](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	InstancePath [▶ 2944]	Gets the instance path. (Inherited from SymbolException [▶ 2933] .)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	Symbol [▶ 2944]	Gets the symbol. (Inherited from SymbolException [▶ 2933] .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Reference







[CannotAccessVirtualSymbolException Class \[► 2103\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.3.3 CannotAccessVirtualSymbolException Methods

The [CannotAccessVirtualSymbolException \[► 2103\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData [► 2945]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from SymbolException [► 2933] .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Reference

[CannotAccessVirtualSymbolException Class \[► 2103\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.4 CannotResolveDataTypeException Class

Class [CannotResolveDataTypeException](#). Implements the [DataTypeException \[► 2121\]](#)

Inheritance Hierarchy

System.Object
 System.Exception
 [TwinCAT.AdsException \[► 61\]](#)
 [TwinCAT.TypeSystem.DataTypeException \[► 2121\]](#)
 TwinCAT.TypeSystem.CannotResolveDataTypeException

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)




Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**











```
[SerializableAttribute]
public class CannotResolveDataTypeException : DataTypeException
```

The [CannotResolveDataTypeException](#) type exposes the following members.





Constructors





	Name	Description
	CannotResolveDataTypeException(Instance) [▶ 2109]	Initializes a new instance of the CannotResolveDataTypeException class.
	CannotResolveDataTypeException(String) [▶ 2108]	Initializes a new instance of the CannotResolveDataTypeException class.
	CannotResolveDataTypeException(SerializationInfo, StreamingContext) [▶ 2109]	Initializes a new instance of the CannotResolveDataTypeException class.

Properties


	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	DataType [▶ 2125]	Gets the type of the data. (Inherited from DataTypeException [▶ 2121].)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)
	TypeName [▶ 2125]	Gets the name of the type. (Inherited from DataTypeException [▶ 2121].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)

	Name	Description
	GetObjectData [▶ 2126]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from DataTypeException [▶ 2121].)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)




Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

[TwinCAT.TypeSystem.DataTypeException](#) [▶ 2121]

6.11.4.1 CannotResolveDataTypeException Constructor

Overload List

	Name	Description
	CannotResolveDataTypeException(Instance) [▶ 2109]	Initializes a new instance of the CannotResolveDataTypeException [▶ 2106] class.
	CannotResolveDataTypeException(String) [▶ 2108]	Initializes a new instance of the CannotResolveDataTypeException [▶ 2106] class.
	CannotResolveDataTypeException(SerializationInfo, StreamingContext) [▶ 2109]	Initializes a new instance of the CannotResolveDataTypeException [▶ 2106] class.

Reference

[CannotResolveDataTypeException Class](#) [▶ 2106]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.4.1.1 CannotResolveDataTypeException Constructor (String)

Initializes a new instance of the [CannotResolveDataTypeException](#) [▶ 2106] class.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public CannotResolveDataTypeException(  
    string typeName  
)
```

Parameters

typeName	Type: System.String Name of the type.
----------	--

Reference

[CannotResolveDataTypeException Class \[▸ 2106\]](#)

[CannotResolveDataTypeException Overload \[▸ 2108\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.4.1.2 CannotResolveDataTypeException Constructor (Instance)

Initializes a new instance of the [CannotResolveDataTypeException \[▸ 2106\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public CannotResolveDataTypeException(  
    IInstance symbol  
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IInstance [▸ 2549] The symbol.
--------	--

Reference

[CannotResolveDataTypeException Class \[▸ 2106\]](#)

[CannotResolveDataTypeException Overload \[▸ 2108\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.4.1.3 CannotResolveDataTypeException Constructor (SerializationInfo, StreamingContext)

Initializes a new instance of the [CannotResolveDataTypeException \[▸ 2106\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected CannotResolveDataTypeException(
    SerializationInfo serializationInfo,
    StreamingContext streamingContext
)
```

Parameters

serializationInfo	Type: System.Runtime.Serialization.SerializationInfo The serialization information.
streamingContext	Type: System.Runtime.Serialization.StreamingContext The streaming context.

Reference

[CannotResolveDataTypeException Class \[► 2106\]](#)











[CannotResolveDataTypeException Overload \[► 2108\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.4.2 CannotResolveDataTypeException Properties

The [CannotResolveDataTypeException \[► 2106\]](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	DataType [► 2125]	Gets the type of the data. (Inherited from DataTypeException [► 2121] .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)
	TypeName [► 2125]	Gets the name of the type. (Inherited from DataTypeException [► 2121] .)

Reference









[CannotResolveDataTypeException Class \[► 2106\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.4.3 CannotResolveDataTypeException Methods

The [CannotResolveDataTypeException](#) [▶ 2106] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData [▶ 2126]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from DataTypeException [▶ 2121].)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Reference


[CannotResolveDataTypeException Class](#) [▶ 2106]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.4.4 CannotResolveDataTypeException Events

The [CannotResolveDataTypeException](#) [▶ 2106] type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)

Reference

[CannotResolveDataTypeException Class](#) [▶ 2106]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.5 DataTypeCategory Enumeration

Category of a DataType / Instance

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum DataTypeCategory
```

Members

	Member name	Value	Description
	None	0	Uninitialized / NotProcessed (0)
	Primitive	1	Simple / Base Data Type (1)
	Alias	2	Alias data type (2)
	Enum	3	Enumeration data type (3)
	Array	4	Array data type (4)
	Struct	5	Structure data type (5)
	FunctionBlock	6	Function block (POU) (6)
	Program	7	Program (POU) (7)
	Function	8	Function (POU) (8)
	SubRange	9	SubRange (9)
	String	10	Fixed length string (10)
	Bitset	12	Bitset (12)
	Pointer	13	Pointer type (13)
	Union	14	Union type (14)
	Reference	15	Reference type (15)
	Interface	16	The interface

Reference

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.6 DataTypeCollection Class

Collection of [DataTypes](#). [► 2475]

Inheritance Hierarchy

System.Object

[TwinCAT.TypeSystem.Generic.DataTypeCollection](#) [► 3087].[IDataType](#) [► 2475].

[TwinCAT.TypeSystem.DataTypeCollection](#)

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229


Syntax


C#

```
public class DataTypeCollection : DataTypeCollection<IDataType>,
    IDataTypeInfoCollection, IDataTypeInfoCollection<IDataType>, IList<IDataType>,
    ICollection<IDataType>, IEnumerable<IDataType>, IEnumerable
```






The `DataTypeCollection` type exposes the following members.

Constructors
















	Name	Description
	DataTypeCollection. [► 2114]	Initializes a new instance of the <code>DataTypeCollection</code> class.







	Name	Description
	DataTypeCollection(IEnumerable.IDataType.) [▶ 2115]	Initializes a new instance of the <code>DataTypeCollection</code> class (Copy constructor).

Properties


	Name	Description
	Count [▶ 3090]	Gets the count of contained IDataType [▶ 2475]s. (Inherited from DataTypeCollection.T. [▶ 3087].)
	Empty [▶ 2116]	Gets the empty collection.
	IsReadOnly [▶ 3091]	Gets a value indicating whether this instance is read only. (Inherited from DataTypeCollection.T. [▶ 3087].)
	Item.Int32. [▶ 3092]	Gets or sets the IDataType [▶ 2475] at the specified index. (Inherited from DataTypeCollection.T. [▶ 3087].)
	Item.String. [▶ 3092]	Gets the IDataType [▶ 2475] with the specified name. (Inherited from DataTypeCollection.T. [▶ 3087].)

Methods

	Name	Description
	Add [▶ 3094]	Adds the specified item to the collection. (Inherited from DataTypeCollection.T. [▶ 3087].)
	AddRange [▶ 3094]	Adds a range of types (Inherited from DataTypeCollection.T. [▶ 3087].)
	AsReadOnly [▶ 2117]	Returns A <code>ReadOnly-Version</code> of the <code>DataTypeCollection</code> .
	Clear [▶ 3095]	Clears the collection. (Inherited from DataTypeCollection.T. [▶ 3087].)
	Clone [▶ 2117]	Clones this <code>DataTypeCollection</code> (Shallow Copy)
	Contains [▶ 3096]	Determines whether this <code>DataTypeCollection</code> contains the specified IDataType [▶ 2475]. (Inherited from DataTypeCollection.T. [▶ 3087].)
	ContainsType [▶ 3096]	Determines whether the container contains the specified IDataType [▶ 2475]. (Inherited from DataTypeCollection.T. [▶ 3087].)
	CopyTo [▶ 3097]	Copies the data types to the specified array, starting at the array index. (Inherited from DataTypeCollection.T. [▶ 3087].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from <code>Object</code> .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from <code>Object</code> .)
	GetEnumerator [▶ 3098]	Gets the enumerator. (Inherited from DataTypeCollection.T. [▶ 3087].)
	GetHashCode	Serves as the default hash function. (Inherited from <code>Object</code> .)
	GetType	Gets the Type of the current instance. (Inherited from <code>Object</code> .)
	IndexOf [▶ 3098]	Determines the Index of the specified IDataType [▶ 2475]. (Inherited from DataTypeCollection.T. [▶ 3087].)
	Insert [▶ 3099]	Inserts an IDataType [▶ 2475] into the <code>DataTypeCollection</code> . (Inherited from DataTypeCollection.T. [▶ 3087].)

	Name	Description
	LookupType [▶ 3099]	Determines the specified IDataType [▶ 2475] (Inherited from DataTypeCollection.T. [▶ 3087].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 3100]	Removes the specified IDataType [▶ 2475]. (Inherited from DataTypeCollection.T. [▶ 3087].)
	RemoveAt [▶ 3100]	Removes the IDataType [▶ 2475] object at the specified index. (Inherited from DataTypeCollection.T. [▶ 3087].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetType [▶ 3101]	Tries to get the specified IDataType [▶ 2475] from the IDataTypeCollection.T. [▶ 2486]. (Inherited from DataTypeCollection.T. [▶ 3087].)

Fields



	Name	Description
	list [▶ 3102]	Internal list of data types (Inherited from DataTypeCollection.T. [▶ 3087].)

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.6.1 DataTypeCollection Constructor

Overload List

	Name	Description
	DataTypeCollection. [▶ 2114]	Initializes a new instance of the DataTypeCollection [▶ 2112] class.
	DataTypeCollection(IEnumerable.IDataT ype.) [▶ 2115]	Initializes a new instance of the DataTypeCollection [▶ 2112] class (Copy constructor).

Reference

[DataTypeCollection Class](#) [[▶ 2112](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.6.1.1 DataTypeCollection Constructor

Initializes a new instance of the [DataTypeCollection](#) [[▶ 2112](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DataTypeCollection()
```

Reference

[DataTypeCollection Class \[▶ 2112\]](#)

[DataTypeCollection Overload \[▶ 2114\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.6.1.2 DataTypeCollection Constructor (IEnumerable.IDataType.)

Initializes a new instance of the [DataTypeCollection \[▶ 2112\]](#) class (Copy constructor).

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DataTypeCollection(
    IEnumerable<IDataType> coll
)
```

Parameters

coll	Type: System.Collections.Generic.IEnumerable.IDataType [▶ 2475]. The coll.
------	---

Reference

[DataTypeCollection Class \[▶ 2112\]](#)






[DataTypeCollection Overload \[▶ 2114\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.6.2 DataTypeCollection Properties

The [DataTypeCollection \[▶ 2112\]](#) type exposes the following members.

Properties

	Name	Description
	Count [▶ 3090]	Gets the count of contained IDataType [▶ 2475] s. (Inherited from DataTypeCollection.T. [▶ 3087].)
	Empty [▶ 2116]	Gets the empty collection.
	IsReadOnly [▶ 3091]	Gets a value indicating whether this instance is read only. (Inherited from DataTypeCollection.T. [▶ 3087].)
	Item.Int32. [▶ 3092]	Gets or sets the IDataType [▶ 2475] at the specified index. (Inherited from DataTypeCollection.T. [▶ 3087].)
	Item.String. [▶ 3092]	Gets the IDataType [▶ 2475] with the specified name. (Inherited from DataTypeCollection.T. [▶ 3087].)

Reference

[DataTypeCollection Class \[▶ 2112\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.6.2.1 DataTypeCollection.Empty Property

Gets the empty collection.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static DataTypeCollection Empty { get; }
```

Property Value

Type: [DataTypeCollection \[▶ 2112\]](#)

The empty.

Reference














[DataTypeCollection Class \[▶ 2112\]](#)









[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.6.3 DataTypeCollection Methods

The [DataTypeCollection \[▶ 2112\]](#) type exposes the following members.

Methods

	Name	Description
	Add [▶ 3094]	Adds the specified item to the collection. (Inherited from DataTypeCollection.T. [▶ 3087].)
	AddRange [▶ 3094]	Adds a range of types (Inherited from DataTypeCollection.T. [▶ 3087].)
	AsReadOnly [▶ 2117]	Returns A ReadOnly-Version of the DataTypeCollection [▶ 2112] .
	Clear [▶ 3095]	Clears the collection. (Inherited from DataTypeCollection.T. [▶ 3087].)
	Clone [▶ 2117]	Clones this DataTypeCollection [▶ 2112] (Shallow Copy)
	Contains [▶ 3096]	Determines whether this DataTypeCollection [▶ 2112] contains the specified IDataType [▶ 2475] . (Inherited from DataTypeCollection.T. [▶ 3087].)
	ContainsType [▶ 3096]	Determines whether the container contains the specified IDataType [▶ 2475] . (Inherited from DataTypeCollection.T. [▶ 3087].)
	CopyTo [▶ 3097]	Copies the data types to the specified array, starting at the array index. (Inherited from DataTypeCollection.T. [▶ 3087].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 3098]	Gets the enumerator. (Inherited from DataTypeCollection.T. [▶ 3087].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)

	Name	Description
	IndexOf [▶ 3098]	Determines the Index of the specified IDataType [▶ 2475] . (Inherited from DataTypeCollection.T. [▶ 3087] .)
	Insert [▶ 3099]	Inserts an IDataType [▶ 2475] into the DataTypeCollection [▶ 2112] . (Inherited from DataTypeCollection.T. [▶ 3087] .)
	LookupType [▶ 3099]	Determines the specified IDataType [▶ 2475] (Inherited from DataTypeCollection.T. [▶ 3087] .)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 3100]	Removes the specified IDataType [▶ 2475] . (Inherited from DataTypeCollection.T. [▶ 3087] .)
	RemoveAt [▶ 3100]	Removes the IDataType [▶ 2475] object at the specified index. (Inherited from DataTypeCollection.T. [▶ 3087] .)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetType [▶ 3101]	Tries to get the specified IDataType [▶ 2475] from the IDataTypeCollection.T. [▶ 2486] . (Inherited from DataTypeCollection.T. [▶ 3087] .)

Reference

[DataTypeCollection Class \[▶ 2112\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.6.3.1 DataTypeCollection.AsReadOnly Method

Returns A ReadOnly-Version of the [DataTypeCollection \[▶ 2112\]](#).

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyDataTypeCollection AsReadOnly()
```

Return Value

Type: [ReadOnlyDataTypeCollection \[▶ 2815\]](#)

A read only version of this [DataTypeCollection \[▶ 2112\]](#).

Reference

[DataTypeCollection Class \[▶ 2112\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.6.3.2 DataTypeCollection.Clone Method

Clones this [DataTypeCollection \[▶ 2112\]](#) (Shallow Copy)

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DataTypeCollection Clone()
```

Return Value

Type: [DataTypeCollection](#) [[▶ 2112](#)]

A clone of this [DataTypeCollection](#) [[▶ 2112](#)].

Reference


[DataTypeCollection Class](#) [[▶ 2112](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.6.4 DataTypeCollection Fields

The [DataTypeCollection](#) [[▶ 2112](#)] type exposes the following members.

Fields

	Name	Description
	list [▶ 3102]	Internal list of data types (Inherited from DataTypeCollection.T. [▶ 3087].)

Reference

[DataTypeCollection Class](#) [[▶ 2112](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.7 DataTypeEventArgs Class

Class [DataTypeEventArgs](#).

Inheritance Hierarchy

System.Object
 System.EventArgs
 TwinCAT.TypeSystem.DataTypeEventArgs

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#


```
public class DataTypeEventArgs : EventArgs
```

The [DataTypeEventArgs](#) type exposes the following members.







Constructors

	Name	Description
	DataTypeEventArgs [▶ 2119]	Initializes a new instance of the DataTypeEventArgs class.

Properties

	Name	Description
	DataTypes [▶ 2120]	The data types

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

System.EventArgs

6.11.7.1 DataTypeEventArgs Constructor

Initializes a new instance of the [DataTypeEventArgs](#) [[▶ 2118](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DataTypeEventArgs(
    IEnumerable<IDataType> types
)
```

Parameters

types	Type: System.Collections.Generic.IEnumerable. IDataType [▶ 2475]. The types.
-------	---

Reference


[DataTypeEventArgs Class](#) [[▶ 2118](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.7.2 DataTypeEventArgs Properties

The [DataTypeEventArgs](#) [[▶ 2118](#)] type exposes the following members.

Properties

	Name	Description
	DataTypes [▶ 2120]	The data types

Reference

[DataTypeEventArgs Class](#) [[▶ 2118](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.7.2.1 DataTypeEventArgs.DataTypes Property

The data types

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public IEnumerable<IDataType> DataTypes { get; }
```

Property Value

Type: [IEnumerable.IDataType](#) [[▶ 2475](#)].

Reference







[DataTypeEventArgs Class](#) [[▶ 2118](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.7.3 DataTypeEventArgs Methods

The [DataTypeEventArgs](#) [[▶ 2118](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[DataTypeEventArgs Class](#) [[▶ 2118](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.8 DataTypeException Class

Data Type Exception

Inheritance Hierarchy

System.Object
 System.Exception
 [TwinCAT.AdsException](#) [▶ 61]
 TwinCAT.TypeSystem.DataTypeException
 [TwinCAT.TypeSystem.CannotResolveDataTypeException](#) [▶ 2106]

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229




Syntax

C#








```
[SerializableAttribute]
public class DataTypeException : AdsException
```




The DataTypeException type exposes the following members.

Constructors









	Name	Description
	DataTypeException(SerializationInfo, StreamingContext) [▶ 2123]	Initializes a new instance of the DataTypeException class.
	DataTypeException(String, String) [▶ 2123]	Initializes a new instance of the DataTypeException class.
	DataTypeException(String, IDataTypes) [▶ 2124]	Initializes a new instance of the DataTypeException class.

Properties


	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	DataType [▶ 2125]	Gets the type of the data.
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)

	Name	Description
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)
	TypeName [▶ 2125]	Gets the name of the type.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData [▶ 2126]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Overrides Exception.GetObjectData(SerializationInfo, StreamingContext).)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Events



	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)


Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.8.1 DataTypeException Constructor

Overload List

	Name	Description
	DataTypeException (SerializationInfo , StreamingContext) [▶ 2123]	Initializes a new instance of the DataTypeException [▶ 2121] class.
	DataTypeException (String , String) [▶ 2123]	Initializes a new instance of the DataTypeException [▶ 2121] class.

	Name	Description
	<code>DataTypeException(String, IDataType)</code> [▶ 2124]	Initializes a new instance of the DataTypeException [▶ 2121] class.

Reference

[DataTypeException Class](#) [▶ 2121]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.8.1.1 DataTypeException Constructor (SerializationInfo, StreamingContext)

Initializes a new instance of the [DataTypeException](#) [▶ 2121] class.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected DataTypeException(
    SerializationInfo serializationInfo,
    StreamingContext streamingContext
)
```

Parameters

serializationInfo	Type: System.Runtime.Serialization.SerializationInfo The serialization information.
streamingContext	Type: System.Runtime.Serialization.StreamingContext The streaming context.

Reference

[DataTypeException Class](#) [▶ 2121]

[DataTypeException Overload](#) [▶ 2122]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.8.1.2 DataTypeException Constructor (String, String)

Initializes a new instance of the [DataTypeException](#) [▶ 2121] class.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DataTypeException(
    string message,
    string type
)
```

Parameters

message	Type: System.String The message.
type	Type: System.String The type.

Reference

[DataTypeException Class \[▸ 2121\]](#)

[DataTypeException Overload \[▸ 2122\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.8.1.3 DataTypeException Constructor (String, IDatatype)

Initializes a new instance of the [DataTypeException \[▸ 2121\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public DataTypeException(
    string message,
    IDatatype type
)
```

Parameters

message	Type: System.String The message.
type	Type: TwinCAT.TypeSystem.IDatatype [▸ 2475] The type.

Reference

[DataTypeException Class \[▸ 2121\]](#)





[DataTypeException Overload \[▸ 2122\]](#)







[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.8.2 DataTypeException Properties

The [DataTypeException \[▸ 2121\]](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	DataType [▸ 2125]	Gets the type of the data.
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	Name	Description
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)
	<u>Type</u> Name [▶ 2125]	Gets the name of the type.

Reference

[DataTypedException Class \[▶ 2121\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.8.2.1 DataTypeException.DataType Property

Gets the type of the data.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataType? DataType { get; }
```

Property Value

Type: [IDataType \[▶ 2475\]](#)

The type of the data.

Reference

[DataTypedException Class \[▶ 2121\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.8.2.2 DataTypeException.TypeName Property

Gets the name of the type.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string TypeName { get; }
```

Property Value









Type: String

The name of the type.

Reference[DataTypeException Class \[► 2121\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.8.3 DataTypeException Methods**

The [DataTypeException \[► 2121\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData [► 2126]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Overrides Exception.GetObjectData(SerializationInfo, StreamingContext).)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Reference[DataTypeException Class \[► 2121\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.8.3.1 DataTypeException.GetObjectData Method**

When overridden in a derived class, sets the SerializationInfo with information about the exception.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public override void GetObjectData(
    SerializationInfo info,
    StreamingContext context
)
```

Parameters

info	Type: System.Runtime.Serialization.SerializationInfo The SerializationInfo that holds the serialized object data about the exception being thrown.
------	---

context	Type: System.Runtime.Serialization.StreamingContext The StreamingContext that contains contextual information about the source or destination.
---------	---

Implements

ISerializable.GetObjectData(SerializationInfo, StreamingContext)

Reference


[DataTypeException Class \[▶ 2121\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.8.4 DataTypeException Events

The [DataTypeException \[▶ 2121\]](#) type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)

Reference

[DataTypeException Class \[▶ 2121\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.9 DataTypeNameEventArgs Class

Class DataTypeNameEventArgs.

Inheritance Hierarchy

System.Object
 System.EventArgs
 TwinCAT.TypeSystem.DataTypeNameEventArgs

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#


```
public class DataTypeNameEventArgs : EventArgs
```

The DataTypeNameEventArgs type exposes the following members.







Constructors

	Name	Description
	DataTypeNameEventArgs [▶ 2128]	Initializes a new instance of the DataTypeNameEventArgs class.

Properties

	Name	Description
	TypeName [▶ 2129]	The type name

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

System.EventArgs

6.11.9.1 DataTypeNameEventArgs Constructor

Initializes a new instance of the [DataTypeNameEventArgs](#) [[▶ 2127](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public DataTypeNameEventArgs(
    string typeName
)
```

Parameters

typeName	Type: System.String Name of the type.
----------	--

Reference


[DataTypeNameEventArgs Class](#) [[▶ 2127](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.9.2 DataTypeNameEventArgs Properties

The [DataTypeNameEventArgs](#) [[▶ 2127](#)] type exposes the following members.

Properties

	Name	Description
	TypeName [▶ 2129]	The type name

Reference

[DataTypeNameEventArgs Class](#) [[▶ 2127](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.9.2.1 DataTypeNameEventArgs.TypeName Property

The type name

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string TypeName { get; }
```

Property Value

Type: String

Reference







[DataTypeNameEventArgs Class](#) [[▶ 2127](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.9.3 DataTypeNameEventArgs Methods

The [DataTypeNameEventArgs](#) [[▶ 2127](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[DataTypeNameEventArgs Class](#) [[▶ 2127](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.10 Dimension Class

Represents a single dimension of an [IArrayType](#) [[▶ 2459](#)]

Inheritance Hierarchy

System.Object

 TwinCAT.TypeSystem.Dimension

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229




Syntax

C#







```
public class Dimension : IDimension
```

The Dimension type exposes the following members.

Properties

	Name	Description
	ElementCount [▶ 2131]	Gets the number of elements within that IDimension [▶ 2490].
	LowerBound [▶ 2131]	Gets the lower bound of elements within that IDimension [▶ 2490].
	UpperBound [▶ 2132]	Gets the upper bound of elements within this Dimension

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)


Reference



[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.10.1 Dimension Properties

The [Dimension](#) [[▶ 2130](#)] type exposes the following members.

Properties

	Name	Description
	ElementCount [▶ 2131]	Gets the number of elements within that IDimension [▶ 2490].

	Name	Description
	LowerBound [▶ 2131]	Gets the lower bound of elements within that IDimension [▶ 2490].
	UpperBound [▶ 2132]	Gets the upper bound of elements within this Dimension [▶ 2130]

Reference

[Dimension Class](#) [[▶ 2130](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.10.1.1 Dimension.ElementCount Property

Gets the number of elements within that [IDimension](#) [[▶ 2490](#)].

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ElementCount { get; }
```

Property Value

Type: Int32

The element count.

Implements

[IDimension.ElementCount](#) [[▶ 2491](#)]

Reference

[Dimension Class](#) [[▶ 2130](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.10.1.2 Dimension.LowerBound Property

Gets the lower bound of elements within that [IDimension](#) [[▶ 2490](#)].

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int LowerBound { get; }
```

Property Value

Type: Int32

The lower bound.

Implements[IDimension.LowerBound](#) [[▶ 2491](#)]**Reference**[Dimension Class](#) [[▶ 2130](#)][TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]**6.11.10.1.3 Dimension.UpperBound Property**Gets the upper bound of elements within this [Dimension](#) [[▶ 2130](#)]**Namespace:** [TwinCAT.TypeSystem](#) [[▶ 2083](#)]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**







```
public int UpperBound { get; }
```

Property Value

Type: Int32

The upper bound.

Reference[Dimension Class](#) [[▶ 2130](#)][TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]**6.11.10.2 Dimension Methods**The [Dimension](#) [[▶ 2130](#)] type exposes the following members.**Methods**

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference[Dimension Class](#) [[▶ 2130](#)][TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.11 DimensionCollection Class

Collection class for Array Dimensions

Inheritance Hierarchy

System.Object

TwinCAT.TypeSystem.DimensionCollection

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229





Syntax

C#








```
public class DimensionCollection : IDimensionCollection,
    IList<IDimension>, ICollection<IDimension>, IEnumerable<IDimension>,
    IEnumerable
```


The DimensionCollection type exposes the following members.

Constructors


















	Name	Description
	DimensionCollection n. [▶ 2135]	Initializes a new instance of the DimensionCollection class.
	DimensionCollection n(IEnumerable.IDimension.) [▶ 2136]	Initializes a new instance of the DimensionCollection class.
	DimensionCollection n(Int32) [▶ 2135]	Initializes a new instance of an 1-Dimensional representing DimensionCollection class.
	DimensionCollection n(.Int32.) [▶ 2136]	Initializes a new instance of the DimensionCollection class.

Properties

	Name	Description
	Count [▶ 2137]	Gets the number of elements contained in the ICollection.T..
	ElementCount [▶ 2138]	Gets the Number of elements in all Dimensions
	Empty [▶ 2140]	Gets an empty DimensionCollection
	IsNonZeroBased [▶ 2141]	Gets a value indicating whether this instance is non zero index based.
	IsReadOnly [▶ 2138]	Gets a value indicating whether the ICollection.T. is read-only.
	Item [▶ 2139]	Gets or sets the element at the specified index.
	LowerBounds [▶ 2139]	Gets the lower bounds.

	Name	Description
	UpperBounds [▶ 2140]	Gets the upper bounds.

Methods

	Name	Description
	Add [▶ 2142]	Adds an item to the ICollection.T..
	AsReadOnly [▶ 2143]	Returns a read only version of this DimensionCollection.
	Clear [▶ 2143]	Removes all items from the ICollection.T..
	Contains [▶ 2144]	Determines whether the ICollection.T. contains a specific value.
	CopyTo [▶ 2144]	Copies to.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetDimensionLengths [▶ 2145]	Gets an array the specifies the Lengths of each Array Dimension
	GetEnumerator [▶ 2146]	Returns an enumerator that iterates through the collection.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 2146]	Determines the index of a specific item in the IList.T..
	Insert [▶ 2147]	Inserts an item to the IList.T. at the specified index.
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 2148]	Removes the first occurrence of a specific object from the ICollection.T..
	RemoveAt [▶ 2148]	Removes the IList.T. item at the specified index.
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Extension Methods





	Name	Description
	AddDimension [▶ 2982]	Adds a Dimension (FluentInterface) (Defined by IDimensionCollectionExtension [▶ 2981] .)

Reference

[TwinCAT.TypeSystem Namespace](#) [\[▶ 2083\]](#)

6.11.11.1 DimensionCollection Constructor

Overload List

	Name	Description
	DimensionCollection n. [2135]	Initializes a new instance of the DimensionCollection [2133] class.
	DimensionCollection (IEnumerable.IDimension.) [2136]	Initializes a new instance of the DimensionCollection [2133] class.
	DimensionCollection (Int32) [2135]	Initializes a new instance of an 1-Dimensional representing DimensionCollection [2133] class.
	DimensionCollection (.Int32.) [2136]	Initializes a new instance of the DimensionCollection [2133] class.

Reference

[DimensionCollection Class](#) [[2133](#)]

[TwinCAT.TypeSystem Namespace](#) [[2083](#)]

6.11.11.1.1 DimensionCollection Constructor

Initializes a new instance of the [DimensionCollection](#) [[2133](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DimensionCollection()
```

Reference

[DimensionCollection Class](#) [[2133](#)]

[DimensionCollection Overload](#) [[2135](#)]

[TwinCAT.TypeSystem Namespace](#) [[2083](#)]

6.11.11.1.2 DimensionCollection Constructor (Int32)

Initializes a new instance of an 1-Dimensional representing [DimensionCollection](#) [[2133](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DimensionCollection(  
    int length  
)
```

Parameters

length	Type: System.Int32 The length.
--------	-----------------------------------

Reference

[DimensionCollection Class](#) [[▶ 2133](#)]

[DimensionCollection Overload](#) [[▶ 2135](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.11.1.3 DimensionCollection Constructor (.Int32.)

Initializes a new instance of the [DimensionCollection](#) [[▶ 2133](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public DimensionCollection(
    int[] dimLengths
)
```

Parameters

dimLengths	Type: .System.Int32. The dim lengths.
------------	--

Reference

[DimensionCollection Class](#) [[▶ 2133](#)]

[DimensionCollection Overload](#) [[▶ 2135](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.11.1.4 DimensionCollection Constructor (IEnumerable.IDimension.)

Initializes a new instance of the [DimensionCollection](#) [[▶ 2133](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public DimensionCollection(
    IEnumerable<IDimension> coll
)
```

Parameters

coll	Type: System.Collections.Generic.IEnumerable. IDimension [▶ 2490]. The coll.
------	---

Reference

[DimensionCollection Class \[▸ 2133\]](#)









[DimensionCollection Overload \[▸ 2135\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.11.2 DimensionCollection Properties

The [DimensionCollection \[▸ 2133\]](#) type exposes the following members.

Properties

	Name	Description
	Count [▸ 2137]	Gets the number of elements contained in the ICollection.T..
	ElementCount [▸ 2138]	Gets the Number of elements in all Dimensions
	Empty [▸ 2140]	Gets an empty DimensionCollection [▸ 2133]
	IsNonZeroBased [▸ 2141]	Gets a value indicating whether this instance is non zero index based.
	IsReadOnly [▸ 2138]	Gets a value indicating whether the ICollection.T. is read-only.
	Item [▸ 2139]	Gets or sets the element at the specified index.
	LowerBounds [▸ 2139]	Gets the lower bounds.
	UpperBounds [▸ 2140]	Gets the upper bounds.

Reference

[DimensionCollection Class \[▸ 2133\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.11.2.1 DimensionCollection.Count Property

Gets the number of elements contained in the ICollection.T..

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Count { get; }
```

Property Value

Type: Int32
The count.

Implements

ICollection.T..Count

Exceptions

Exception	Condition
NotImplementedException	

Reference

[DimensionCollection Class](#) [► 2133]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.11.2.2 DimensionCollection.ElementCount Property

Gets the Number of elements in all Dimensions

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ElementCount { get; }
```

Property Value

Type: Int32

Implements

[ICollection.ElementCount](#) [► 2494]

Reference

[DimensionCollection Class](#) [► 2133]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.11.2.3 DimensionCollection.IsReadOnly Property

Gets a value indicating whether the ICollection.T. is read-only.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsReadOnly { get; }
```

Property Value

Type: Boolean

true if this instance is read only; otherwise, false.

Implements

ICollection.T..IsReadOnly

Reference

[DimensionCollection Class](#) [► 2133]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.11.2.4 DimensionCollection.Item Property

Gets or sets the element at the specified index.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDimension this[
    int index
] { get; set; }
```

Parameters

index	Type: System.Int32 The index.
-------	----------------------------------

Return Value

Type: [IDimension](#) [► 2490]

IDimension.

Implements

ICollection.T..Item.Int32.

Exceptions

Exception	Condition
NotSupportedException	

Remarks

Index = 0 is the lowest dimesion.

Reference

[DimensionCollection Class](#) [► 2133]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.11.2.5 DimensionCollection.LowerBounds Property

Gets the lower bounds.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int[] LowerBounds { get; }
```

Property Value

Type: .Int32.
The lower bounds.

Implements

[IDimensionCollection.LowerBounds](#) [[▶ 2494](#)]

Reference

[DimensionCollection Class](#) [[▶ 2133](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.11.2.6 DimensionCollection.UpperBounds Property

Gets the upper bounds.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int[] UpperBounds { get; }
```

Property Value

Type: .Int32.
The upper bounds.

Implements

[IDimensionCollection.UpperBounds](#) [[▶ 2494](#)]

Reference

[DimensionCollection Class](#) [[▶ 2133](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.11.2.7 DimensionCollection.Empty Property

Gets an empty [DimensionCollection](#) [[▶ 2133](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static DimensionCollection Empty { get; }
```

Property Value

Type: [DimensionCollection](#) [[▶ 2133](#)]
 The Empty collectio.

Reference

- [DimensionCollection Class](#) [[▶ 2133](#)]
- [TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.11.2.8 DimensionCollection.IsNonZeroBased Property

Gets a value indicating whether this instance is non zero index based.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]
Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsNonZeroBased { get; }
```

Property Value

Type: Boolean
 true if this instance is non zero indexed based; otherwise, false.

Implements

[IDimensionCollection.IsNonZeroBased](#) [[▶ 2495](#)]








Reference











- [DimensionCollection Class](#) [[▶ 2133](#)]
- [TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.11.3 DimensionCollection Methods

The [DimensionCollection](#) [[▶ 2133](#)] type exposes the following members.

Methods

	Name	Description
	Add [▶ 2142]	Adds an item to the ICollection.T..
	AsReadOnly [▶ 2143]	Returns a read only version of this DimensionCollection [▶ 2133].
	Clear [▶ 2143]	Removes all items from the ICollection.T..
	Contains [▶ 2144]	Determines whether the ICollection.T. contains a specific value.
	CopyTo [▶ 2144]	Copies to.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)

	Name	Description
	GetDimensionLengths [▶ 2145]	Gets an array the specifies the Lengths of each Array Dimension
	GetEnumerator [▶ 2146]	Returns an enumerator that iterates through the collection.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 2146]	Determines the index of a specific item in the IList.T..
	Insert [▶ 2147]	Inserts an item to the IList.T. at the specified index.
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 2148]	Removes the first occurrence of a specific object from the ICollection.T..
	RemoveAt [▶ 2148]	Removes the IList.T. item at the specified index.
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Extension Methods

	Name	Description
	AddDimension [▶ 2982]	Adds a Dimension (FluentInterface) (Defined by IDimensionCollectionExtension [▶ 2981].)

Reference

[DimensionCollection Class](#) [[▶ 2133](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.11.3.1 DimensionCollection.Add Method

Adds an item to the ICollection.T..

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Add(
    IDimension item
)
```

Parameters

item	Type: TwinCAT.TypeSystem.IDimension [▶ 2490] The object to add to the ICollection.T..
------	--

Implements

ICollection.T..Add(T)

Exceptions

Exception	Condition
NotImplementedException	

Reference

[DimensionCollection Class](#) [► 2133]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.11.3.2 DimensionCollection.AsReadOnly Method

Returns a read only version of this [DimensionCollection](#) [► 2133].

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public ReadOnlyDimensionCollection AsReadOnly()
```

Field Value

Type: [ReadOnlyDimensionCollection](#) [► 2818]

As read only.

Return Value

Type: [ReadOnlyDimensionCollection](#) [► 2818]

ReadOnlyDimensionCollection.

Reference

[DimensionCollection Class](#) [► 2133]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.11.3.3 DimensionCollection.Clear Method

Removes all items from the ICollection.T..

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public void Clear()
```

Implements

ICollection.T..Clear.

Exceptions

Exception	Condition
NotImplementedException	

Reference[DimensionCollection Class \[► 2133\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.11.3.4 DimensionCollection.Contains Method**

Determines whether the ICollection.T. contains a specific value.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public bool Contains(
    IDimension item
)
```

Parameters

item	Type: TwinCAT.TypeSystem.IDimension [► 2490] The object to locate in the ICollection.T..
------	---

Return Value

Type: Boolean
true if item is found in the ICollection.T.; otherwise, false.

Implements

ICollection.T..Contains(T)

Exceptions

Exception	Condition
NotImplementedException	

Reference[DimensionCollection Class \[► 2133\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.11.3.5 DimensionCollection.CopyTo Method**

Copies to.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public void CopyTo(
    IDimension[] array,
    int arrayIndex
)
```


Parameters

array	Type: .TwinCAT.TypeSystem.IDimension [▶ 2490]. The array.
arrayIndex	Type: System.Int32 Index of the array.

Implements

[ICollection.T..CopyTo\(.T., Int32\)](#)

Exceptions

Exception	Condition
NotImplementedException	

Reference

[DimensionCollection Class](#) [[▶ 2133](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.11.3.6 DimensionCollection.Empty Method

Gets an empty [DimensionCollection](#) [[▶ 2133](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax**C#**

```
public static DimensionCollection Empty()
```

Field Value

Type: [DimensionCollection](#) [[▶ 2133](#)]

The Empty collectio.

Reference

[DimensionCollection Class](#) [[▶ 2133](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.11.3.7 DimensionCollection.GetDimensionLengths Method

Gets an array the specifies the Lengths of each Array Dimension

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public int[] GetDimensionLengths()
```

Return Value

Type: `.Int32`.
`System.Int32[]`.

Implements

[IDimensionCollection.GetDimensionLengths. \[▸ 2496\]](#)

Reference

[DimensionCollection Class \[▸ 2133\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.11.3.8 DimensionCollection.GetEnumerator Method

Returns an enumerator that iterates through the collection.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public IEnumerator<IDimension> GetEnumerator()
```

Return Value

Type: `IEnumerator.IDimension` [\[▸ 2490\]](#).
A `IEnumerator.T`. that can be used to iterate through the collection.

Implements

`IEnumerable.T`..GetEnumerator.

Exceptions

Exception	Condition
NotImplementedException	

Reference

[DimensionCollection Class \[▸ 2133\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.11.3.9 DimensionCollection.IndexOf Method

Determines the index of a specific item in the `IList.T`..

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int IndexOf(
    IDimension item
)
```

Parameters

item	Type: TwinCAT.TypeSystem.IDimension [▶ 2490] The object to locate in the IList.T..
------	---

Return Value

Type: Int32
The index of item if found in the list; otherwise, -1.

Implements

IList.T..IndexOf(T)

Exceptions

Exception	Condition
NotImplementedException	

Reference

[DimensionCollection Class](#) [▶ 2133]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.11.3.10 DimensionCollection.Insert Method

Inserts an item to the IList.T. at the specified index.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Insert(
    int index,
    IDimension item
)
```

Parameters

index	Type: System.Int32 The zero-based index at which item should be inserted.
item	Type: TwinCAT.TypeSystem.IDimension [▶ 2490] The object to insert into the IList.T..

Implements

IList.T..Insert(Int32, T)

Exceptions

Exception	Condition
NotImplementedException	

Reference

[DimensionCollection Class](#) [► 2133]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.11.3.11 DimensionCollection.Remove Method

Removes the first occurrence of a specific object from the ICollection.T..

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public bool Remove(
    IDimension item
)
```

Parameters

item	Type: TwinCAT.TypeSystem.IDimension [► 2490] The object to remove from the ICollection.T..
------	---

Return Value

Type: Boolean

true if item was successfully removed from the ICollection.T.; otherwise, false. This method also returns false if item is not found in the original ICollection.T..

Implements

ICollection.T..Remove(T)

Exceptions

Exception	Condition
NotImplementedException	

Reference

[DimensionCollection Class](#) [► 2133]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.11.3.12 DimensionCollection.RemoveAt Method

Removes the IList.T. item at the specified index.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void RemoveAt(
    int index
)
```

Parameters

index	Type: System.Int32 The zero-based index of the item to remove.
-------	---

Implements

ICollection.RemoveAt(Int32)

Exceptions

Exception	Condition
NotSupportedException	

Reference

[DimensionCollection Class](#) [► 2133]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.12 DynamicAliasInstance Class

Class DynamicAliasInstance. This class cannot be inherited.

Inheritance Hierarchy

System.Object
 System.Dynamic.DynamicObject
 [TwinCAT.TypeSystem.DynamicSymbol](#) [► 2266]
 TwinCAT.TypeSystem.DynamicAliasInstance

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229




Syntax





















C#

```
public sealed class DynamicAliasInstance : DynamicSymbol,
    IAliasInstance, ISymbol, IAttributedInstance, IInstance, IBitSize
```

The DynamicAliasInstance type exposes the following members.

Properties


















	Name	Description
	InnerSymbol [► 2274]	Gets the inner symbol of this DynamicSymbol [► 2266] (Inherited from DynamicSymbol [► 2266].)
	AccessRights [► 2274]	Gets the access rights. (Inherited from DynamicSymbol [► 2266].)
	Attributes [► 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [► 2266].)






	Name	Description
	BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from DynamicSymbol [▶ 2266] .)
	ByteSize [▶ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [▶ 2266] .)
	Category [▶ 2277]	Gets the category. (Inherited from DynamicSymbol [▶ 2266] .)
	Comment [▶ 2277]	Gets the comment of the Instance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266] .)
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266] .)
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549] . (Inherited from DynamicSymbol [▶ 2266] .)
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances, what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266] .)
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266] .)
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266] .)
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266] .)
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DynamicSymbol [▶ 2266] .)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from DynamicSymbol [▶ 2266] .)
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266] .)
	IsPointer [▶ 2282]	Indicates that the Instance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266] .)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266] .)
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266] .)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266] .)
	IsReference [▶ 2284]	Indicates that the Instance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266] .)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266] .)
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266] .)
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266] .)
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266] .)
	Size [▶ 2286]	Gets the size of the Instance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266] .)

	Name	Description
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)



Methods

	Name	Description
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)
	GetDynamicMemberNames [▶ 2157]	Returns the enumeration of all dynamic member names. (Overrides DynamicSymbol.GetDynamicMemberNames . [▶ 2293].)
	GetHashCode [▶ 2293]	Gets the HashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue . [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValueAsync [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue . [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject .)

	Name	Description
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetIndex [▶ 2157]	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Overrides DynamicObject.TryGetIndex(GetIndexBinder, .Object., Object.))
	TryGetMember [▶ 2158]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Overrides DynamicSymbol.TryGetMember(GetMemberBinder, Object.) [▶ 2308].)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject.)
	TryInvokeMember	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicObject.)
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	TrySetIndex [▶ 2159]	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Overrides DynamicObject.TrySetIndex(SetIndexBinder, .Object., Object.))
	TrySetMember [▶ 2159]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Overrides DynamicObject.TrySetMember(SetMemberBinder, Object.))
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
 	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from DynamicSymbol [▶ 2266].)
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte e., Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	WriteRawValueAsync [2315]	Writes the raw value of the IValueSymbol [2775] (Ads Read / Write) (Inherited from DynamicSymbol [2266].)
	WriteValue(Object) [2316]	Writes the specified value to the DynamicSymbol [2266]. (Inherited from DynamicSymbol [2266].)
 	WriteValue(Object, Int32) [2319]	Writes the specified value to the DynamicSymbol [2266]. (Inherited from DynamicSymbol [2266].)
	WriteValueAsync [2321]	Writes the Value of the IValueSymbol [2775] (Inherited from DynamicSymbol [2266].)

Events

	Name	Description
	RawValueChanged [2323]	Occurs when the RawValue of the IValueSymbol [2775] has changed. (Inherited from DynamicSymbol [2266].)
	ValueChanged [2323]	Occurs when the (Primitive) value of the IValueSymbol [2775] has changed. (Inherited from DynamicSymbol [2266].)

Extension Methods

	Name	Description
	ReferencesExternalData [3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [[2083](#)]







[TwinCAT.TypeSystem.DynamicSymbol](#) [[2266](#)]

[TwinCAT.TypeSystem.IAliasInstance](#) [[2433](#)]

6.11.12.1 DynamicAliasInstance Properties

The [DynamicAliasInstance](#) [[2149](#)] type exposes the following members.

Properties

	Name	Description
	InnerSymbol [2274]	Gets the inner symbol of this DynamicSymbol [2266] (Inherited from DynamicSymbol [2266].)
	AccessRights [2274]	Gets the access rights. (Inherited from DynamicSymbol [2266].)
	Attributes [2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [2266].)
	BitSize [2276]	Gets the size of the IDataType [2475] in bits. (Inherited from DynamicSymbol [2266].)
	ByteSize [2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [2266].)
	Category [2277]	Gets the category. (Inherited from DynamicSymbol [2266].)

	Name	Description
	Comment [▶ 2277]	Gets the comment of the Instance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266].)
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances, what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266].)
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266].)
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DynamicSymbol [▶ 2266].)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is a container type. (Inherited from DynamicSymbol [▶ 2266].)
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266].)
	IsPointer [▶ 2282]	Indicates that the Instance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266].)
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266].)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266].)
	IsReference [▶ 2284]	Indicates that the Instance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
	Size [▶ 2286]	Gets the size of the Instance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)

Reference
















[DynamicAliasInstance Class \[▶ 2149\]](#)

















[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)






6.11.12.2 DynamicAliasInstance Methods

The [DynamicAliasInstance \[▶ 2149\]](#) type exposes the following members.

Methods

	Name	Description
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)
	GetDynamicMemberNames [▶ 2157]	Returns the enumeration of all dynamic member names. (Overrides DynamicSymbol.GetDynamicMemberNames. [▶ 2293].)
	GetHashCode [▶ 2293]	Gets the GetHashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue. [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValueAsync [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject.)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject.)

	Name	Description
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetIndex [▶ 2157]	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Overrides DynamicObject.TryGetIndex(GetIndexBinder, .Object., Object.) .)
	TryGetMember [▶ 2158]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Overrides DynamicSymbol.TryGetMember(GetMemberBinder, Object.) [▶ 2308].)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject.)
	TryInvokeMember	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicObject.)
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	TrySetIndex [▶ 2159]	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Overrides DynamicObject.TrySetIndex(SetIndexBinder, .Object., Object.) .)
	TrySetMember [▶ 2159]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Overrides DynamicObject.TrySetMember(SetMemberBinder, Object.) .)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
 	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from DynamicSymbol [▶ 2266].)
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte e., Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[DynamicAliasInstance Class](#) [[▶ 2149](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.12.2.1 DynamicAliasInstance.GetDynamicMemberNames Method

Returns the enumeration of all dynamic member names.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override IEnumerable<string> GetDynamicMemberNames()
```

Return Value

Type: [IEnumerable.String](#).

A sequence that contains dynamic member names.

Reference

[DynamicAliasInstance Class](#) [[▶ 2149](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.12.2.2 DynamicAliasInstance.TryGetIndex Method

Provides the implementation for operations that get a value by index. Classes derived from the [DynamicObject](#) class can override this method to specify dynamic behavior for indexing operations.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TryGetIndex(
    GetIndexBinder binder,
    Object[] indexes,
    out Object?? result
)
```

Parameters

binder	Type: System.Dynamic.GetIndexBinder Provides information about the operation.
indexes	Type: System.Object. The indexes that are used in the operation. For example, for the sampleObject[3] operation in C# (sampleObject(3) in Visual Basic), where sampleObject is derived from the DynamicObject class, indexes[0] is equal to 3.
result	Type: System.Object. The result of the index operation.

Return Value

Type: Boolean

true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a run-time exception is thrown.)

Reference

[DynamicAliasInstance Class \[► 2149\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.12.2.3 DynamicAliasInstance.TryGetMember Method

Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TryGetMember(
    GetMemberBinder binder,
    out Object?? result
)
```

Parameters

binder	Type: System.Dynamic.GetMemberBinder Provides information about the object that called the dynamic operation. The binder.Name property provides the name of the member on which the dynamic operation is performed. For example, for the Console.WriteLine(sampleObject.SampleProperty) statement, where sampleObject is an instance of the class derived from the DynamicObject class, binder.Name returns "SampleProperty". The binder.IgnoreCase property specifies whether the member name is case-sensitive.
result	Type: System.Object. The result of the get operation. For example, if the method is called for a property, you can assign the property value to result.

Return Value

Type: Boolean
 true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a run-time exception is thrown.)

Reference

[DynamicAliasInstance Class \[► 2149\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.12.2.4 DynamicAliasInstance.TrySetIndex Method

Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TrySetIndex(
    SetIndexBinder binder,
    Object[] indexes,
    Object? value
)
```

Parameters

binder	Type: System.Dynamic.SetIndexBinder Provides information about the operation.
indexes	Type: .System.Object. The indexes that are used in the operation. For example, for the sampleObject[3] = 10 operation in C# (sampleObject(3) = 10 in Visual Basic), where sampleObject is derived from the DynamicObject class, indexes[] is equal to 3.
value	Type: System.Object The value to set to the object that has the specified index. For example, for the sampleObject[3] = 10 operation in C# (sampleObject(3) = 10 in Visual Basic), where sampleObject is derived from the DynamicObject class, value is equal to 10.

Return Value

Type: Boolean
 true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a language-specific run-time exception is thrown.)

Reference

[DynamicAliasInstance Class \[► 2149\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.12.2.5 DynamicAliasInstance.TrySetMember Method

Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TrySetMember(
    SetMemberBinder binder,
    Object? value
)
```

Parameters

binder	Type: System.Dynamic.SetMemberBinder Provides information about the object that called the dynamic operation. The binder.Name property provides the name of the member to which the value is being assigned. For example, for the statement sampleObject.SampleProperty = "Test", where sampleObject is an instance of the class derived from the DynamicObject class, binder.Name returns "SampleProperty". The binder.IgnoreCase property specifies whether the member name is case-sensitive.
value	Type: System.Object The value to set to the member. For example, for sampleObject.SampleProperty = "Test", where sampleObject is an instance of the class derived from the DynamicObject class, the value is "Test".

Return Value

Type: Boolean

true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a language-specific run-time exception is thrown.)

Reference



[DynamicAliasInstance Class](#) [► 2149]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.12.3 DynamicAliasInstance Events

The [DynamicAliasInstance](#) [► 2149] type exposes the following members.

Events

	Name	Description
	RawValueChanged [► 2323]	Occurs when the RawValue of the IValueSymbol [► 2775] has changed. (Inherited from DynamicSymbol [► 2266].)
	ValueChanged [► 2323]	Occurs when the (Primitive) value of the IValueSymbol [► 2775] has changed. (Inherited from DynamicSymbol [► 2266].)

Reference

[DynamicAliasInstance Class](#) [► 2149]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.13 DynamicArrayInstance Class

Dynamic Array Instance

Inheritance Hierarchy

System.Object
 System.Dynamic.DynamicObject
 TwinCAT.TypeSystem.DynamicSymbol [▶ 2266]
 TwinCAT.TypeSystem.DynamicArrayInstance
 TwinCAT.TypeSystem.DynamicOversamplingArrayInstance [▶ 2176]

Namespace: TwinCAT.TypeSystem [▶ 2083]
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

















Syntax

C#

```
public class DynamicArrayInstance : DynamicSymbol,
    IArrayInstance, ISymbol, IAttributedInstance, IInstance, IBitSize
```




The DynamicArrayInstance type exposes the following members.






















Properties










	Name	Description
	_InnerSymbol [▶ 2274]	Gets the inner symbol of this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	AccessRights [▶ 2274]	Gets the access rights. (Inherited from DynamicSymbol [▶ 2266].)
	AllowGIOAccess [▶ 2275]	Indicates, that the aggregates symbols is an IProcessImageAddress (and most probably IAdsSymbol) (Inherited from DynamicSymbol [▶ 2266].)
	Attributes [▶ 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [▶ 2266].)
	BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from DynamicSymbol [▶ 2266].)
	ByteSize [▶ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [▶ 2266].)
	Category [▶ 2277]	Gets the category. (Inherited from DynamicSymbol [▶ 2266].)
	Comment [▶ 2277]	Gets the comment of the IInstance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266].)
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	Dimensions [▶ 2167]	Gets the dimensions as read only collection.
	Elements [▶ 2168]	Gets the contained Array Elements as read only collection.
	ElementType [▶ 2168]	Gets the type of the contained elements.
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances, what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266].)
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266].)














	Name	Description
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266].)
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DynamicSymbol [▶ 2266].)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is a container type. (Inherited from DynamicSymbol [▶ 2266].)
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266].)
	IsPointer [▶ 2282]	Indicates that the Instance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266].)
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266].)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266].)
	IsReference [▶ 2284]	Indicates that the Instance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)
	Item [▶ 2169]	Gets the ISymbol [▶ 2691] with the specified indices.
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
	Size [▶ 2286]	Gets the size of the Instance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)

Methods



	Name	Description
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetDynamicMemberNames [▶ 2293]	Returns the enumeration of all dynamic member names. (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	GetHashCode [▶ 2293]	Gets the HashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	OnReadAnyValue [▶ 2293]	Handler function for reading ADS 'Any' Values. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadRawValue [▶ 2294]	Handler function for reading Raw symbol value. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadRawValueAsync [▶ 2294]	Handler function reading the raw value of the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadValue [▶ 2295]	Handler function for the (Inherited from DynamicSymbol [▶ 2266].)
	OnReadValueAsync [▶ 2296]	Handler function reading the DynamicSymbols [▶ 2266] value asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	OnSetInstanceName [▶ 2296]	Sets a new InstanceName InstancePath (Inherited from DynamicSymbol [▶ 2266].)
	OnTryReadValue [▶ 2297]	Handler function for the (Inherited from DynamicSymbol [▶ 2266].)
	OnTryWriteValue [▶ 2297]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteRawValue [▶ 2298]	Handler function for reading symbols raw value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteRawValueAsync [▶ 2298]	Handler function for writing the raw DynamicSymbol [▶ 2266] value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteValue [▶ 2299]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteValueAsync [▶ 2300]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue. [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValueAsync [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)


	Name	Description
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266] . (Inherited from DynamicSymbol [▶ 2266] .)
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266] .)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266] .)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject .)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryGetElement(IList.. Int32..., ISymbol.) [▶ 2173]	Tries to get the array element with the specified indices (jagged array support).
	TryGetElement(.Int3 2., ISymbol.) [▶ 2174]	Tries to get the array element
	TryGetIndex [▶ 2174]	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Overrides DynamicObject.TryGetIndex(GetIndexBinder, .Object., Object.) .)
	TryGetMember [▶ 2308]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Inherited from DynamicSymbol [▶ 2266] .)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject .)
	TryInvokeMember	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicObject .)
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266] .)
	TrySetIndex [▶ 2175]	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Overrides DynamicObject.TrySetIndex(SetIndexBinder, .Object., Object.) .)

	Name	Description
	TrySetMember	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicObject.)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
 	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from DynamicSymbol [▶ 2266].)
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte, Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)

Events

	Name	Description
	RawValueChanged [▶ 2323]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)
	ValueChanged [▶ 2323]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)

Fields

	Name	Description
	syncObject [▶ 2327]	Synchronization object (Inherited from DynamicSymbol [▶ 2266].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)





















Reference















[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.13.1 DynamicArrayInstance Properties

The [DynamicArrayInstance](#) [▶ 2160] type exposes the following members.

Properties

	Name	Description
	InnerSymbol [▶ 2274]	Gets the inner symbol of this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	AccessRights [▶ 2274]	Gets the access rights. (Inherited from DynamicSymbol [▶ 2266].)
	AllowIOAccess [▶ 2275]	Indicates, that the aggregates symbols is an IProcessImageAddress (and most probably IAdsSymbol) (Inherited from DynamicSymbol [▶ 2266].)
	Attributes [▶ 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [▶ 2266].)
	BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from DynamicSymbol [▶ 2266].)
	ByteSize [▶ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [▶ 2266].)
	Category [▶ 2277]	Gets the category. (Inherited from DynamicSymbol [▶ 2266].)
	Comment [▶ 2277]	Gets the comment of the IInstance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266].)
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	Dimensions [▶ 2167]	Gets the dimensions as read only collection.
	Elements [▶ 2168]	Gets the contained Array Elements as read only collection.
	ElementType [▶ 2168]	Gets the type of the contained elements.
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances , what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266].)
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266].)
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DynamicSymbol [▶ 2266].)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from DynamicSymbol [▶ 2266].)
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266].)
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266].)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266].)
	IsReference [▶ 2284]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)
	Item [▶ 2169]	Gets the ISymbol [▶ 2691] with the specified indices.
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
	Size [▶ 2286]	Gets the size of the IInstance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)

Reference

[DynamicArrayInstance Class](#) [[▶ 2160](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.13.1.1 DynamicArrayInstance.Dimensions Property

Gets the dimensions as read only collection.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDimensionCollection Dimensions { get; }
```

Property Value

Type: [IDimensionCollection](#) [[▶ 2492](#)]

The dimensions.

Implements

[IArrayInstance.Dimensions](#) [[▶ 2456](#)]

Reference

[DynamicArrayInstance Class](#) [[▶ 2160](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.13.1.2 DynamicArrayInstance.Elements Property

Gets the contained Array Elements as read only collection.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISymbolCollection<ISymbol> Elements { get; }
```

Property Value

Type: [ISymbolCollection](#) [[▶ 2700](#)].[ISymbol](#) [[▶ 2691](#)].

The elements.

Implements

[IArrayInstance.Elements](#) [[▶ 2456](#)]

Reference

[DynamicArrayInstance Class](#) [[▶ 2160](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.13.1.3 DynamicArrayInstance.ElementType Property

Gets the type of the contained elements.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataType? ElementType { get; }
```

Property Value

Type: [IDataType](#) [[▶ 2475](#)]

The type of the element.

Implements

[IArrayInstance.ElementType](#) [[▶ 2457](#)]

Reference

[DynamicArrayInstance Class](#) [▶ 2160]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.13.1.4 DynamicArrayInstance.Item Property

Gets the [ISymbol](#) [▶ 2691] with the specified indices.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISymbol this[
    int[] indices
] { get; }
```

Parameters

indices	Type: .System.Int32. The indices.
---------	--------------------------------------

Return Value

Type: [ISymbol](#) [▶ 2691]

ISymbol.

Implements

[IArrayInstance.Item..Int32..](#) [▶ 2457]

Exceptions

Exception	Condition
ArgumentOutOfRangeException	indices

Reference



[DynamicArrayInstance Class](#) [▶ 2160]






















[TwinCAT.TypeSystem Namespace](#) [▶ 2083]


















6.11.13.2 DynamicArrayInstance Methods














The [DynamicArrayInstance](#) [▶ 2160] type exposes the following members.

Methods

	Name	Description
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)

	Name	Description
	GetDynamicMemberNames [▶ 2293]	Returns the enumeration of all dynamic member names. (Inherited from DynamicSymbol [▶ 2266].)
	GetHashCode [▶ 2293]	Gets the HashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object .)
	OnReadAnyValue [▶ 2293]	Handler function for reading ADS 'Any' Values. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadRawValue [▶ 2294]	Handler function for reading Raw symbol value. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadRawValueAsync [▶ 2294]	Handler function reading the raw value of the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadValue [▶ 2295]	Handler function for the (Inherited from DynamicSymbol [▶ 2266].)
	OnReadValueAsync [▶ 2296]	Handler function reading the DynamicSymbols [▶ 2266] value asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	OnSetInstanceName [▶ 2296]	Sets a new InstanceName InstancePath (Inherited from DynamicSymbol [▶ 2266].)
	OnTryReadValue [▶ 2297]	Handler function for the (Inherited from DynamicSymbol [▶ 2266].)
	OnTryWriteValue [▶ 2297]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteRawValue [▶ 2298]	Handler function for reading symbols raw value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteRawValueAsync [▶ 2298]	Handler function for writing the raw DynamicSymbol [▶ 2266] value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteValue [▶ 2299]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteValueAsync [▶ 2300]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue. [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValueAsync [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject .)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryGetElement(IList.. Int32.., ISymbol.) [▶ 2173]	Tries to get the array element with the specified indices (jagged array support).
	TryGetElement(Int32.., ISymbol.) [▶ 2174]	Tries to get the array element
	TryGetIndex [▶ 2174]	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Overrides DynamicObject.TryGetIndex(GetIndexBinder, .Object., Object.) .)
	TryGetMember [▶ 2308]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Inherited from DynamicSymbol [▶ 2266].)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject .)
	TryInvokeMember	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicObject .)
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	TrySetIndex [▶ 2175]	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Overrides DynamicObject.TrySetIndex(SetIndexBinder, .Object., Object.) .)

	Name	Description
	TrySetMember	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicObject.)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
 	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from DynamicSymbol [▶ 2266].)
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte, Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)



Reference

[DynamicArrayInstance Class](#) [▶ 2160]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.13.2.1 DynamicArrayInstance.TryGetElement Method

Overload List

	Name	Description
	TryGetElement(IList..Int32.., ISymbol.) [▶ 2173]	Tries to get the array element with the specified indices (jagged array support).
	TryGetElement(.Int32.., ISymbol.) [▶ 2174]	Tries to get the array element

Reference

[DynamicArrayInstance Class](#) [▶ 2160]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.13.2.1.1 DynamicArrayInstance.TryGetElement Method (IList..Int32.., ISymbol.)

Tries to get the array element with the specified indices (jagged array support).

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetElement(
    IList<int[]> jaggedIndices,
    out ISymbol?? symbol
)
```

Parameters

jaggedIndices	Type: System.Collections.Generic.IList..Int32.. The jagged indices list.
symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691]. The symbol.

Return Value

Type: Boolean
true if found, false if the jagged indices specifiers is out-of-range.

Exceptions

Exception	Condition
ArgumentNullException	jaggedIndices
ArgumentOutOfRangeException	jaggedIndices

Reference

[DynamicArrayInstance Class](#) [▶ 2160]

[TryGetElement Overload](#) [▶ 2173]

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.13.2.1.2 DynamicArrayInstance.TryGetElement Method (.Int32., ISymbol.)

Tries to get the array element

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetElement(
    int[] indices,
    out ISymbol?? symbol
)
```

Parameters

indices	Type: <code>.System.Int32</code> . The indices.
symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] . The found Array element symbol (out-parameter).

Return Value

Type: Boolean

true if found, false if the indices specifiers is out-of-range.

Reference

[DynamicArrayInstance Class \[► 2160\]](#)

[TryGetElement Overload \[► 2173\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.13.2.2 DynamicArrayInstance.TryGetIndex Method

Provides the implementation for operations that get a value by index. Classes derived from the `DynamicObject` class can override this method to specify dynamic behavior for indexing operations.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TryGetIndex(
    GetIndexBinder binder,
    Object[] indexes,
    out Object?? result
)
```

Parameters

binder	Type: <code>System.Dynamic.GetIndexBinder</code> Provides information about the operation.
--------	---

indexes	Type: <code>.System.Object</code> . The indexes that are used in the operation. For example, for the <code>sampleObject[3]</code> operation in C# (<code>sampleObject(3)</code> in Visual Basic), where <code>sampleObject</code> is derived from the <code>DynamicObject</code> class, <code>indexes[0]</code> is equal to 3.
result	Type: <code>System.Object</code> . The result of the index operation.

Return Value

Type: Boolean

true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a run-time exception is thrown.)

Reference

[DynamicArrayInstance Class](#) |> 2160]

[TwinCAT.TypeSystem Namespace](#) |> 2083]

6.11.13.2.3 DynamicArrayInstance.TrySetIndex Method

Provides the implementation for operations that set a value by index. Classes derived from the `DynamicObject` class can override this method to specify dynamic behavior for operations that access objects by a specified index.

Namespace: [TwinCAT.TypeSystem](#) |> 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TrySetIndex(
    SetIndexBinder binder,
    Object[] indexes,
    Object? value
)
```

Parameters

binder	Type: <code>System.Dynamic.SetIndexBinder</code> Provides information about the operation.
indexes	Type: <code>.System.Object</code> . The indexes that are used in the operation. For example, for the <code>sampleObject[3] = 10</code> operation in C# (<code>sampleObject(3) = 10</code> in Visual Basic), where <code>sampleObject</code> is derived from the <code>DynamicObject</code> class, <code>indexes[0]</code> is equal to 3.
value	Type: <code>System.Object</code> The value to set to the object that has the specified index. For example, for the <code>sampleObject[3] = 10</code> operation in C# (<code>sampleObject(3) = 10</code> in Visual Basic), where <code>sampleObject</code> is derived from the <code>DynamicObject</code> class, <code>value</code> is equal to 10.

Return Value

Type: Boolean

true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a language-specific run-time exception is thrown.)

Reference



[DynamicArrayInstance Class](#) |> 2160]

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.13.3 DynamicArrayInstance Events

The [DynamicArrayInstance \[► 2160\]](#) type exposes the following members.

Events

	Name	Description
	RawValueChanged [► 2323]	Occurs when the RawValue of the IValueSymbol [► 2775] has changed. (Inherited from DynamicSymbol [► 2266] .)
	ValueChanged [► 2323]	Occurs when the (Primitive) value of the IValueSymbol [► 2775] has changed. (Inherited from DynamicSymbol [► 2266] .)

Reference


[DynamicArrayInstance Class \[► 2160\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.13.4 DynamicArrayInstance Fields

The [DynamicArrayInstance \[► 2160\]](#) type exposes the following members.

Fields

	Name	Description
	syncObject [► 2327]	Synchronization object (Inherited from DynamicSymbol [► 2266] .)

Reference

[DynamicArrayInstance Class \[► 2160\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.14 DynamicOversamplingArrayInstance Class

Dynamic Array Instance

Inheritance Hierarchy

System.Object
 System.Dynamic.DynamicObject
[TwinCAT.TypeSystem.DynamicSymbol \[► 2266\]](#)
[TwinCAT.TypeSystem.DynamicArrayInstance \[► 2160\]](#)
 TwinCAT.TypeSystem.DynamicOversamplingArrayInstance

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229
























Syntax












C#

```
public sealed class DynamicOversamplingArrayInstance : DynamicArrayInstance,
    IOversamplingArrayInstance, IArrayInstance, ISymbol, IAttributedInstance, IInstance,
    IBitSize
```









The [DynamicOversamplingArrayInstance](#) type exposes the following members.


















Properties















	Name	Description
	InnerSymbol [▶ 2274]	Gets the inner symbol of this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266] .)
	AccessRights [▶ 2274]	Gets the access rights. (Inherited from DynamicSymbol [▶ 2266] .)
	Attributes [▶ 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [▶ 2266] .)
	BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from DynamicSymbol [▶ 2266] .)
	ByteSize [▶ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [▶ 2266] .)
	Category [▶ 2277]	Gets the category. (Inherited from DynamicSymbol [▶ 2266] .)
	Comment [▶ 2277]	Gets the comment of the IInstance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266] .)
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266] .)
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549] . (Inherited from DynamicSymbol [▶ 2266] .)
	Dimensions [▶ 2167]	Gets the dimensions as read only collection. (Inherited from DynamicArrayInstance [▶ 2160] .)
	Elements [▶ 2168]	Gets the contained Array Elements as read only collection. (Inherited from DynamicArrayInstance [▶ 2160] .)
	ElementType [▶ 2168]	Gets the type of the contained elements. (Inherited from DynamicArrayInstance [▶ 2160] .)
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances, what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266] .)
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266] .)
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266] .)
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266] .)
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DynamicSymbol [▶ 2266] .)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from DynamicSymbol [▶ 2266] .)
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266] .)
	IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266] .)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266] .)
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266] .)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266] .)

	Name	Description
	IsReference [▶ 2284]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)
	Item [▶ 2169]	Gets the ISymbol [▶ 2691] with the specified indices. (Inherited from DynamicArrayInstance [▶ 2160].)
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
	OversamplingElement [▶ 2182]	Gets the oversampling element.
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
	Size [▶ 2286]	Gets the size of the IInstance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)



Methods

	Name	Description
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)
	GetDynamicMemberNames [▶ 2293]	Returns the enumeration of all dynamic member names. (Inherited from DynamicSymbol [▶ 2266].)
	GetHashCode [▶ 2293]	Gets the GetHashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	ReadRawValueAsyn c [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject .)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryGetElement(IList.. Int32.., ISymbol.) [▶ 2173]	Tries to get the array element with the specified indices (jagged array support). (Inherited from DynamicArrayInstance [▶ 2160].)
	TryGetElement(.Int3 2., ISymbol.) [▶ 2174]	Tries to get the array element (Inherited from DynamicArrayInstance [▶ 2160].)
	TryGetIndex [▶ 2174]	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicArrayInstance [▶ 2160].)
	TryGetMember [▶ 2308]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Inherited from DynamicSymbol [▶ 2266].)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject .)
	TryInvokeMember	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicObject .)
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	TrySetIndex [▶ 2175]	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicArrayInstance [▶ 2160].)
	TrySetMember	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicObject .)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject .)
 	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from DynamicSymbol [▶ 2266].)
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte, Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)

Events

	Name	Description
	RawValueChanged [▶ 2323]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)
	ValueChanged [▶ 2323]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference















[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.14.1 DynamicOversamplingArrayInstance Properties

The [DynamicOversamplingArrayInstance \[▶ 2176\]](#) type exposes the following members.

Properties

	Name	Description
	InnerSymbol [▶ 2274]	Gets the inner symbol of this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266] .)
	AccessRights [▶ 2274]	Gets the access rights. (Inherited from DynamicSymbol [▶ 2266] .)
	Attributes [▶ 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [▶ 2266] .)
	BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from DynamicSymbol [▶ 2266] .)
	ByteSize [▶ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [▶ 2266] .)
	Category [▶ 2277]	Gets the category. (Inherited from DynamicSymbol [▶ 2266] .)
	Comment [▶ 2277]	Gets the comment of the IInstance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266] .)
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266] .)
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549] . (Inherited from DynamicSymbol [▶ 2266] .)
	Dimensions [▶ 2167]	Gets the dimensions as read only collection. (Inherited from DynamicArrayInstance [▶ 2160] .)
	Elements [▶ 2168]	Gets the contained Array Elements as read only collection. (Inherited from DynamicArrayInstance [▶ 2160] .)
	ElementType [▶ 2168]	Gets the type of the contained elements. (Inherited from DynamicArrayInstance [▶ 2160] .)
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances, what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266] .)
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266] .)
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266] .)
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266] .)
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DynamicSymbol [▶ 2266] .)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from DynamicSymbol [▶ 2266] .)
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266] .)
	IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266] .)

	Name	Description
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266].)
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266].)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266].)
	IsReference [▶ 2284]	Indicates that the Instance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)
	Item [▶ 2169]	Gets the ISymbol [▶ 2691] with the specified indices. (Inherited from DynamicArrayInstance [▶ 2160].)
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
	OversamplingElement [▶ 2182]	Gets the oversampling element.
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
	Size [▶ 2286]	Gets the size of the Instance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)

Reference

[DynamicOversamplingArrayInstance Class](#) [[▶ 2176](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.14.1.1 DynamicOversamplingArrayInstance.OversamplingElement Property

Gets the oversampling element.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISymbol OversamplingElement { get; }
```


Property Value

Type: [ISymbol](#) [[▶ 2691](#)]
 The oversampling element.

Implements

[IOversamplingArrayInstance.OversamplingElement](#) [[▶ 2579](#)]

Exceptions

Exception	Condition
NotImplementedException	

Reference












[DynamicOversamplingArrayInstance Class](#) [[▶ 2176](#)]

















[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]













6.11.14.2 DynamicOversamplingArrayInstance Methods

The [DynamicOversamplingArrayInstance](#) [[▶ 2176](#)] type exposes the following members.

Methods

	Name	Description
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)
	GetDynamicMemberNames [▶ 2293]	Returns the enumeration of all dynamic member names. (Inherited from DynamicSymbol [▶ 2266].)
	GetHashCode [▶ 2293]	Gets the HashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue. [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValueAsyn c [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject .)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryGetElement(IList.. Int32.. ISymbol.) [▶ 2173]	Tries to get the array element with the specified indices (jagged array support). (Inherited from DynamicArrayInstance [▶ 2160].)
	TryGetElement(.Int32.. ISymbol.) [▶ 2174]	Tries to get the array element (Inherited from DynamicArrayInstance [▶ 2160].)
	TryGetIndex [▶ 2174]	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicArrayInstance [▶ 2160].)
	TryGetMember [▶ 2308]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Inherited from DynamicSymbol [▶ 2266].)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject .)
	TryInvokeMember	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicObject .)
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	TrySetIndex [▶ 2175]	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicArrayInstance [▶ 2160].)
	TrySetMember	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicObject .)

	Name	Description
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
 	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from DynamicSymbol [▶ 2266].)
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte, Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference



[DynamicOversamplingArrayInstance Class](#) [▶ 2176]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.14.3 DynamicOversamplingArrayInstance Events

The [DynamicOversamplingArrayInstance](#) [▶ 2176] type exposes the following members.

Events

	Name	Description
	RawValueChanged [▶ 2323]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)
	ValueChanged [▶ 2323]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)

Reference

[DynamicOversamplingArrayInstance Class](#) [► 2176]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.15 DynamicPointerInstance Class

Dynamic Pointer Instance

Inheritance Hierarchy

System.Object

System.Dynamic.DynamicObject

[TwinCAT.TypeSystem.DynamicSymbol](#) [► 2266]

[TwinCAT.TypeSystem.DynamicPointerInstance](#)

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229










Syntax

C#

```
public sealed class DynamicPointerInstance : DynamicSymbol,
    IPointerInstance, ISymbol, IAttributedInstance, IInstance, IBitSize
```




The DynamicPointerInstance type exposes the following members.


















Properties
















	Name	Description
	InnerSymbol [► 2274]	Gets the inner symbol of this DynamicSymbol [► 2266] (Inherited from DynamicSymbol [► 2266].)
	AccessRights [► 2274]	Gets the access rights. (Inherited from DynamicSymbol [► 2266].)
	Attributes [► 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [► 2266].)
	BitSize [► 2276]	Gets the size of the IDataType [► 2475] in bits. (Inherited from DynamicSymbol [► 2266].)
	ByteSize [► 2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [► 2266].)
	Category [► 2277]	Gets the category. (Inherited from DynamicSymbol [► 2266].)
	Comment [► 2277]	Gets the comment of the IInstance [► 2549] (Inherited from DynamicSymbol [► 2266].)
	Connection [► 2277]	Gets the connection bound to this DynamicSymbol [► 2266] (Inherited from DynamicSymbol [► 2266].)
	DataType [► 2278]	Gets the IDataType [► 2475] of the IInstance [► 2549]. (Inherited from DynamicSymbol [► 2266].)
	HasValue [► 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances, what means the Symbols with locations). (Inherited from DynamicSymbol [► 2266].)
	InstanceName [► 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [► 2266].)
	InstancePath [► 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [► 2266].)

	Name	Description
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266].)
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DynamicSymbol [▶ 2266].)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is a container type. (Inherited from DynamicSymbol [▶ 2266].)
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266].)
	IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266].)
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266].)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266].)
	IsReference [▶ 2284]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
	Reference [▶ 2191]	Gets the resolved reference of Pointer / Reference (or NULL if PVOID)
	Size [▶ 2286]	Gets the size of the IInstance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)


Methods


	Name	Description
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)
	GetDynamicMemberNames [▶ 2194]	Returns the enumeration of all dynamic member names. (Overrides DynamicSymbol.GetDynamicMemberNames . [▶ 2293].)
	GetHashCode [▶ 2293]	Gets the HashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue. [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValueAsyn c [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject.)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject.)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetIndex	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicObject.)
	TryGetMember [▶ 2195]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Overrides DynamicSymbol.TryGetMember(GetMemberBinder, Object.) [▶ 2308].)

	Name	Description
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject.)
	TryInvokeMember	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicObject.)
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266] .)
	TrySetIndex	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicObject.)
	TrySetMember	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicObject.)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
 	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266] . (Inherited from DynamicSymbol [▶ 2266] .)
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from DynamicSymbol [▶ 2266] .)
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266] .)
	WriteRawValue(Byte) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266] .)
	WriteRawValue(Byte, Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266] .)
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266] .)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266] . (Inherited from DynamicSymbol [▶ 2266] .)
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266] . (Inherited from DynamicSymbol [▶ 2266] .)
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266] .)

Events

	Name	Description
	RawValueChanged [▶ 2323]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266] .)

	Name	Description
	ValueChanged [▶ 2323]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)















Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.15.1 DynamicPointerInstance Properties

The [DynamicPointerInstance](#) [▶ 2186] type exposes the following members.

Properties

	Name	Description
	_InnerSymbol [▶ 2274]	Gets the inner symbol of this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	AccessRights [▶ 2274]	Gets the access rights. (Inherited from DynamicSymbol [▶ 2266].)
	Attributes [▶ 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [▶ 2266].)
	BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from DynamicSymbol [▶ 2266].)
	ByteSize [▶ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [▶ 2266].)
	Category [▶ 2277]	Gets the category. (Inherited from DynamicSymbol [▶ 2266].)
	Comment [▶ 2277]	Gets the comment of the IInstance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266].)
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances, what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266].)
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266].)
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is a container type. (Inherited from DynamicSymbol [▶ 2266].)
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266].)
	IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266].)
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266].)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266].)
	IsReference [▶ 2284]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
	Reference [▶ 2191]	Gets the resolved reference of Pointer / Reference (or NULL if PVOID)
	Size [▶ 2286]	Gets the size of the IInstance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)

Reference

[DynamicPointerInstance Class](#) [▶ 2186]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.15.1 DynamicPointerInstance.Reference Property

Gets the resolved reference of Pointer / Reference (or NULL if PVOID)

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISymbol? Reference { get; }
```

Property Value

Type: [ISymbol](#) [▶ 2691]

The reference symbol or NULL if PVOID Pointer.

Implements

[IPointerInstance.Reference](#) [▶ 2583]

Reference














[DynamicPointerInstance Class](#) [▶ 2186]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]








6.11.15.2 DynamicPointerInstance Methods

The [DynamicPointerInstance](#) [▶ 2186] type exposes the following members.

Methods

	Name	Description
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)
	GetDynamicMemberNames [▶ 2194]	Returns the enumeration of all dynamic member names. (Overrides DynamicSymbol.GetDynamicMemberNames . [▶ 2293].)
	GetHashCode [▶ 2293]	Gets the HashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue . [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValueAsync [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue . [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject.)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject.)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetIndex	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicObject.)
	TryGetMember [▶ 2195]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Overrides DynamicSymbol.TryGetMember(GetMemberBinder, Object.) [▶ 2308].)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject.)
	TryInvokeMember	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicObject.)
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	TrySetIndex	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicObject.)
	TrySetMember	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicObject.)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
 	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266] . (Inherited from DynamicSymbol [▶ 2266].)
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from DynamicSymbol [▶ 2266] .)
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266] .)

	Name	Description
	WriteRawValue(Byte) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266] .)
	WriteRawValue(Byte, Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266] .)
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266] .)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266] . (Inherited from DynamicSymbol [▶ 2266] .)
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266] . (Inherited from DynamicSymbol [▶ 2266] .)
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266] .)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021] .)

Reference

[DynamicPointerInstance Class \[▶ 2186\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.15.2.1 **DynamicPointerInstance.GetDynamicMemberNames Method**

Returns the enumeration of all dynamic member names.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override IEnumerable<string> GetDynamicMemberNames()
```

Return Value

Type: [IEnumerable.String](#).

A sequence that contains dynamic member names.

Reference

[DynamicPointerInstance Class \[▶ 2186\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.15.2 DynamicPointerInstance.TryGetMember Method

Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TryGetMember(
    GetMemberBinder binder,
    out Object?? result
)
```

Parameters

binder	Type: System.Dynamic.GetMemberBinder Provides information about the object that called the dynamic operation. The binder.Name property provides the name of the member on which the dynamic operation is performed. For example, for the Console.WriteLine(sampleObject.SampleProperty) statement, where sampleObject is an instance of the class derived from the DynamicObject class, binder.Name returns "SampleProperty". The binder.IgnoreCase property specifies whether the member name is case-sensitive.
result	Type: System.Object. The result of the get operation. For example, if the method is called for a property, you can assign the property value to result.

Return Value

Type: Boolean

true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a run-time exception is thrown.)

Reference



[DynamicPointerInstance Class](#) [[▶ 2186](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.15.3 DynamicPointerInstance Events

The [DynamicPointerInstance](#) [[▶ 2186](#)] type exposes the following members.

Events

	Name	Description
	RawValueChanged [▶ 2323]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)
	ValueChanged [▶ 2323]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)

Reference

[DynamicPointerInstance Class](#) [[▶ 2186](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.16 DynamicPointerValue Class

Class DynamicPointerValue.

Inheritance Hierarchy

System.Object
 System.Dynamic.DynamicObject
 TwinCAT.TypeSystem.DynamicValue [▶ 2347]
 TwinCAT.TypeSystem.DynamicPointerValue

Namespace: TwinCAT.TypeSystem [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229











Syntax

C#





```
public class DynamicPointerValue : DynamicValue
```











The DynamicPointerValue type exposes the following members.


Properties

	Name	Description
	Age [▶ 2351]	Gets the age of the value (last successful read of the value) (Inherited from DynamicValue [▶ 2347].)
	CachedRaw [▶ 2351]	Gets the cached Raw internal Data. (Inherited from DynamicValue [▶ 2347].)
	DataType [▶ 2352]	Gets the data type bound to this IValue [▶ 2745] (Inherited from DynamicValue [▶ 2347].)
	IsPrimitive [▶ 2353]	Gets a value indicating whether this IValue [▶ 2745] is a primitive value. (Inherited from DynamicValue [▶ 2347].)
	ParentValue [▶ 2356]	Gets the parent value. (Inherited from DynamicValue [▶ 2347].)
	ResolvedType [▶ 2353]	Gets the resolved type. (Inherited from DynamicValue [▶ 2347].)
	Symbol [▶ 2353]	Gets the symbol that is bound to this value. (Inherited from DynamicValue [▶ 2347].)
	TimeStamp [▶ 2354]	Gets the Time stamp of the last successful read of the Value. (Inherited from DynamicValue [▶ 2347].)
	UpdateMode [▶ 2354]	Gets / Sets the update mode (Inherited from DynamicValue [▶ 2347].)
	ValueFactory [▶ 2355]	The value factory (Inherited from DynamicValue [▶ 2347].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetDynamicMemberNames [▶ 2201]	Returns the enumeration of all dynamic member names. (Overrides DynamicValue.GetDynamicMemberNames. [▶ 2358].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)

	Name	Description
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Read [▶ 2359]	Reads the value (via ADS) (Inherited from DynamicValue [▶ 2347].)
	ReadAsync [▶ 2359]	read as an asynchronous operation. (Inherited from DynamicValue [▶ 2347].)
	ResolveValue [▶ 2360]	Resolves the Value object to its primitive value. (Inherited from DynamicValue [▶ 2347].)
	setMemberValue [▶ 2374]	Writes the specified member element. (Inherited from DynamicValue [▶ 2347].)
	ToString [▶ 2361]	Returns a String that represents this instance. (Inherited from DynamicValue [▶ 2347].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject.)
	TryConvert [▶ 2361]	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicValue [▶ 2347].)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetArrayElementValues [▶ 2362]	Returns Array Element values. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndex [▶ 2363]	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndexValue (.Int32., Object.) [▶ 2364]	Reads the specified array element. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndexValue (.Object., Object.) [▶ 2364]	Tries the get index value. (Inherited from DynamicValue [▶ 2347].)
	TryGetMember [▶ 2365]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Inherited from DynamicValue [▶ 2347].)




	Name	Description
	TryGetMemberValue [▶ 2202]	Tries the get member value. (Overrides DynamicValue.TryGetMemberValue(String, Object.) [▶ 2366].)
	TryInvoke [▶ 2366]	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicValue [▶ 2347].)
	TryInvokeMember [▶ 2367]	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicValue [▶ 2347].)
	TryResolveValue [▶ 2368]	Tries to resolves the Value object to its primitive value. (Inherited from DynamicValue [▶ 2347].)
	TrySetIndex [▶ 2369]	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicValue [▶ 2347].)
	TrySetIndexValue(Int32, Object) [▶ 2370]	Tries to set the indexed value on Arrays (Inherited from DynamicValue [▶ 2347].)
	TrySetIndexValue(Object, Object) [▶ 2370]	Tries to set the indexed value on Arrays (Inherited from DynamicValue [▶ 2347].)
	TrySetMember [▶ 2371]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicValue [▶ 2347].)
	TrySetMemberValue [▶ 2372]	Tries to Set a Member/Property Value (Inherited from DynamicValue [▶ 2347].)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject .)
	Write [▶ 2372]	Writes the value (via ADS) (Inherited from DynamicValue [▶ 2347].)
	WriteAsync [▶ 2373]	write as an asynchronous operation. (Inherited from DynamicValue [▶ 2347].)

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

[TwinCAT.TypeSystem.DynamicValue](#) [[▶ 2347](#)]











Also see about this

-  [DynamicValue.ReadMember Method](#) [[▶ 2360](#)]
-  [DynamicValue.TrySetIndexValue Method](#) [[▶ 2369](#)]
-  [DynamicValue.WriteMember Method](#) [[▶ 2373](#)]

6.11.16.1 DynamicPointerValue Properties

The [DynamicPointerValue](#) [[▶ 2196](#)] type exposes the following members.

Properties

	Name	Description
	Age [▶ 2351]	Gets the age of the value (last successful read of the value) (Inherited from DynamicValue [▶ 2347].)
	CachedRaw [▶ 2351]	Gets the cached Raw internal Data. (Inherited from DynamicValue [▶ 2347].)
	DataType [▶ 2352]	Gets the data type bound to this IValue [▶ 2745] (Inherited from DynamicValue [▶ 2347].)
	IsPrimitive [▶ 2353]	Gets a value indicating whether this IValue [▶ 2745] is a primitive value. (Inherited from DynamicValue [▶ 2347].)
	ParentValue [▶ 2356]	Gets the parent value. (Inherited from DynamicValue [▶ 2347].)
	ResolvedType [▶ 2353]	Gets the resolved type. (Inherited from DynamicValue [▶ 2347].)
	Symbol [▶ 2353]	Gets the symbol that is bound to this value. (Inherited from DynamicValue [▶ 2347].)
	TimeStamp [▶ 2354]	Gets the Time stamp of the last successful read of the Value. (Inherited from DynamicValue [▶ 2347].)
	UpdateMode [▶ 2354]	Gets / Sets the update mode (Inherited from DynamicValue [▶ 2347].)
	ValueFactory [▶ 2355]	The value factory (Inherited from DynamicValue [▶ 2347].)

Reference








[DynamicPointerValue Class \[▶ 2196\]](#)


















[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)











6.11.16.2 DynamicPointerValue Methods

The [DynamicPointerValue \[▶ 2196\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetDynamicMemberNames [▶ 2201]	Returns the enumeration of all dynamic member names. (Overrides DynamicValue.GetDynamicMemberNames. [▶ 2358].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)

	Name	Description
	Read [▶ 2359]	Reads the value (via ADS) (Inherited from DynamicValue [▶ 2347].)
	ReadAsync [▶ 2359]	read as an asynchronous operation. (Inherited from DynamicValue [▶ 2347].)
	ResolveValue [▶ 2360]	Resolves the Value object to its primitive value. (Inherited from DynamicValue [▶ 2347].)
	setMemberValue [▶ 2374]	Writes the specified member element. (Inherited from DynamicValue [▶ 2347].)
	ToString [▶ 2361]	Returns a String that represents this instance. (Inherited from DynamicValue [▶ 2347].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject.)
	TryConvert [▶ 2361]	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicValue [▶ 2347].)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetArrayElement Values [▶ 2362]	Returns Array Element values. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndex [▶ 2363]	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndexValue(.I nt32., Object.) [▶ 2364]	Reads the specified array element. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndexValue(. Object., Object.) [▶ 2364]	Tries the get index value. (Inherited from DynamicValue [▶ 2347].)
	TryGetMember [▶ 2365]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Inherited from DynamicValue [▶ 2347].)
	TryGetMemberValu e [▶ 2202]	Tries the get member value. (Overrides DynamicValue.TryGetMemberValue(String, Object.) [▶ 2366].)
	TryInvoke [▶ 2366]	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicValue [▶ 2347].)




	Name	Description
	TryInvokeMember [▶ 2367]	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicValue [▶ 2347].)
	TryResolveValue [▶ 2368]	Tries to resolves the Value object to its primitive value. (Inherited from DynamicValue [▶ 2347].)
	TrySetIndex [▶ 2369]	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicValue [▶ 2347].)
	TrySetIndexValue(Int32, Object) [▶ 2370]	Tries to set the indexed value on Arrays (Inherited from DynamicValue [▶ 2347].)
	TrySetIndexValue(Object, Object) [▶ 2370]	Tries to set the indexed value on Arrays (Inherited from DynamicValue [▶ 2347].)
	TrySetMember [▶ 2371]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicValue [▶ 2347].)
	TrySetMemberValue [▶ 2372]	Tries to Set a Member/Property Value (Inherited from DynamicValue [▶ 2347].)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject .)
	Write [▶ 2372]	Writes the value (via ADS) (Inherited from DynamicValue [▶ 2347].)
	WriteAsync [▶ 2373]	write as an asynchronous operation. (Inherited from DynamicValue [▶ 2347].)

Reference

[DynamicPointerValue Class](#) [▶ 2196]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

Also see about this

-  [DynamicValue.ReadMember Method](#) [▶ 2360]
-  [DynamicValue.TrySetIndexValue Method](#) [▶ 2369]
-  [DynamicValue.WriteMember Method](#) [▶ 2373]

6.11.16.2.1 DynamicPointerValue.GetDynamicMemberNames Method

Returns the enumeration of all dynamic member names.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override IEnumerable<string> GetDynamicMemberNames ()
```

Return Value

Type: IEnumerable.String.
A sequence that contains dynamic member names.

Reference

[DynamicPointerValue Class \[▸ 2196\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.16.2.2 DynamicPointerValue.TryGetMemberValue Method

Tries the get member value.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public override bool TryGetMemberValue(
    string name,
    out Object?? result
)
```

Parameters

name	Type: System.String The name.
result	Type: System.Object. The result.

Return Value

Type: Boolean
true if the member value is returned, false otherwise.

Implements

[IStructValue.TryGetMemberValue\(String, Object.\) \[▸ 2680\]](#)

Exceptions

Exception	Condition
SymbolException [▸ 2933]	

Reference

[DynamicPointerValue Class \[▸ 2196\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.17 DynamicReferenceInstance Class

Dynamic Reference Instance

Inheritance Hierarchy

System.Object
 System.Dynamic.DynamicObject
 TwinCAT.TypeSystem.DynamicSymbol [▶ 2266]
 TwinCAT.TypeSystem.DynamicReferenceInstance

Namespace: TwinCAT.TypeSystem [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

















Syntax

C#

```
public class DynamicReferenceInstance : DynamicSymbol,
    IReferenceInstance, ISymbol, IAttributedInstance, IInstance, IBitSize
```





The DynamicReferenceInstance type exposes the following members.






















Properties

	Name	Description
	InnerSymbol [▶ 2274]	Gets the inner symbol of this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	AccessRights [▶ 2274]	Gets the access rights. (Inherited from DynamicSymbol [▶ 2266].)
	AllowGIOAccess [▶ 2275]	Indicates, that the aggregates symbols is an IProcessImageAddress (and most probably IAdsSymbol) (Inherited from DynamicSymbol [▶ 2266].)
	Attributes [▶ 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [▶ 2266].)
	BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from DynamicSymbol [▶ 2266].)
	ByteSize [▶ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [▶ 2266].)
	Category [▶ 2277]	Gets the category. (Inherited from DynamicSymbol [▶ 2266].)
	Comment [▶ 2277]	Gets the comment of the IInstance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266].)
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances, what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266].)
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266].)
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DynamicSymbol [▶ 2266].)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from DynamicSymbol [▶ 2266].)










	Name	Description
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266].)
	IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266].)
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266].)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266].)
	IsReference [▶ 2284]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
	ReferencedType [▶ 2209]	Gets the referenced type of the IReferenceInstance [▶ 2595]
	ResolvedByteSize [▶ 2210]	Gets the resolved byte size of the IReferenceInstance [▶ 2595].
	ResolvedCategory [▶ 2210]	Gets the Category of the Referenced Symbol.
	ResolvedType [▶ 2211]	Gets the resolved type of the IReferenceInstance [▶ 2595].
	Size [▶ 2286]	Gets the size of the IInstance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)

Methods



	Name	Description
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetDynamicMemberNames [▶ 2293]	Returns the enumeration of all dynamic member names. (Inherited from DynamicSymbol [▶ 2266].)
	GetHashCode [▶ 2293]	Gets the GetHashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	OnReadAnyValue [▶ 2293]	Handler function for reading ADS 'Any' Values. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadRawValue [▶ 2294]	Handler function for reading Raw symbol value. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadRawValueAsync [▶ 2294]	Handler function reading the raw value of the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadValue [▶ 2295]	Handler function for the (Inherited from DynamicSymbol [▶ 2266].)
	OnReadValueAsync [▶ 2296]	Handler function reading the DynamicSymbols [▶ 2266] value asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	OnSetInstanceName [▶ 2296]	Sets a new InstanceName InstancePath (Inherited from DynamicSymbol [▶ 2266].)
	OnTryReadValue [▶ 2297]	Handler function for the (Inherited from DynamicSymbol [▶ 2266].)
	OnTryWriteValue [▶ 2297]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteRawValue [▶ 2298]	Handler function for reading symbols raw value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteRawValueAsync [▶ 2298]	Handler function for writing the raw DynamicSymbol [▶ 2266] value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteValue [▶ 2299]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteValueAsync [▶ 2300]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue. [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValueAsync [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)



	Name	Description
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject .)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryGetIndex [▶ 2214]	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Overrides DynamicObject.TryGetIndex(GetIndexBinder, .Object., Object.) .)
	TryGetMember [▶ 2308]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Inherited from DynamicSymbol [▶ 2266].)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject .)
	TryInvokeMember	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicObject .)
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	TrySetIndex [▶ 2215]	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Overrides DynamicObject.TrySetIndex(SetIndexBinder, .Object., Object.) .)
	TrySetMember	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicObject .)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject .)
 	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from DynamicSymbol [▶ 2266].)
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte, Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)

Events

	Name	Description
	RawValueChanged [▶ 2323]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)
	ValueChanged [▶ 2323]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)

Fields

	Name	Description
	resolvedReferenceType [▶ 2217]	The resolved alias type
	syncObject [▶ 2327]	Synchronization object (Inherited from DynamicSymbol [▶ 2266].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference












[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.17.1 DynamicReferenceInstance Properties

The [DynamicReferenceInstance](#) [▶ 2202] type exposes the following members.

Properties

	Name	Description
	InnerSymbol [▶ 2274]	Gets the inner symbol of this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	AccessRights [▶ 2274]	Gets the access rights. (Inherited from DynamicSymbol [▶ 2266].)
	AllowIIOAccess [▶ 2275]	Indicates, that the aggregates symbols is an IProcessImageAddress (and most probably IAdsSymbol) (Inherited from DynamicSymbol [▶ 2266].)
	Attributes [▶ 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [▶ 2266].)
	BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from DynamicSymbol [▶ 2266].)
	ByteSize [▶ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [▶ 2266].)
	Category [▶ 2277]	Gets the category. (Inherited from DynamicSymbol [▶ 2266].)
	Comment [▶ 2277]	Gets the comment of the IInstance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266].)
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances , what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266].)
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266].)
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DynamicSymbol [▶ 2266].)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from DynamicSymbol [▶ 2266].)
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266].)
	IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266].)
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266].)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266].)
	IsReference [▶ 2284]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
	ReferencedType [▶ 2209]	Gets the referenced type of the IReferenceInstance [▶ 2595]
	ResolvedByteSize [▶ 2210]	Gets the resolved byte size of the IReferenceInstance [▶ 2595].
	ResolvedCategory [▶ 2210]	Gets the Category of the Referenced Symbol.
	ResolvedType [▶ 2211]	Gets the resolved type of the IReferenceInstance [▶ 2595].
	Size [▶ 2286]	Gets the size of the IInstance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)

Reference

[DynamicReferenceInstance Class](#) [▶ 2202]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.17.1.1 DynamicReferenceInstance.ReferencedType Property

Gets the referenced type of the [IReferenceInstance](#) [▶ 2595]

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataType? ReferencedType { get; }
```

Property Value

Type: [IDataType](#) [▶ 2475]

The referenced type

Implements

[IReferenceInstance.ReferencedType](#) [▶ 2598]

Reference[DynamicReferenceInstance Class \[► 2202\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.17.1.2 DynamicReferenceInstance.ResolvedByteSize Property**Gets the resolved byte size of the [IReferenceInstance \[► 2595\]](#).**Namespace:** [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public int ResolvedByteSize { get; }
```

Property Value

Type: Int32

The resolved byte size.

Implements[IReferenceInstance.ResolvedByteSize \[► 2599\]](#)**Reference**[DynamicReferenceInstance Class \[► 2202\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.17.1.3 DynamicReferenceInstance.ResolvedCategory Property**

Gets the Category of the Referenced Symbol.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public DataTypeCategory ResolvedCategory { get; }
```

Property ValueType: [DataTypeCategory \[► 2111\]](#)

The resolved category.

Implements[IReferenceInstance.ResolvedCategory \[► 2599\]](#)**Exceptions**

Exception	Condition
NotImplementedException	

Reference

[DynamicReferenceInstance Class \[▶ 2202\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.17.1.4 DynamicReferenceInstance.ResolvedType Property

Gets the resolved type of the [IReferenceInstance \[▶ 2595\]](#).

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataType? ResolvedType { get; }
```

Property Value

Type: [IDataType \[▶ 2475\]](#)
The resolved type.

Implements

[IReferenceInstance.ResolvedType \[▶ 2599\]](#)

Reference







[DynamicReferenceInstance Class \[▶ 2202\]](#)























[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.17.2 DynamicReferenceInstance Methods







The [DynamicReferenceInstance \[▶ 2202\]](#) type exposes the following members.

Methods

	Name	Description
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266] .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetDynamicMemberNames [▶ 2293]	Returns the enumeration of all dynamic member names. (Inherited from DynamicSymbol [▶ 2266] .)
	GetHashCode [▶ 2293]	Gets the HashCode of the Address (Inherited from DynamicSymbol [▶ 2266] .)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)

	Name	Description
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	OnReadAnyValue [▶ 2293]	Handler function for reading ADS 'Any' Values. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadRawValue [▶ 2294]	Handler function for reading Raw symbol value. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadRawValueAsync [▶ 2294]	Handler function reading the raw value of the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadValue [▶ 2295]	Handler function for the (Inherited from DynamicSymbol [▶ 2266].)
	OnReadValueAsync [▶ 2296]	Handler function reading the DynamicSymbols [▶ 2266] value asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	OnSetInstanceName [▶ 2296]	Sets a new InstanceName InstancePath (Inherited from DynamicSymbol [▶ 2266].)
	OnTryReadValue [▶ 2297]	Handler function for the (Inherited from DynamicSymbol [▶ 2266].)
	OnTryWriteValue [▶ 2297]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteRawValue [▶ 2298]	Handler function for reading symbols raw value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteRawValueAsync [▶ 2298]	Handler function for writing the raw DynamicSymbol [▶ 2266] value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteValue [▶ 2299]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteValueAsync [▶ 2300]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue. [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValueAsync [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)

	Name	Description
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject.)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetIndex [▶ 2214]	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Overrides DynamicObject.TryGetIndex(GetIndexBinder, .Object., Object.).)
	TryGetMember [▶ 2308]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Inherited from DynamicSymbol [▶ 2266].)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject.)
	TryInvokeMember	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicObject.)
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	TrySetIndex [▶ 2215]	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Overrides DynamicObject.TrySetIndex(SetIndexBinder, .Object., Object.).)
	TrySetMember	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicObject.)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
 	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from DynamicSymbol [▶ 2266].)
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(.Byte) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	WriteRawValue(Byte, Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266] .)
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266] .)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266] . (Inherited from DynamicSymbol [▶ 2266] .)
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266] . (Inherited from DynamicSymbol [▶ 2266] .)
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266] .)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021] .)

Reference

[DynamicReferenceInstance Class \[▶ 2202\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.17.2.1 DynamicReferenceInstance.TryGetIndex Method

Provides the implementation for operations that get a value by index. Classes derived from the [DynamicObject](#) class can override this method to specify dynamic behavior for indexing operations.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TryGetIndex(
    GetIndexBinder binder,
    Object[] indexes,
    out Object?? result
)
```

Parameters

binder	Type: System.Dynamic.GetIndexBinder Provides information about the operation.
indexes	Type: System.Object. The indexes that are used in the operation. For example, for the <code>sampleObject[3]</code> operation in C# (<code>sampleObject(3)</code> in Visual Basic), where <code>sampleObject</code> is derived from the <code>DynamicObject</code> class, <code>indexes[0]</code> is equal to 3.
result	Type: System.Object. The result of the index operation.

Return Value

Type: Boolean

true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a run-time exception is thrown.)

Reference

[DynamicReferenceInstance Class \[► 2202\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.17.2 DynamicReferenceInstance.TrySetIndex Method

Provides the implementation for operations that set a value by index. Classes derived from the `DynamicObject` class can override this method to specify dynamic behavior for operations that access objects by a specified index.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public override bool TrySetIndex(
    SetIndexBinder binder,
    Object[] indexes,
    Object? value
)
```

Parameters

binder	Type: System.Dynamic.SetIndexBinder Provides information about the operation.
indexes	Type: System.Object. The indexes that are used in the operation. For example, for the <code>sampleObject[3] = 10</code> operation in C# (<code>sampleObject(3) = 10</code> in Visual Basic), where <code>sampleObject</code> is derived from the <code>DynamicObject</code> class, <code>indexes[0]</code> is equal to 3.
value	Type: System.Object The value to set to the object that has the specified index. For example, for the <code>sampleObject[3] = 10</code> operation in C# (<code>sampleObject(3) = 10</code> in Visual Basic), where <code>sampleObject</code> is derived from the <code>DynamicObject</code> class, <code>value</code> is equal to 10.

Return Value

Type: Boolean

true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a language-specific run-time exception is thrown.)

Reference



[DynamicReferenceInstance Class \[► 2202\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.17.3 DynamicReferenceInstance Events

The [DynamicReferenceInstance \[► 2202\]](#) type exposes the following members.

Events

	Name	Description
	RawValueChanged [▶ 2323]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)
	ValueChanged [▶ 2323]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)

Reference



[DynamicReferenceInstance Class](#) [▶ 2202]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.17.4 DynamicReferenceInstance Fields

The [DynamicReferenceInstance](#) [▶ 2202] type exposes the following members.

Fields

	Name	Description
	resolvedReferenceType [▶ 2217]	The resolved alias type
	syncObject [▶ 2327]	Synchronization object (Inherited from DynamicSymbol [▶ 2266].)

Reference

[DynamicReferenceInstance Class](#) [▶ 2202]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.17.4.1 DynamicReferenceInstance.normalizedDict Field

Dictionary of normalized Instance Names

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
protected Dictionary<string, ISymbol> normalizedDict
```

Field Value

Type: [Dictionary.String, ISymbol](#) [▶ 2691].

Reference

[DynamicReferenceInstance Class](#) [▶ 2202]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.17.4.2 DynamicReferenceInstance.resolvedReferenceType Field

The resolved alias type

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected IDataTypeInfo resolvedReferenceType
```

Field Value

Type: [IDataTypeInfo \[▸ 2475\]](#)

Reference

[DynamicReferenceInstance Class \[▸ 2202\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.18 DynamicReferenceValue Class

Class DynamicReferenceValue.

Inheritance Hierarchy

System.Object

System.Dynamic.DynamicObject

[TwinCAT.TypeSystem.DynamicValue \[▸ 2347\]](#)

TwinCAT.TypeSystem.DynamicReferenceValue

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229







Syntax





C#

```
public class DynamicReferenceValue : DynamicValue
```

















The DynamicReferenceValue type exposes the following members.
















Properties




	Name	Description
	Age [▸ 2351]	Gets the age of the value (last successful read of the value) (Inherited from DynamicValue [▸ 2347] .)
	CachedRaw [▸ 2351]	Gets the cached Raw internal Data. (Inherited from DynamicValue [▸ 2347] .)
	DataType [▸ 2352]	Gets the data type bound to this IValue [▸ 2745] (Inherited from DynamicValue [▸ 2347] .)
	IsPrimitive [▸ 2353]	Gets a value indicating whether this IValue [▸ 2745] is a primitive value. (Inherited from DynamicValue [▸ 2347] .)
	ParentValue [▸ 2356]	Gets the parent value. (Inherited from DynamicValue [▸ 2347] .)
	ResolvedType [▸ 2353]	Gets the resolved type. (Inherited from DynamicValue [▸ 2347] .)

	Name	Description
	Symbol [▶ 2353]	Gets the symbol that is bound to this value. (Inherited from DynamicValue [▶ 2347].)
	TimeStamp [▶ 2354]	Gets the Time stamp of the last successful read of the Value. (Inherited from DynamicValue [▶ 2347].)
	UpdateMode [▶ 2354]	Gets / Sets the update mode (Inherited from DynamicValue [▶ 2347].)
	ValueFactory [▶ 2355]	The value factory (Inherited from DynamicValue [▶ 2347].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetDynamicMemberNames [▶ 2358]	Returns the enumeration of all dynamic member names. (Inherited from DynamicValue [▶ 2347].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Read [▶ 2359]	Reads the value (via ADS) (Inherited from DynamicValue [▶ 2347].)
	ReadAsync [▶ 2359]	read as an asynchronous operation. (Inherited from DynamicValue [▶ 2347].)
	ResolveValue [▶ 2360]	Resolves the Value object to its primitive value. (Inherited from DynamicValue [▶ 2347].)
	setMemberValue [▶ 2374]	Writes the specified member element. (Inherited from DynamicValue [▶ 2347].)
	ToString [▶ 2361]	Returns a String that represents this instance. (Inherited from DynamicValue [▶ 2347].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert [▶ 2361]	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicValue [▶ 2347].)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)

	Name	Description
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetArrayElement Values [▶ 2362]	Returns Array Element values. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndex [▶ 2363]	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndexValue(.Int32., Object.) [▶ 2364]	Reads the specified array element. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndexValue(Object., Object.) [▶ 2364]	Tries the get index value. (Inherited from DynamicValue [▶ 2347].)
	TryGetMember [▶ 2365]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Inherited from DynamicValue [▶ 2347].)
	TryGetMemberValue [▶ 2366]	Tries the get member value. (Inherited from DynamicValue [▶ 2347].)
	TryInvoke [▶ 2366]	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicValue [▶ 2347].)
	TryInvokeMember [▶ 2367]	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicValue [▶ 2347].)
	TryResolveValue [▶ 2368]	Tries to resolves the Value object to its primitive value. (Inherited from DynamicValue [▶ 2347].)
	TrySetIndex [▶ 2369]	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicValue [▶ 2347].)
	TrySetIndexValue(Int32., Object.) [▶ 2370]	Tries to set the indexed value on Arrays (Inherited from DynamicValue [▶ 2347].)
	TrySetIndexValue(Object., Object.) [▶ 2370]	Tries to set the indexed value on Arrays (Inherited from DynamicValue [▶ 2347].)
	TrySetMember [▶ 2371]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicValue [▶ 2347].)
	TrySetMemberValue [▶ 2372]	Tries to Set a Member/Property Value (Inherited from DynamicValue [▶ 2347].)





	Name	Description
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
	Write [▶ 2372]	Writes the value (via ADS) (Inherited from DynamicValue [▶ 2347].)
	WriteAsync [▶ 2373]	write as an asynchronous operation. (Inherited from DynamicValue [▶ 2347].)

Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

[TwinCAT.TypeSystem.DynamicValue \[▶ 2347\]](#)











Also see about this

-  [DynamicReferenceValue.ReadMember Method \[▶ 2223\]](#)
-  [DynamicValue.ReadMember Method \[▶ 2360\]](#)
-  [DynamicValue.TrySetIndexValue Method \[▶ 2369\]](#)
-  [DynamicValue.WriteMember Method \[▶ 2373\]](#)

6.11.18.1 DynamicReferenceValue Properties

The [DynamicReferenceValue \[▶ 2217\]](#) type exposes the following members.

Properties

	Name	Description
	Age [▶ 2351]	Gets the age of the value (last successful read of the value) (Inherited from DynamicValue [▶ 2347].)
	CachedRaw [▶ 2351]	Gets the cached Raw internal Data. (Inherited from DynamicValue [▶ 2347].)
	DataType [▶ 2352]	Gets the data type bound to this IValue [▶ 2745] (Inherited from DynamicValue [▶ 2347].)
	IsPrimitive [▶ 2353]	Gets a value indicating whether this IValue [▶ 2745] is a primitive value. (Inherited from DynamicValue [▶ 2347].)
	ParentValue [▶ 2356]	Gets the parent value. (Inherited from DynamicValue [▶ 2347].)
	ResolvedType [▶ 2353]	Gets the resolved type. (Inherited from DynamicValue [▶ 2347].)
	Symbol [▶ 2353]	Gets the symbol that is bound to this value. (Inherited from DynamicValue [▶ 2347].)
	TimeStamp [▶ 2354]	Gets the Time stamp of the last successful read of the Value. (Inherited from DynamicValue [▶ 2347].)
	UpdateMode [▶ 2354]	Gets / Sets the update mode (Inherited from DynamicValue [▶ 2347].)
	ValueFactory [▶ 2355]	The value factory (Inherited from DynamicValue [▶ 2347].)

Reference



















[DynamicReferenceValue Class \[▶ 2217\]](#)

















[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.18.2 DynamicReferenceValue Methods

The [DynamicReferenceValue](#) [[▶ 2217](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetDynamicMemberNames [▶ 2358]	Returns the enumeration of all dynamic member names. (Inherited from DynamicValue [▶ 2347].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Read [▶ 2359]	Reads the value (via ADS) (Inherited from DynamicValue [▶ 2347].)
	ReadAsync [▶ 2359]	read as an asynchronous operation. (Inherited from DynamicValue [▶ 2347].)
	ResolveValue [▶ 2360]	Resolves the Value object to its primitive value. (Inherited from DynamicValue [▶ 2347].)
	setMemberValue [▶ 2374]	Writes the specified member element. (Inherited from DynamicValue [▶ 2347].)
	ToString [▶ 2361]	Returns a String that represents this instance. (Inherited from DynamicValue [▶ 2347].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert [▶ 2361]	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicValue [▶ 2347].)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryGetArrayElementValues [▶ 2362]	Returns Array Element values. (Inherited from DynamicValue [▶ 2347].)

	Name	Description
	TryGetIndex [▶ 2363]	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndexValue(Int32, Object) [▶ 2364]	Reads the specified array element. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndexValue(Object, Object) [▶ 2364]	Tries the get index value. (Inherited from DynamicValue [▶ 2347].)
	TryGetMember [▶ 2365]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Inherited from DynamicValue [▶ 2347].)
	TryGetMemberValue [▶ 2366]	Tries the get member value. (Inherited from DynamicValue [▶ 2347].)
	TryInvoke [▶ 2366]	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicValue [▶ 2347].)
	TryInvokeMember [▶ 2367]	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicValue [▶ 2347].)
	TryResolveValue [▶ 2368]	Tries to resolve the Value object to its primitive value. (Inherited from DynamicValue [▶ 2347].)
	TrySetIndex [▶ 2369]	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicValue [▶ 2347].)
	TrySetIndexValue(Int32, Object) [▶ 2370]	Tries to set the indexed value on Arrays (Inherited from DynamicValue [▶ 2347].)
	TrySetIndexValue(Object, Object) [▶ 2370]	Tries to set the indexed value on Arrays (Inherited from DynamicValue [▶ 2347].)
	TrySetMember [▶ 2371]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicValue [▶ 2347].)
	TrySetMemberValue [▶ 2372]	Tries to Set a Member/Property Value (Inherited from DynamicValue [▶ 2347].)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject .)
	Write [▶ 2372]	Writes the value (via ADS) (Inherited from DynamicValue [▶ 2347].)
	WriteAsync [▶ 2373]	write as an asynchronous operation. (Inherited from DynamicValue [▶ 2347].)

Reference

[DynamicReferenceValue Class](#) [[▶ 2217](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

Also see about this

- [DynamicReferenceValue.ReadMember Method](#) [[▶ 2223](#)]
- [DynamicValue.ReadMember Method](#) [[▶ 2360](#)]
- [DynamicValue.TrySetIndexValue Method](#) [[▶ 2369](#)]
- [DynamicValue.WriteMember Method](#) [[▶ 2373](#)]

6.11.18.2.1 DynamicReferenceValue.ReadMember Method

Reads the specified member element.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
protected override Object ReadMember(  
    ISymbol memberInstance  
)
```

Parameters

memberInstance Type: [TwinCAT.TypeSystem.ISymbol](#) [[▶ 2691](#)]
The member instance.

Return Value

Type: [Object](#)

Reference

[DynamicReferenceValue Class](#) [[▶ 2217](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.19 DynamicRpcStructInstance Class

Dynamic struct instance with RPC Methods.

Inheritance Hierarchy

[System.Object](#)

[System.Dynamic.DynamicObject](#)

[TwinCAT.TypeSystem.DynamicSymbol](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem.DynamicStructInstance](#) [[▶ 2251](#)]

 TwinCAT.TypeSystem.DynamicRpcStructInstance

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5






Syntax














C#

```
public sealed class DynamicRpcStructInstance : DynamicStructInstance,
    IRpcStructInstance, IStructInstance, ISymbol, IAttributedInstance, IInstance,
    IBitSize, IRpcCallableInstance
```









The DynamicRpcStructInstance type exposes the following members.

















Properties




















	Name	Description
	InnerSymbol [▶ 2274]	Gets the inner symbol of this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	AccessRights [▶ 2274]	Gets the access rights. (Inherited from DynamicSymbol [▶ 2266].)
	Attributes [▶ 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [▶ 2266].)
	BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from DynamicSymbol [▶ 2266].)
	ByteSize [▶ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [▶ 2266].)
	Category [▶ 2277]	Gets the category. (Inherited from DynamicSymbol [▶ 2266].)
	Comment [▶ 2277]	Gets the comment of the IInstance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266].)
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	HasRpcMethods [▶ 2258]	Gets a value indicating whether this instance has RPC methods (Inherited from DynamicStructInstance [▶ 2251].)
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances, what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266].)
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266].)
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DynamicSymbol [▶ 2266].)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is a container type. (Inherited from DynamicSymbol [▶ 2266].)
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266].)
	IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266].)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266].)
	IsReference [▶ 2284]	Indicates that the Instance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)
	MemberInstances [▶ 2259]	Gets the member instances of the Struct Instance [▶ 2666]. (Inherited from DynamicStructInstance [▶ 2251].)
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
	RpcMethods [▶ 2229]	Gets the Method descriptions for the IRpcCallableType [▶ 2624]
	Size [▶ 2286]	Gets the size of the Instance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)



Methods

	Name	Description
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)
	GetDynamicMemberNames [▶ 2263]	Returns the enumeration of all dynamic member names. (Inherited from DynamicStructInstance [▶ 2251].)
	GetHashCode [▶ 2293]	Gets the HashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
 	InvokeRpcMethodAsync [▶ 2238]	Invokes the specified RPC Method asynchronously
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	ReadRawValue. [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValueAsyn c [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValueAsyn c [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject .)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryGetIndex	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicObject .)
	TryGetMember [▶ 2243]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Overrides DynamicStructInstance.TryGetMember(GetMemberBinder, Object.) [▶ 2264].)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject .)
	TryInvokeMember [▶ 2244]	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Overrides DynamicObject.TryInvokeMember(InvokeMemberBinder, .Object., Object.) .)

	Name	Description
 	TryInvokeRpcMethod(String, Object, Object.) [▶ 2245]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(String, Object, Object, Object.) [▶ 2247]	Invokes the specified RPC Method
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	TrySetIndex	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicObject .)
	TrySetMember [▶ 2250]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Overrides DynamicStructInstance.TrySetMember(SetMemberBinder, Object) [▶ 2264].)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject .)
 	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from DynamicSymbol [▶ 2266].)
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte.) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte, Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)

Events

	Name	Description
	RawValueChanged [▶ 2323]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)
	ValueChanged [▶ 2323]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)

Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

[TwinCAT.TypeSystem.DynamicSymbol](#) [▶ 2266]

[TwinCAT.TypeSystem.IStructInstance](#) [▶ 2666]














[TwinCAT.TypeSystem.IRpcStructInstance](#) [▶ 2642]

[TwinCAT.TypeSystem.IRpcCallableInstance](#) [▶ 2606]

6.11.19.1 DynamicRpcStructInstance Properties

The [DynamicRpcStructInstance](#) [▶ 2223] type exposes the following members.

Properties

	Name	Description
	InnerSymbol [▶ 2274]	Gets the inner symbol of this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	AccessRights [▶ 2274]	Gets the access rights. (Inherited from DynamicSymbol [▶ 2266].)
	Attributes [▶ 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [▶ 2266].)
	BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from DynamicSymbol [▶ 2266].)
	ByteSize [▶ 2276]	Gets the (aligned) size of the Type/Instance in Bytes (Inherited from DynamicSymbol [▶ 2266].)
	Category [▶ 2277]	Gets the category. (Inherited from DynamicSymbol [▶ 2266].)
	Comment [▶ 2277]	Gets the comment of the IInstance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266].)
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	HasRpcMethods [▶ 2258]	Gets a value indicating whether this instance has RPC methods (Inherited from DynamicStructInstance [▶ 2251].)
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances, what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266].)
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266].)
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DynamicSymbol [▶ 2266].)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from DynamicSymbol [▶ 2266].)
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266].)
	IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266].)
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266].)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266].)
	IsReference [▶ 2284]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)
	MemberInstances [▶ 2259]	Gets the member instances of the Struct Instance [▶ 2666]. (Inherited from DynamicStructInstance [▶ 2251].)
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
	RpcMethods [▶ 2229]	Gets the Method descriptions for the IRpcCallableType [▶ 2624]
	Size [▶ 2286]	Gets the size of the IInstance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)

Reference

[DynamicRpcStructInstance Class](#) [[▶ 2223](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.19.1.1 DynamicRpcStructInstance.RpcMethods Property

Gets the Method descriptions for the [IRpcCallableType](#) [[▶ 2624](#)]

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public IRpcMethodCollection RpcMethods { get; }
```

Property Value

Type: [IRpcMethodCollection](#) [▶ 2629]

The methods.

Implements

[IRpcCallableInstance.RpcMethods](#) [▶ 2607]

Reference











[DynamicRpcStructInstance Class](#) [▶ 2223]

















[TwinCAT.TypeSystem Namespace](#) [▶ 2083]
















6.11.19.2 DynamicRpcStructInstance Methods

The [DynamicRpcStructInstance](#) [▶ 2223] type exposes the following members.

Methods

	Name	Description
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)
	GetDynamicMemberNames [▶ 2263]	Returns the enumeration of all dynamic member names. (Inherited from DynamicStructInstance [▶ 2251].)
	GetHashCode [▶ 2293]	Gets the HashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
 	InvokeRpcMethodAsync [▶ 2238]	Invokes the specified RPC Method asynchronously
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	ReadRawValueAsyn c [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject.)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject.)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetIndex	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicObject.)
	TryGetMember [▶ 2243]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Overrides DynamicStructInstance.TryGetMember(GetMemberBinder, Object.) [▶ 2264].)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject.)
	TryInvokeMember [▶ 2244]	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Overrides DynamicObject.TryInvokeMember(InvokeMemberBinder, Object, Object.) .)
 	TryInvokeRpcMetho d(String, Object, Object.) [▶ 2245]	Invokes the specified RPC Method

	Name	Description
 	TryInvokeRpcMethod(String, Object, Object, Object) [2247]	Invokes the specified RPC Method
	TryReadValue [2309]	Reads the Value of the IValueSymbol [2775] (Inherited from DynamicSymbol [2266].)
	TrySetIndex	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicObject .)
	TrySetMember [2250]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Overrides DynamicStructInstance.TrySetMember(SetMemberBinder, Object) [2264].)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject .)
 	TryWriteValue [2309]	Writes the specified value to the DynamicSymbol [2266]. (Inherited from DynamicSymbol [2266].)
	UpdateAnyValue [2312]	Reads the value of this Value [2775] into the specified managed value. (Inherited from DynamicSymbol [2266].)
	WriteAnyValue [2313]	Writes the value represented by the managed value to this Value [2775] (Inherited from DynamicSymbol [2266].)
	WriteRawValue(Byte) [2314]	Writes the raw value of the IValueSymbol [2775] (Ads Read / Write) (Inherited from DynamicSymbol [2266].)
	WriteRawValue(Byte, Int32) [2314]	Writes the raw value of the IValueSymbol [2775] (Ads Read / Write) (Inherited from DynamicSymbol [2266].)
	WriteRawValueAsync [2315]	Writes the raw value of the IValueSymbol [2775] (Ads Read / Write) (Inherited from DynamicSymbol [2266].)
	WriteValue(Object) [2316]	Writes the specified value to the DynamicSymbol [2266]. (Inherited from DynamicSymbol [2266].)
 	WriteValue(Object, Int32) [2319]	Writes the specified value to the DynamicSymbol [2266]. (Inherited from DynamicSymbol [2266].)
	WriteValueAsync [2321]	Writes the Value of the IValueSymbol [2775] (Inherited from DynamicSymbol [2266].)

Reference

[DynamicRpcStructInstance Class](#) [[2223](#)]

[TwinCAT.TypeSystem Namespace](#) [[2083](#)]

6.11.19.2.1 DynamicRpcStructInstance.GetDynamicMemberNames Method

Gets the dynamic member names.

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public override IEnumerable<string> GetDynamicMemberNames ()
```

Return Value

Type: [IEnumerable.String](#).
[IEnumerable<System.String>](#).




Reference

[DynamicRpcStructInstance Class](#) [[▸ 2223](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.19.2.2 DynamicRpcStructInstance.InvokeRpcMethod Method

Overload List

	Name	Description
	InvokeRpcMethod(String, .Object.) [▸ 2233]	Invokes the specified RPC Method
	InvokeRpcMethod(String, .Object., .Object.) [▸ 2235]	Invokes the specified RPC Method
	InvokeRpcMethod(String, .Object., .AnyTypeSpecifier., .AnyTypeSpecifier., .Object.) [▸ 2237]	Invokes the specified RPC Method

Reference

[DynamicRpcStructInstance Class](#) [[▸ 2223](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.19.2.2.1 DynamicRpcStructInstance.InvokeRpcMethod Method (String, .Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Object InvokeRpcMethod(
    string methodName,
    Object[] inParameters
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The input parameters or NULL

Return Value

Type: [Object](#)
The return value of the Method (as object).

Implements

[IRpcCallableInstance.InvokeRpcMethod\(String, .Object.\)](#) [▶ 2609]

Remarks

This method only supports primitive data types as inParameters. Any available outparameters will be ignored. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */
        }
    }
}
```

```
short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

// Call a Method that has no parameter and returns VOID
main.InvokeRpcMethod("M_Method1", new object[] {});

//Browsing RpcMethods
foreach(IRpcMethod method in main.RpcMethods)
{
    string methodName = method.Name;

    foreach(IRpcMethodParameter parameter in method.Parameters)
    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
```

Reference

[DynamicRpcStructInstance Class](#) [► 2223]

[InvokeRpcMethod Overload](#) [► 2233]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.19.2.2 DynamicRpcStructInstance.InvokeRpcMethod Method (String, .Object., .Object..)

Invokes the specified RPC Method

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934 addedca3e72bc0ea15da1c14

Syntax

C#

```
public Object InvokeRpcMethod(
    string methodName,
    Object[] inParameters,
    out Object[] outParameters
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The input parameters or NULL
outParameters	Type: .System.Object.. The output parameters.

Return Value

Type: [Object](#)

The return value of the Method (as object).

Implements

[IRpcCallableInstance.InvokeRpcMethod\(String, .Object., .Object..\) \[► 2610\]](#)

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier \[▸ 2094\]](#) specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```

Reference

[DynamicRpcStructInstance Class \[▸ 2223\]](#)

[InvokeRpcMethod Overload \[▸ 2233\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.19.2.2.3 DynamicRpcStructInstance.InvokeRpcMethod Method (String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object..)

Invokes the specified RPC Method

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Object InvokeRpcMethod(  
    string methodName,  
    Object[] inParameters,  
    AnyTypeSpecifier[] outSpecifiers,  
    AnyTypeSpecifier retSpecifier,  
    out Object[] outParameters  
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object. The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094]. The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094] The ret specifier (specifying the return value) or NULL.
outParameters	Type: .System.Object. The out parameters.

Return Value

Type: [Object](#)
The return value of the Method (as object).

Implements

[IRpcCallableInstance.InvokeRpcMethod\(String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object..\)](#)
[\[▶ 2612\]](#)

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram  
{  
    /// <summary>  
    /// Defines the entry point of the application.  
    /// </summary>  
    /// <param name="args">The arguments.</param>
```

```

static void Main(string[] args)
{
    // Get the AdsAddress from command-line arguments
    AmsAddress address = ArgParser.Parse(args);

    using (AdsClient client = new AdsClient())
    {
        //client.Synchronize = false;

        // Connect to the target device
        client.Connect(address);

        SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.VirtualTree);
        ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

        // Get the Symbols (Dynamic Symbols)

        IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

        // Call a Method that has the following signature (within MAIN Program)
        /* {attribute 'TcRpcEnable'}
        METHOD PUBLIC M_Add : INT
        VAR_INPUT
            i1 : INT := 0;
            i2 : INT := 0;
        END_VAR
        */

        short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

        // Call a Method that has no parameter and returns VOID
        main.InvokeRpcMethod("M_Method1", new object[] {});

        //Browsing RpcMethods
        foreach(IRpcMethod method in main.RpcMethods)
        {
            string methodName = method.Name;

            foreach(IRpcMethodParameter parameter in method.Parameters)
            {
                string parameterName = parameter.Name;
                string parameterType = parameter.TypeName;
            }
        }
    }
}

```

Reference

[DynamicRpcStructInstance Class \[► 2223\]](#)

[InvokeRpcMethod Overload \[► 2233\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.19.2.3 DynamicRpcStructInstance.InvokeRpcMethodAsync Method

Invokes the specified RPC Method asynchronously

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```

public Task<ResultRpcMethodAccess> InvokeRpcMethodAsync(
    string methodName,
    Object[]? inParameters,
    CancellationToken cancel
)

```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
cancel	Type: System.Threading.CancellationToken The cancellation token

Return Value

Type: [Task.ResultRpcMethodAccess](#) [▸ 3213].

A task that represents the asynchronous 'InvokeRpcMethod' operation. The [ResultRpcMethodAccess](#) [▸ 3213] results contains the return value ([ReturnValue](#) [▸ 3216]) together with the output parameters. The succeeded communication is indicated by the [ErrorCode](#) property ([ErrorCode](#) [▸ 3202]) after the communication.

Implements

[IRpcCallableInstance.InvokeRpcMethodAsync\(String, .Object., CancellationToken\)](#) [▸ 2614]

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [▸ 2094] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] { (short) 3, (short) 4 });

            // Call a Method that has no parameter and returns VOID
        }
    }
}
```

```

main.InvokeRpcMethod("M_Method1", new object[] {});

//Browsing RpcMethods
foreach(IRpcMethod method in main.RpcMethods)
{
    string methodName = method.Name;

    foreach(IRpcMethodParameter parameter in method.Parameters)
    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
}

```

Reference

[DynamicRpcStructInstance Class](#) [► 2223]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.19.2.3.1 DynamicRpcStructInstance.InvokeRpcMethodAsync Method (String, .Object., CancellationToken)

Invokes the specified RPC Method asynchronously

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```

public Task<ResultRpcMethodAccess> InvokeRpcMethodAsync(
    string methodName,
    Object[] inParameters,
    CancellationToken cancel
)

```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
cancel	Type: System.Threading.CancellationToken The cancellation token

Return Value

Type: [Task.ResultRpcMethodAccess](#) [► 3213].

A task that represents the asynchronous 'InvokeRpcMethod' operation. The [ResultRpcMethodAccess](#) [► 3213] results contains the return value ([ReturnValue](#) [► 3216]) together with the output parameters. The succeeded communication is indicated by the [ErrorCode](#) property ([ErrorCode](#) [► 3202]) after the communication.

Implements

[IRpcCallableInstance.InvokeRpcMethodAsync\(String, .Object., CancellationToken\)](#) [► 2615]

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier \[▸ 2094\]](#) specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```

Reference

[DynamicRpcStructInstance Class \[▸ 2223\]](#)

[InvokeRpcMethodAsync Overload \[▸ 2238\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.19.2.3.2 DynamicRpcStructInstance.InvokeRpcMethodAsync Method (String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, CancellationToken)

Invokes the specified RPC Method asynchronously

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public Task<ResultRpcMethodAccess> InvokeRpcMethodAsync (
    string methodName,
    Object[] inParameters,
    AnyTypeSpecifier[] outSpecifiers,
    AnyTypeSpecifier retSpecifier,
    CancellationToken cancel
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094]. The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094] The ret specifier (specifying the return value) or NULL.
cancel	Type: System.Threading.CancellationToken The cancellation token

Return Value

Type: [Task.ResultRpcMethodAccess](#) [▶ 3213].

A task that represents the asynchronous 'InvokeRpcMethod' operation. The [ResultRpcMethodAccess](#) [▶ 3213] results contains the return value ([ReturnValue](#) [▶ 3216]) together with the output parameters. The succeeded communication is indicated by the [ErrorCode](#) property ([ErrorCode](#) [▶ 3202]) after the communication.

Implements

[IRpcCallableInstance.InvokeRpcMethodAsync\(String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, CancellationToken\)](#) [▶ 2617]

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```

class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}

```

Reference

[DynamicRpcStructInstance Class \[► 2223\]](#)

[InvokeRpcMethodAsync Overload \[► 2238\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.19.2.4 DynamicRpcStructInstance.TryGetMember Method

Provides the implementation for operations that get member values. Classes derived from the [DynamicObject](#) class can override this method to specify dynamic behavior for operations such as getting a value for a property.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public override bool TryGetMember(  
    GetMemberBinder binder,  
    out Object?? result  
)
```

Parameters

binder	Type: System.Dynamic.GetMemberBinder Provides information about the object that called the dynamic operation. The binder.Name property provides the name of the member on which the dynamic operation is performed. For example, for the Console.WriteLine(sampleObject.SampleProperty) statement, where sampleObject is an instance of the class derived from the DynamicObject class, binder.Name returns "SampleProperty". The binder.IgnoreCase property specifies whether the member name is case-sensitive.
result	Type: System.Object . The result of the get operation. For example, if the method is called for a property, you can assign the property value to result.

Return Value

Type: [Boolean](#)
true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a run-time exception is thrown.)

Reference

[DynamicRpcStructInstance Class](#) [► 2223]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.19.2.5 DynamicRpcStructInstance.TryInvokeMember Method

Provides the implementation for operations that invoke a member. Classes derived from the [DynamicObject](#) class can override this method to specify dynamic behavior for operations such as calling a method.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public override bool TryInvokeMember(  
    InvokeMemberBinder binder,  
    Object?[]? args,  
    out Object?? returnValue  
)
```

Parameters

binder	Type: System.Dynamic.InvokeMemberBinder Provides information about the dynamic operation. The binder.Name property provides the name of the member on which the dynamic operation is performed. For example, for the statement sampleObject.SampleMethod(100), where sampleObject is an instance of the class derived from the DynamicObject class, binder.Name returns "SampleMethod". The binder.IgnoreCase property specifies whether the member name is case-sensitive.
--------	--

args Type: .System.Object.
 The arguments that are passed to the object member during the invoke operation. For example, for the statement sampleObject.SampleMethod(100), where sampleObject is derived from the DynamicObject class, args[] is equal to 100.

returnValue Type: System.Object.
 The result of the member invocation.

Return Value



Type: Boolean
 true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a language-specific run-time exception is thrown.)

Reference

- [DynamicRpcStructInstance Class \[▶ 2223\]](#)
- [TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.19.2.6 DynamicRpcStructInstance.TryInvokeRpcMethod Method

Overload List

	Name	Description
	TryInvokeRpcMethod(String, .Object., Object.) [▶ 2245]	Invokes the specified RPC Method
	TryInvokeRpcMethod(String, .Object., Object., Object.) [▶ 2247]	Invokes the specified RPC Method

Reference

- [DynamicRpcStructInstance Class \[▶ 2223\]](#)
- [TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.19.2.6.1 DynamicRpcStructInstance.TryInvokeRpcMethod Method (String, .Object., Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public int TryInvokeRpcMethod(
    string methodName,
    Object[]? inParameters,
    out Object?? retValue
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
retValue	Type: System.Object . The return value of the RPC method as object.

Return Value

Type: [Int32](#)
The return value of the Method (as object).

Implements

[IRpcCallableInstance.TryInvokeRpcMethod\(String, .Object., Object.\)](#) [► 2619]

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [► 2094] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] { (short) 3, (short) 4 });

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
        }
    }
}
```

```
foreach(IRpcMethod method in main.RpcMethods)
{
    string methodName = method.Name;

    foreach(IRpcMethodParameter parameter in method.Parameters)
    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
```

Reference

[DynamicRpcStructInstance Class](#) [► 2223]

[TryInvokeRpcMethod Overload](#) [► 2245]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.19.2.6.2 DynamicRpcStructInstance.TryInvokeRpcMethod Method (String, .Object., .Object., Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public int TryInvokeRpcMethod(
    string methodName,
    Object[]? inParameters,
    out Object[]? outParameters,
    out Object?? retValue
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object. The parameters.
outParameters	Type: .System.Object.. The out parameters.
retValue	Type: System.Object. The return value of the RPC method as object.

Return Value

Type: [Int32](#)
The ADS Error Code.

Implements

[IRpcCallableInstance.TryInvokeRpcMethod\(String, .Object., .Object., Object.\)](#) [► 2620]

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier \[▸ 2094\]](#) specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```

Reference

[DynamicRpcStructInstance Class \[▸ 2223\]](#)

[TryInvokeRpcMethod Overload \[▸ 2245\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.19.2.6.3 DynamicRpcStructInstance.TryInvokeRpcMethod Method (String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object., Object.)

Invokes the the specified RpcMethod of the [IRpcCallableInstance](#) [▶ 2606].

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public int TryInvokeRpcMethod(  
    string methodName,  
    Object[] inParameters,  
    AnyTypeSpecifier[] outSpecifiers,  
    AnyTypeSpecifier retSpecifier,  
    out Object[] outParameters,  
    out Object retVal  
)
```

Parameters

methodName	Type: System.String Name of the method.
inParameters	Type: .System.Object . The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094]. The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [▶ 2094] The ret specifier (specifying the return value) or NULL.
outParameters	Type: .System.Object . The out parameters.
retValue	Type: System.Object . The return value of the RPC method./>

Return Value

Type: [Int32](#)
AdsErrorCode.

Implements

[IRpcCallableInstance.TryInvokeRpcMethod\(String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object., Object.\)](#) [▶ 2622]

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```

class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}

```

Reference

[DynamicRpcStructInstance Class \[► 2223\]](#)

[TryInvokeRpcMethod Overload \[► 2245\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.19.2.7 DynamicRpcStructInstance.TrySetMember Method

Provides the implementation for operations that set member values. Classes derived from the [DynamicObject](#) class can override this method to specify dynamic behavior for operations such as setting a value for a property.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public override bool TrySetMember(
    SetMemberBinder binder,
    Object? value
)
```

Parameters

- binder** Type: [System.Dynamic.SetMemberBinder](#)
 Provides information about the object that called the dynamic operation. The binder.Name property provides the name of the member to which the value is being assigned. For example, for the statement sampleObject.SampleProperty = "Test", where sampleObject is an instance of the class derived from the [DynamicObject](#) class, binder.Name returns "SampleProperty". The binder.IgnoreCase property specifies whether the member name is case-sensitive.
- value** Type: [System.Object](#)
 The value to set to the member. For example, for sampleObject.SampleProperty = "Test", where sampleObject is an instance of the class derived from the [DynamicObject](#) class, the value is "Test".

Return Value

Type: [Boolean](#)
 true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a language-specific run-time exception is thrown.)



Reference

- [DynamicRpcStructInstance Class \[▶ 2223\]](#)
- [TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.19.3 DynamicRpcStructInstance Events

The [DynamicRpcStructInstance \[▶ 2223\]](#) type exposes the following members.

Events

	Name	Description
	RawValueChanged [▶ 2323]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266] .)
	ValueChanged [▶ 2323]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266] .)

Reference

- [DynamicRpcStructInstance Class \[▶ 2223\]](#)
- [TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.20 DynamicStructInstance Class

Dynamic struct instance

Inheritance Hierarchy

System.Object

System.Dynamic.DynamicObject

TwinCAT.TypeSystem.DynamicSymbol [[▶ 2266](#)]

TwinCAT.TypeSystem.DynamicInterfaceInstance [[▶ 2995](#)]

TwinCAT.TypeSystem.DynamicStructInstance

TwinCAT.TypeSystem.DynamicVirtualStructInstance [[▶ 2374](#)]

Namespace: TwinCAT.TypeSystem [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229
















Syntax

C#

```
public class DynamicStructInstance : DynamicInterfaceInstance,
    IRpcStructInstance, IStructInstance, IInterfaceInstance, ISymbol, IAttributedInstance,
    IInstance, IBitSize, IRpcCallableInstance
```





The DynamicStructInstance type exposes the following members.






















Properties



















	Name	Description
	InnerSymbol [▶ 2274]	Gets the inner symbol of this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	AccessRights [▶ 2274]	Gets the access rights. (Inherited from DynamicSymbol [▶ 2266].)
	AllowGIOAccess [▶ 2275]	Indicates, that the aggregates symbols is an IProcessImageAddress (and most probably IAdsSymbol) (Inherited from DynamicSymbol [▶ 2266].)
	Attributes [▶ 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [▶ 2266].)
	BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from DynamicSymbol [▶ 2266].)
	ByteSize [▶ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [▶ 2266].)
	Category [▶ 2277]	Gets the category. (Inherited from DynamicSymbol [▶ 2266].)
	Comment [▶ 2277]	Gets the comment of the IInstance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266].)
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	HasRpcMethods [▶ 3001]	Gets a value indicating whether this instance has RPC methods (Inherited from DynamicInterfaceInstance [▶ 2995].)
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances, what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266].)
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266].)















	Name	Description
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DynamicSymbol [▶ 2266].)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from DynamicSymbol [▶ 2266].)
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266].)
	IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266].)
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266].)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266].)
	IsReference [▶ 2284]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)
	MemberInstances [▶ 3002]	Gets the member instances of the Struct Instance [▶ 2666]. (Inherited from DynamicInterfaceInstance [▶ 2995].)
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
	RpcMethods [▶ 3003]	Gets the Method descriptions for the IRpcCallableType [▶ 2624] (Inherited from DynamicInterfaceInstance [▶ 2995].)
	Size [▶ 2286]	Gets the size of the IInstance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)

Methods



	Name	Description
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetDynamicMemberNames [▶ 3006]	Returns the enumeration of all dynamic member names. (Inherited from DynamicInterfaceInstance [▶ 2995].)
	GetHashCode [▶ 2293]	Gets the HashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
 	InvokeRpcMethodAsync [▶ 3007]	Invokes the specified RPC Method asynchronously (Inherited from DynamicInterfaceInstance [▶ 2995].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	OnReadAnyValue [▶ 2293]	Handler function for reading ADS 'Any' Values. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadRawValue [▶ 2294]	Handler function for reading Raw symbol value. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadRawValueAsync [▶ 2294]	Handler function reading the raw value of the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadValue [▶ 2295]	Handler function for the (Inherited from DynamicSymbol [▶ 2266].)
	OnReadValueAsync [▶ 2296]	Handler function reading the DynamicSymbols [▶ 2266] value asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	OnSetInstanceName [▶ 2296]	Sets a new InstanceName InstancePath (Inherited from DynamicSymbol [▶ 2266].)
	OnTryReadValue [▶ 2297]	Handler function for the (Inherited from DynamicSymbol [▶ 2266].)
	OnTryWriteValue [▶ 2297]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteRawValue [▶ 2298]	Handler function for reading symbols raw value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteRawValueAsync [▶ 2298]	Handler function for writing the raw DynamicSymbol [▶ 2266] value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteValue [▶ 2299]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteValueAsync [▶ 2300]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue. [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValueAsync [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)


	Name	Description
	ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject .)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryGetIndex	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicObject .)
	TryGetMember [▶ 3008]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Inherited from DynamicInterfaceInstance [▶ 2995].)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject .)
	TryInvokeMember [▶ 3009]	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicInterfaceInstance [▶ 2995].)
 	TryInvokeRpcMethod(String, Object, Object) [▶ 3010]	Invokes the specified RPC Method (Inherited from DynamicInterfaceInstance [▶ 2995].)
 	TryInvokeRpcMethod(String, Object, Object, Object) [▶ 3012]	Invokes the specified RPC Method (Inherited from DynamicInterfaceInstance [▶ 2995].)
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	TrySetIndex	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicObject.)
	TrySetMember [▶ 3013]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicInterfaceInstance [▶ 2995].)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
 	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from DynamicSymbol [▶ 2266].)
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(.Byte) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(.Byte, Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)

Events

	Name	Description
	RawValueChanged [▶ 2323]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)
	ValueChanged [▶ 2323]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)

Fields

	Name	Description
	syncObject [▶ 2327]	Synchronization object (Inherited from DynamicSymbol [▶ 2266].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
















Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.20.1 DynamicStructInstance Properties

The [DynamicStructInstance](#) [[▶ 2251](#)] type exposes the following members.

Properties

	Name	Description
	InnerSymbol [▶ 2274]	Gets the inner symbol of this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	AccessRights [▶ 2274]	Gets the access rights. (Inherited from DynamicSymbol [▶ 2266].)
	AllowGIOAccess [▶ 2275]	Indicates, that the aggregates symbols is an IProcessImageAddress (and most probably IAdsSymbol) (Inherited from DynamicSymbol [▶ 2266].)
	Attributes [▶ 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [▶ 2266].)
	BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from DynamicSymbol [▶ 2266].)
	ByteSize [▶ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [▶ 2266].)
	Category [▶ 2277]	Gets the category. (Inherited from DynamicSymbol [▶ 2266].)
	Comment [▶ 2277]	Gets the comment of the Instance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266].)
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	HasRpcMethods [▶ 3001]	Gets a value indicating whether this instance has RPC methods (Inherited from DynamicInterfaceInstance [▶ 2995].)
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances , what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266].)
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266].)
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned ($BitSize \% 8 == 0$) (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is a container type. (Inherited from DynamicSymbol [▶ 2266].)
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266].)
	IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266].)
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266].)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266].)
	IsReference [▶ 2284]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)
	MemberInstances [▶ 3002]	Gets the member instances of the Struct Instance [▶ 2666]. (Inherited from DynamicInterfaceInstance [▶ 2995].)
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
	RpcMethods [▶ 3003]	Gets the Method descriptions for the IRpcCallableType [▶ 2624] (Inherited from DynamicInterfaceInstance [▶ 2995].)
	Size [▶ 2286]	Gets the size of the IInstance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)

Reference

[DynamicStructInstance Class](#) [▶ 2251]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.20.1.1 DynamicStructInstance.HasRpcMethods Property

Gets a value indicating whether this instance has RPC methods

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public bool HasRpcMethods { get; }
```

Property Value

Type: [Boolean](#)

true if this instance has RPC methods; otherwise, false.

Implements

[IStructInstance.HasRpcMethods](#) [[▶ 2670](#)]

Remarks

If the struct instance supports RPC Methods, then the instance class is also supporting [IRpcStructInstance](#) [[▶ 2642](#)]

Reference

[DynamicStructInstance Class](#) [[▶ 2251](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

[TwinCAT.TypeSystem.IRpcStructInstance](#) [[▶ 2642](#)]

[TwinCAT.TypeSystem.IRpcMethod](#) [[▶ 2625](#)]

[TwinCAT.TypeSystem.IRpcMethodParameter](#) [[▶ 2635](#)]

6.11.20.1.2 DynamicStructInstance.MemberInstances Property

Gets the member instances of the [Struct Instance](#) [[▶ 2666](#)].

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public ISymbolCollection<ISymbol> MemberInstances { get; }
```

Property Value

Type: [ISymbolCollection](#) [[▶ 2700](#)].[ISymbol](#) [[▶ 2691](#)].

The member instances.

Implements

[IStructInstance.MemberInstances](#) [[▶ 2670](#)]

Reference

[DynamicStructInstance Class](#) [[▶ 2251](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.20.1.3 DynamicStructInstance.NormalizedDict Property

Dictionary of normalized Instance Names

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
protected IDictionary<string, ISymbol> NormalizedDict { get; }
```

Property Value

Type: [IDictionary.String](#), [ISymbol](#) [▶ 2691].

Reference












[DynamicStructInstance Class](#) [▶ 2251]




















[TwinCAT.TypeSystem Namespace](#) [▶ 2083]



















6.11.20.2 DynamicStructInstance Methods









The [DynamicStructInstance](#) [▶ 2251] type exposes the following members.

Methods

	Name	Description
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetDynamicMemberNames [▶ 3006]	Returns the enumeration of all dynamic member names. (Inherited from DynamicInterfaceInstance [▶ 2995].)
	GetHashCode [▶ 2293]	Gets the HashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
 	InvokeRpcMethodAsync [▶ 3007]	Invokes the specified RPC Method asynchronously (Inherited from DynamicInterfaceInstance [▶ 2995].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	OnReadAnyValue [▶ 2293]	Handler function for reading ADS 'Any' Values. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadRawValue [▶ 2294]	Handler function for reading Raw symbol value. (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	OnReadRawValueAsync [▶ 2294]	Handler function reading the raw value of the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadValue [▶ 2295]	Handler function for the (Inherited from DynamicSymbol [▶ 2266].)
	OnReadValueAsync [▶ 2296]	Handler function reading the DynamicSymbols [▶ 2266] value asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	OnSetInstanceName [▶ 2296]	Sets a new InstanceName InstancePath (Inherited from DynamicSymbol [▶ 2266].)
	OnTryReadValue [▶ 2297]	Handler function for the (Inherited from DynamicSymbol [▶ 2266].)
	OnTryWriteValue [▶ 2297]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteRawValue [▶ 2298]	Handler function for reading symbols raw value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteRawValueAsync [▶ 2298]	Handler function for writing the raw DynamicSymbol [▶ 2266] value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteValue [▶ 2299]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteValueAsync [▶ 2300]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue. [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValueAsync [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject .)

	Name	Description
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetIndex	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicObject.)
	TryGetMember ▶ 3008	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Inherited from DynamicInterfaceInstance ▶ 2995 .)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject.)
	TryInvokeMember ▶ 3009	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicInterfaceInstance ▶ 2995 .)
 	TryInvokeRpcMethod(String, Object, Object.) ▶ 3010	Invokes the specified RPC Method (Inherited from DynamicInterfaceInstance ▶ 2995 .)
 	TryInvokeRpcMethod(String, Object, Object, Object.) ▶ 3012	Invokes the specified RPC Method (Inherited from DynamicInterfaceInstance ▶ 2995 .)
	TryReadValue ▶ 2309	Reads the Value of the IValueSymbol ▶ 2775 (Inherited from DynamicSymbol ▶ 2266 .)
	TrySetIndex	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicObject.)
	TrySetMember ▶ 3013	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicInterfaceInstance ▶ 2995 .)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
 	TryWriteValue ▶ 2309	Writes the specified value to the DynamicSymbol ▶ 2266 . (Inherited from DynamicSymbol ▶ 2266 .)
	UpdateAnyValue ▶ 2312	Reads the value of this Value ▶ 2775 into the specified managed value. (Inherited from DynamicSymbol ▶ 2266 .)

	Name	Description
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte, Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[DynamicStructInstance Class](#) [▶ 2251]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.20.2.1 DynamicStructInstance.GetDynamicMemberNames Method

Returns the enumeration of all dynamic member names.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public override IEnumerable<string> GetDynamicMemberNames()
```

Return Value

Type: [IEnumerable.String](#).

A sequence that contains dynamic member names.

Reference

[DynamicStructInstance Class](#) [▶ 2251]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.20.2.2 **DynamicStructInstance.TryGetMember Method**

Provides the implementation for operations that get member values. Classes derived from the [DynamicObject](#) class can override this method to specify dynamic behavior for operations such as getting a value for a property.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public override bool TryGetMember(  
    GetMemberBinder binder,  
    out Object?? result  
)
```

Parameters

binder	Type: System.Dynamic.GetMemberBinder Provides information about the object that called the dynamic operation. The binder.Name property provides the name of the member on which the dynamic operation is performed. For example, for the Console.WriteLine(sampleObject.SampleProperty) statement, where sampleObject is an instance of the class derived from the DynamicObject class, binder.Name returns "SampleProperty". The binder.IgnoreCase property specifies whether the member name is case-sensitive.
result	Type: System.Object . The result of the get operation. For example, if the method is called for a property, you can assign the property value to result.

Return Value

Type: [Boolean](#)

true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a run-time exception is thrown.)

Reference

[DynamicStructInstance Class](#) [[▶ 2251](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.20.2.3 **DynamicStructInstance.TrySetMember Method**

Provides the implementation for operations that set member values. Classes derived from the [DynamicObject](#) class can override this method to specify dynamic behavior for operations such as setting a value for a property.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public override bool TrySetMember(  
    SetMemberBinder binder,  
    Object? value  
)
```


Parameters

- binder Type: [System.Dynamic.SetMemberBinder](#)
 Provides information about the object that called the dynamic operation. The binder.Name property provides the name of the member to which the value is being assigned. For example, for the statement sampleObject.SampleProperty = "Test", where sampleObject is an instance of the class derived from the [DynamicObject](#) class, binder.Name returns "SampleProperty". The binder.IgnoreCase property specifies whether the member name is case-sensitive.
- value Type: [System.Object](#)
 The value to set to the member. For example, for sampleObject.SampleProperty = "Test", where sampleObject is an instance of the class derived from the [DynamicObject](#) class, the value is "Test".

Return Value

Type: [Boolean](#)
 true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a language-specific run-time exception is thrown.)



Reference

- [DynamicStructInstance Class \[▶ 2251\]](#)
- [TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.20.3 DynamicStructInstance Events

The [DynamicStructInstance \[▶ 2251\]](#) type exposes the following members.

Events

	Name	Description
	RawValueChanged [▶ 2323]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266] .)
	ValueChanged [▶ 2323]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266] .)


Reference

- [DynamicStructInstance Class \[▶ 2251\]](#)
- [TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.20.4 DynamicStructInstance Fields

The [DynamicStructInstance \[▶ 2251\]](#) type exposes the following members.

Fields

	Name	Description
	syncObject [▶ 2327]	Synchronization object (Inherited from DynamicSymbol [▶ 2266] .)

Reference

- [DynamicStructInstance Class \[▶ 2251\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.21 DynamicSymbol Class

Dynamic [Symbol \[▸ 2691\]](#) object.

Inheritance Hierarchy

System.Object
 System.Dynamic.DynamicObject
 TwinCAT.TypeSystem.DynamicSymbol
[More... \[▸ 2272\]](#)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229















Syntax

















C#

```
public class DynamicSymbol : DynamicObject,
    IDynamicSymbol, ISymbol, IAttributedInstance, IInstance, IBitSize,
    ISymbolFactoryServicesProvider, IValueSymbol, IValueRawSymbol, IHierarchicalSymbol, IValueAccess
    orProvider,
    IContextMaskProvider
```







The DynamicSymbol type exposes the following members.






















Properties

	Name	Description
	InnerSymbol [▸ 2274]	Gets the inner symbol of this DynamicSymbol
	AccessRights [▸ 2274]	Gets the access rights.
	AllowGIOAccess [▸ 2275]	Indicates, that the aggregates symbols is an IProcessImageAddress (and most probably IAdsSymbol)
	Attributes [▸ 2275]	Gets the Symbol Attributes
	BitSize [▸ 2276]	Gets the size of the IDataType [▸ 2475] in bits.
	ByteSize [▸ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes
	Category [▸ 2277]	Gets the category.
	Comment [▸ 2277]	Gets the comment of the Instance [▸ 2549]
	Connection [▸ 2277]	Gets the connection bound to this DynamicSymbol
	DataType [▸ 2278]	Gets the IDataType [▸ 2475] of the Instance [▸ 2549] .
	HasValue [▸ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances, what means the Symbols with locations).
	InstanceName [▸ 2279]	Gets the name of the instance (without periods (.))
	InstancePath [▸ 2279]	Gets the relative / absolute access path to the instance (with periods (.))
	IsBitType [▸ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping







	Name	Description
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is a container type.
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent.
	IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type.
	IsReadOnly [▶ 2283]	Indicates that this instance is read only.
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive.
	IsReference [▶ 2284]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static.
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters),
	NotificationSettings [▶ 2285]	Gets the notification settings.
	Parent [▶ 2286]	Gets the parent Symbol
	Size [▶ 2286]	Gets the size of the IInstance [▶ 2549] in bytes.
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691]
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549].
	ValueEncoding [▶ 2288]	Gets the value encoding.

Methods



	Name	Description
	Equals [▶ 2292]	Equals (Overrides Object.Equals(Object).)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetDynamicMemberNames [▶ 2293]	Returns the enumeration of all dynamic member names. (Overrides DynamicObject.GetDynamicMemberNames..)
	GetHashCode [▶ 2293]	Gets the GetHashCode of the Address (Overrides Object.GetHashCode..)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)

	Name	Description
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	OnReadAnyValue [▶ 2293]	Handler function for reading ADS 'Any' Values.
	OnReadRawValue [▶ 2294]	Handler function for reading Raw symbol value.
	OnReadRawValueAsync [▶ 2294]	Handler function reading the raw value of the DynamicSymbol.
	OnReadValue [▶ 2295]	Handler function for the
	OnReadValueAsync [▶ 2296]	Handler function reading the DynamicSymbols value asynchronously.
	OnSetInstanceName [▶ 2296]	Sets a new InstanceName InstancePath
	OnTryReadValue [▶ 2297]	Handler function for the
	OnTryWriteValue [▶ 2297]	Handler Function for writing value.
	OnWriteRawValue [▶ 2298]	Handler function for reading symbols raw value.
	OnWriteRawValueAsync [▶ 2298]	Handler function for writing the raw DynamicSymbol value.
	OnWriteValue [▶ 2299]	Handler Function for writing value.
	OnWriteValueAsync [▶ 2300]	Handler Function for writing value.
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type
	ReadRawValue. [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value
	ReadRawValueAsync [▶ 2302]	Read raw value as an asynchronous operation.
	ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol.
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol.
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously.
	ToString [▶ 2307]	Returns a String that represents this instance. (Overrides Object.ToString..)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject.)





	Name	Description
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject.)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetIndex	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicObject.)
	TryGetMember [▶ 2308]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Overrides DynamicObject.TryGetMember(GetMemberBinder, Object).)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject.)
	TryInvokeMember	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicObject.)
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775]
	TrySetIndex	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicObject.)
	TrySetMember	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicObject.)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
 	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol.
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value.
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775]
	WriteRawValue(Byte. e.) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)

	Name	Description
	WriteRawValue(Byte, Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol.
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol.
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775]


Events

	Name	Description
	RawValueChanged [▶ 2323]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed.
	ValueChanged [▶ 2323]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed.





Operators

	Name	Description
 	Equality [▶ 2325]	Operator==
 	Inequality [▶ 2326]	Implements the != operator.


Fields

	Name	Description
	syncObject [▶ 2327]	Synchronization object

Extension Methods

	Name	Description
	PollValues(IObservableUnit) [▶ 1355]	Overloaded. Poll symbol values on trigger signals. (Defined by ValueSymbolExtensions [▶ 1333] .)
 	PollValues(TimeSpan) [▶ 1356]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by ValueSymbolExtensions [▶ 1333] .)
	PollValues(IObservableUnit, Boolean) [▶ 1357]	Overloaded. Polls symbol values on trigger signals. (Defined by ValueSymbolExtensions [▶ 1333] .)

	Name	Description
	PollValues(TimeSpan, Boolean) [▶ 1358]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues.T.(IObservable.Unit.) [▶ 1362]	Overloaded. Poll symbol values as a value sequence on trigger signals (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues.T.(TimeSpan) [▶ 1363]	Overloaded. Polls the symbol as value sequence of object values with a specified period time (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues.T.(IObservable.Unit., Func.ResultReadValueAccess2.IValueSymbol, Object., T.) [▶ 1365]	Overloaded. Poll symbol values on trigger signals (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues.T.(IObservable.Unit., Boolean) [▶ 1364]	Overloaded. Poll symbol values on trigger signals (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues.T.(TimeSpan, Boolean) [▶ 1367]	Overloaded. Polls the symbol as value sequence of object values with a specified period time (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues2(IObservable.Unit.) [▶ 1368]	Overloaded. Poll symbol values as a sequence of annotated results (Value + ErrorCode) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues2(TimeSpan) [▶ 1369]	Overloaded. Poll symbol values with communication return codes. (Defined by ValueSymbolExtensions [▶ 1333].)
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	WhenValueChanged [▶ 1344]	Gets an observable sequence when the value of the IValueSymbol [▶ 2775] has changed. (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues(IObservable.Object.) [▶ 1347]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775]. (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues(IObservable.Object., Action.Exception.) [▶ 1348]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775]. (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues(IObservable.Object., CancellationToken) [▶ 1349]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775]. (Defined by ValueSymbolExtensions [▶ 1333].)

	Name	Description
	WriteValues(IObservable.Object., Action.Exception., CancellationToken) [▶ 1350]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775] . (Defined by ValueSymbolExtensions [▶ 1333] .)

Remarks

The DynamicSymbol adds dynamic run time behaviour to the [ISymbol \[▶ 2691\]](#)/[IValueSymbol \[▶ 2775\]](#). That means e.g. for StructSymbols that .NET Properties are defined and dispatched at runtime to the structs fields like they are defined in TwinCAT / ADS Types. Indexed access to Array Symbols is another example where the dynamic runtime support takes place.

Examples

Sample for the dynamic resolution of Symbols:

Dynamic Symbol access

```

/// <summary>
/// Defines the entry point of the application.
/// </summary>
/// <param name="args">The arguments.</param>
static async void Main(string[] args)
{
    // Get the AdsAddress from command-line arguments
    AmsAddress address = ArgParser.Parse(args);

    CancellationTokenSource cancelSource = new CancellationTokenSource();
    CancellationToken cancel = cancelSource.Token;

    using (AdsClient client = new AdsClient())
    {
        // Connect to the target device
        client.Connect(address);

        // Usage of "dynamic" Type and Symbols (>= .NET4 only)
        SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoaderMode.DynamicTree);
        IAdsSymbolLoader dynLoader = (IAdsSymbolLoader)SymbolLoaderFactory.Create(client, settings);
    }
}

```

Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

[TwinCAT.TypeSystem.IDynamicSymbol \[▶ 2496\]](#)

[TwinCAT.TypeSystem.IValueSymbol \[▶ 2775\]](#)

[TwinCAT.TypeSystem.ISymbol \[▶ 2691\]](#)

[System.Dynamic.DynamicObject](#)

Inheritance Hierarchy

System.Object

System.Dynamic.DynamicObject

TwinCAT.TypeSystem.DynamicSymbol

[TwinCAT.TypeSystem.DynamicAliasInstance \[▶ 2149\]](#)

[TwinCAT.TypeSystem.DynamicArrayInstance \[▶ 2160\]](#)

[TwinCAT.TypeSystem.DynamicInterfaceInstance \[▶ 2995\]](#)

[TwinCAT.TypeSystem.DynamicPointerInstance \[▶ 2186\]](#)

[TwinCAT.TypeSystem.DynamicReferenceInstance \[▶ 2202\]](#)







[TwinCAT.TypeSystem.DynamicUnionInstance \[▶ 2336\]](#)

6.11.21.1 DynamicSymbol Properties

The [DynamicSymbol](#) [▶ 2266] type exposes the following members.

Properties

	Name	Description
	InnerSymbol [▶ 2274]	Gets the inner symbol of this DynamicSymbol [▶ 2266]
	AccessRights [▶ 2274]	Gets the access rights.
	AllowGIOAccess [▶ 2275]	Indicates, that the aggregates symbols is an IProcessImageAddress (and most probably IAdsSymbol)
	Attributes [▶ 2275]	Gets the Symbol Attributes
	BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits.
	ByteSize [▶ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes
	Category [▶ 2277]	Gets the category.
	Comment [▶ 2277]	Gets the comment of the IInstance [▶ 2549]
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266]
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549].
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances , what means the Symbols with locations).
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.))
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.))
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned ($BitSize \% 8 == 0$)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is acontainer type.
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent.
	IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type.
	IsReadOnly [▶ 2283]	Indicates that this instance is read only.
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive.
	IsReference [▶ 2284]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static.
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters),

	Name	Description
	NotificationSettings [▶ 2285]	Gets the notification settings.
	Parent [▶ 2286]	Gets the parent Symbol
	Size [▶ 2286]	Gets the size of the Instance [▶ 2549] in bytes.
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691]
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549].
	ValueEncoding [▶ 2288]	Gets the value encoding.

Reference

[DynamicSymbol Class](#) [▶ 2266]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.21.1.1 DynamicSymbol._InnerSymbol Property

Gets the inner symbol of this [DynamicSymbol](#) [▶ 2266]

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IValueSymbol _InnerSymbol { get; }
```

Property Value

Type: [IValueSymbol](#) [▶ 2775]

The inner symbol.

Reference

[DynamicSymbol Class](#) [▶ 2266]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.21.1.2 DynamicSymbol.AccessRights Property

Gets the access rights.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolAccessRights AccessRights { get; }
```

Property Value

Type: [SymbolAccessRights](#) [▶ 2928]
The access rights.

Implements

[IValueSymbol.AccessRights](#) [▶ 2780]

Reference

[DynamicSymbol Class](#) [▶ 2266]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.21.1.3 DynamicSymbol.AllowIGIOAccess Property

Indicates, that the aggregates symbols is an IProcessImageAddress (and most probably IAdsSymbol)

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected bool AllowIGIOAccess { get; }
```

Property Value

Type: Boolean

Reference

[DynamicSymbol Class](#) [▶ 2266]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.21.1.4 DynamicSymbol.Attributes Property

Gets the Symbol Attributes

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ITypeAttributeCollection Attributes { get; }
```

Property Value

Type: [ITypeAttributeCollection](#) [▶ 2726]
The attributes.

Implements

[IAttributedInstance.Attributes](#) [▶ 2471]

Reference

[DynamicSymbol Class](#) [► 2266]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.21.1.5 DynamicSymbol.BitSize Property

Gets the size of the [IDataType](#) [► 2475] in bits.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int BitSize { get; }
```

Property Value

Type: Int32

The size of the bit.

Implements

[IBitSize.BitSize](#) [► 2473]

Reference

[DynamicSymbol Class](#) [► 2266]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.21.1.6 DynamicSymbol.ByteSize Property

Gets the (aligned) size of of the Type/Instance in Bytes

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ByteSize { get; }
```

Property Value

Type: Int32

The size of the byte.

Implements

[IBitSize.ByteSize](#) [► 2474]

Reference

[DynamicSymbol Class](#) [► 2266]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.21.1.7 DynamicSymbol.Category Property

Gets the category.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DataTypeCategory Category { get; }
```

Property Value

Type: [DataTypeCategory](#) [[▶ 2111](#)]

The category.

Implements

[ISymbol.Category](#) [[▶ 2694](#)]

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.1.8 DynamicSymbol.Comment Property

Gets the comment of the [IInstance](#) [[▶ 2549](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Comment { get; }
```

Property Value

Type: String

The comment.

Implements

[IInstance.Comment](#) [[▶ 2550](#)]

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.1.9 DynamicSymbol.Connection Property

Gets the connection bound to this [DynamicSymbol](#) [[▶ 2266](#)]

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IConnection? Connection { get; }
```

Property Value

Type: [IConnection](#) [▶ 79]

The connection.

Implements

[IValueSymbol.Connection](#) [▶ 2781]

Reference

[DynamicSymbol Class](#) [▶ 2266]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.21.1.10 DynamicSymbol.DataType Property

Gets the [IDataType](#) [▶ 2475] of the [IInstance](#) [▶ 2549].

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataType? DataType { get; }
```

Property Value

Type: [IDataType](#) [▶ 2475]

The type of the data.

Implements

[IInstance.DataType](#) [▶ 2551]

Reference

[DynamicSymbol Class](#) [▶ 2266]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.21.1.11 DynamicSymbol.HasValue Property

Gets a value indicating whether this instance has value (only the non VirtualInstances, what means the Symbols with locations).

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool HasValue { get; }
```

Property Value

Type: Boolean
true if this instance has value; otherwise, false.

Implements

[IValueRawSymbol.HasValue](#) [[▶ 2768](#)]

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.1.12 DynamicSymbol.InstanceName Property

Gets the name of the instance (without periods (.))

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string InstanceName { get; }
```

Property Value

Type: String
The name of the instance.

Implements

[IInstance.InstanceName](#) [[▶ 2551](#)]

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.1.13 DynamicSymbol.InstancePath Property

Gets the relative / absolute access path to the instance (with periods (.))

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string InstancePath { get; }
```

Property Value

Type: String
The instance path.

Implements

[IInstance.InstancePath](#) [[▶ 2552](#)]

Remarks

If this path is relative or absolute depends on the context. [IMember](#) [[▶ 2561](#)] are using relative paths, [ISymbol](#) [[▶ 2691](#)]s are using absolute ones.

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.1.14 DynamicSymbol.IsBitType Property

Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsBitType { get; }
```

Property Value

Type: Boolean
true if this instance is bit mapping; otherwise, false.

Implements

[IBitSize.IsBitType](#) [[▶ 2474](#)]

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.1.15 DynamicSymbol.IsByteAligned Property

Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsByteAligned { get; }
```

Property Value

Type: Boolean
true if this instance is byte aligned; otherwise, false.

Implements

[IBitSize.IsByteAligned](#) [[▶ 2475](#)]

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.1.16 DynamicSymbol.IsContainerType Property

Gets a value indicating whether this Symbol is acontainer type.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsContainerType { get; }
```

Property Value

Type: Boolean
true if this instance is container type; otherwise, false.

Implements

[ISymbol.IsContainerType](#) [[▶ 2694](#)]

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.1.17 DynamicSymbol.IsPersistent Property

Gets a value indicating whether this [ISymbol](#) [[▶ 2691](#)] is persistent.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsPersistent { get; }
```

Property Value

Type: Boolean
true if this instance is persistent; otherwise, false.

Implements

[ISymbol.IsPersistent](#) [► 2694]

Reference

[DynamicSymbol Class](#) [► 2266]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.21.1.18 DynamicSymbol.IsPointer Property

Indicates that the [Instance](#) [► 2549] represents a Pointer type (Pointer TO)

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsPointer { get; }
```

Property Value

Type: Boolean
true if is ReferenceTo, otherwise false.

Implements

[Instance.IsPointer](#) [► 2552]

Reference

[DynamicSymbol Class](#) [► 2266]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.21.1.19 DynamicSymbol.IsPrimitiveType Property

Gets a value indicating whether this instance is a primitive type.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsPrimitiveType { get; }
```

Property Value

Type: Boolean
true if this instance is primitive type; otherwise, false.

Implements

[ISymbol.IsPrimitiveType](#) [[▶ 2695](#)]

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.1.20 DynamicSymbol.IsReadOnly Property

Indicates that this instance is read only.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsReadOnly { get; }
```

Property Value

Type: Boolean

Implements

[ISymbol.IsReadOnly](#) [[▶ 2695](#)]

Remarks

Actually, this Flag is restricted to TcCOM-Objects readonly Parameters. Within the PLC this is used for the ApplicationName and ProjectName of PLC instances. Write-Access on these Modules will create an [DeviceAccessDenied](#) [[▶ 664](#)] error.

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.1.21 DynamicSymbol.IsRecursive Property

Gets a value indicating whether this instance is recursive.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsRecursive { get; }
```

Property Value

Type: Boolean

true if this instance is recursive; otherwise, false.

Implements

[ISymbol.IsRecursive](#) [[▶ 2696](#)]

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.1.22 DynamicSymbol.IsReference Property

Indicates that the [IInstance](#) [[▶ 2549](#)] represents a Reference type (REFERENCE TO)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsReference { get; }
```

Property Value

Type: Boolean

true if is ReferenceTo, otherwise false.

Implements

[IInstance.IsReference](#) [[▶ 2552](#)]

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.1.23 DynamicSymbol.IsStatic Property

Gets a value indicating whether this instance is static.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsStatic { get; }
```

Property Value

Type: Boolean

true if this instance is static; otherwise, false.

Implements

[IInstance.IsStatic](#) [[▶ 2553](#)]

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.1.24 DynamicSymbol.NormalizedName Property

Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters),

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string NormalizedName { get; }
```

Property Value

Type: String

The normalized instance name (can be the same like [InstanceName](#) [[▶ 2551](#)])

Implements

[IDynamicSymbol.NormalizedName](#) [[▶ 2499](#)]

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

[IInstance.InstanceName](#) [[▶ 2551](#)]

6.11.21.1.25 DynamicSymbol.NotificationSettings Property

Gets the notification settings.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public INotificationSettings? NotificationSettings { get; set; }
```

Property Value

Type: [INotificationSettings](#) [[▶ 1098](#)]

The notification settings.

Implements

[IValueSymbol.NotificationSettings](#) [[▶ 2781](#)]

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.1.26 DynamicSymbol.Parent Property

Gets the parent Symbol

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISymbol? Parent { get; }
```

Property Value

Type: [ISymbol](#) [[▶ 2691](#)]

The parent.

Implements

[ISymbol.Parent](#) [[▶ 2696](#)]

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.1.27 DynamicSymbol.Size Property

Gets the size of the [Instance](#) [[▶ 2549](#)] in bytes.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Size { get; }
```

Property Value

Type: Int32

The size of the [Instance](#) [[▶ 2549](#)] in bytes.

Implements

[IBitSize.Size](#) [[▶ 2475](#)]

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.1.28 DynamicSymbol.SubSymbols Property

Gets the SubSymbols of the [ISymbol \[► 2691\]](#)

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISymbolCollection<ISymbol> SubSymbols { get; }
```

Property Value

Type: [ISymbolCollection \[► 2700\]](#).[ISymbol \[► 2691\]](#).

Implements

[ISymbol.SubSymbols \[► 2696\]](#)

Remarks

Only used for Array and Struct instances. Otherwise empty

Reference

[DynamicSymbol Class \[► 2266\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.1.29 DynamicSymbol.TypeName Property

Gets the name of the [DataType \[► 2475\]](#) that is used for this [IInstance \[► 2549\]](#).

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string TypeName { get; }
```

Property Value

Type: String
The name of the type.

Implements

[IInstance.TypeName \[► 2553\]](#)

Reference

[DynamicSymbol Class \[► 2266\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.1.30 DynamicSymbol.ValueEncoding Property

Gets the value encoding.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Encoding ValueEncoding { get; }
```

Property Value

Type: Encoding

The value encoding.

Implements

[IAttributedInstance.ValueEncoding \[► 2472\]](#)

Reference









[DynamicSymbol Class \[► 2266\]](#)






















[TwinCAT.TypeSystem Namespace \[► 2083\]](#)












6.11.21.2 DynamicSymbol Methods






The [DynamicSymbol \[► 2266\]](#) type exposes the following members.

Methods

	Name	Description
	Equals [► 2292]	Equals (Overrides Object.Equals(Object).)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetDynamicMemberNames [► 2293]	Returns the enumeration of all dynamic member names. (Overrides DynamicObject.GetDynamicMemberNames..)
	GetHashCode [► 2293]	Gets the GetHashCode of the Address (Overrides Object.GetHashCode..)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	OnReadAnyValue [► 2293]	Handler function for reading ADS 'Any' Values.






	Name	Description
	OnReadRawValue [▶ 2294]	Handler function for reading Raw symbol value.
	OnReadRawValueAsync [▶ 2294]	Handler function reading the raw value of the DynamicSymbol [▶ 2266].
	OnReadValue [▶ 2295]	Handler function for the
	OnReadValueAsync [▶ 2296]	Handler function reading the DynamicSymbols [▶ 2266] value asynchronously.
	OnSetInstanceName [▶ 2296]	Sets a new InstanceName InstancePath
	OnTryReadValue [▶ 2297]	Handler function for the
	OnTryWriteValue [▶ 2297]	Handler Function for writing value.
	OnWriteRawValue [▶ 2298]	Handler function for reading symbols raw value.
	OnWriteRawValueAsync [▶ 2298]	Handler function for writing the raw DynamicSymbol [▶ 2266] value.
	OnWriteValue [▶ 2299]	Handler Function for writing value.
	OnWriteValueAsync [▶ 2300]	Handler Function for writing value.
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type
	ReadRawValue. [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value
	ReadRawValueAsync [▶ 2302]	Read raw value as an asynchronous operation.
	ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266].
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266].
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously.
	ToString [▶ 2307]	Returns a String that represents this instance. (Overrides Object.ToString..)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject.)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject.)

	Name	Description
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetIndex	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicObject.)
	TryGetMember [▶ 2308]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Overrides DynamicObject.TryGetMember(GetMemberBinder, Object.).)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject.)
	TryInvokeMember	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicObject.)
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775]
	TrySetIndex	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicObject.)
	TrySetMember	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicObject.)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
 	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266] .
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value.
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775]
	WriteRawValue(.Byte) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	WriteRawValue(.Byte, e., Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)

	Name	Description
	WriteRawValueAsyn c [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266] .
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266] .
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775]

Extension Methods

	Name	Description
	PollValues(IObserva ble.Unit.) [▶ 1355]	Overloaded. Poll symbol values on trigger signals. (Defined by ValueSymbolExtensions [▶ 1333] .)
 	PollValues(TimeSpa n) [▶ 1356]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by ValueSymbolExtensions [▶ 1333] .)
	PollValues(IObserva ble.Unit., Boolean) [▶ 1357]	Overloaded. Polls symbol values on trigger signals. (Defined by ValueSymbolExtensions [▶ 1333] .)
 	PollValues(TimeSpa n, Boolean) [▶ 1358]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by ValueSymbolExtensions [▶ 1333] .)
	PollValues.T. (IObservable.Unit.) [▶ 1362]	Overloaded. Poll symbol values as a value sequence on trigger signals (typed) (Defined by ValueSymbolExtensions [▶ 1333] .)
 	PollValues.T. (TimeSpan) [▶ 1363]	Overloaded. Polls the symbol as value sequence of object values with a specified period time (typed) (Defined by ValueSymbolExtensions [▶ 1333] .)
 	PollValues.T. (IObservable.Unit., Func.ResultReadVal ueAccess2.IValueSy mbol, Object., T.) [▶ 1365]	Overloaded. Poll symbol values on trigger signals (typed) (Defined by ValueSymbolExtensions [▶ 1333] .)
	PollValues.T. (IObservable.Unit., Boolean) [▶ 1364]	Overloaded. Poll symbol values on trigger signals (typed) (Defined by ValueSymbolExtensions [▶ 1333] .)
	PollValues.T. (TimeSpan, Boolean) [▶ 1367]	Overloaded. Polls the symbol as value sequence of object values with a specified period time (typed) (Defined by ValueSymbolExtensions [▶ 1333] .)
	PollValues2(IObserv able.Unit.) [▶ 1368]	Overloaded. Poll symbol values as a sequence of annotated results (Value + ErrorCode) (Defined by ValueSymbolExtensions [▶ 1333] .)

	Name	Description
	PollValues2(TimeSpan) [▶ 1369]	Overloaded. Poll symbol values with communication return codes. (Defined by ValueSymbolExtensions [▶ 1333].)
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
 	WhenValueChanged [▶ 1344]	Gets an observable sequence when the value of the IValueSymbol [▶ 2775] has changed. (Defined by ValueSymbolExtensions [▶ 1333].)
 	WriteValues(IObservableObject) [▶ 1347]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775]. (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues(IObservableObject, Action.Exception) [▶ 1348]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775]. (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues(IObservableObject, CancellationToken) [▶ 1349]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775]. (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues(IObservableObject, Action.Exception, CancellationToken) [▶ 1350]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775]. (Defined by ValueSymbolExtensions [▶ 1333].)

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.2.1 DynamicSymbol.Equals Method

Equals

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool Equals(
    Object? obj
)
```

Parameters

obj	Type: System.Object The object to compare with the current object.
-----	---

Return Value

Type: Boolean

true if the specified Object is equal to this instance; otherwise, false.

Reference

[DynamicSymbol Class](#) [► 2266]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.21.2.2 DynamicSymbol.GetDynamicMemberNames Method

Returns the enumeration of all dynamic member names.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override IEnumerable<string> GetDynamicMemberNames ()
```

Return Value

Type: IEnumerable.String.

A sequence that contains dynamic member names.

Reference

[DynamicSymbol Class](#) [► 2266]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.21.2.3 DynamicSymbol.GetHashCode Method

Gets the HashCode of the Address

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override int GetHashCode ()
```

Return Value

Type: Int32

A hash code for this instance, suitable for use in hashing algorithms and data structures like a hash table.

Reference

[DynamicSymbol Class](#) [► 2266]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.21.2.4 DynamicSymbol.OnReadAnyValue Method

Handler function for reading ADS 'Any' Values.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Object OnReadAnyValue(
    Type managedType
)
```

Parameters

managedType	Type: System.Type Managed type to read.
-------------	--

Return Value

Type: Object
System.Object.

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.2.5 DynamicSymbol.OnReadRawValue Method

Handler function for reading Raw symbol value.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual byte[] OnReadRawValue(
    int timeout
)
```

Parameters

timeout	Type: System.Int32 The timeout.
---------	------------------------------------

Return Value

Type: .Byte.
System.Byte[].

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.2.6 DynamicSymbol.OnReadRawValueAsync Method

Handler function reading the raw value of the [DynamicSymbol](#) [[▶ 2266](#)].

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<ResultReadRawAccess> OnReadRawValueAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultReadRawAccess \[▸ 3206\]](#).
 Task<ResultReadRawAccess>.

Reference

- [DynamicSymbol Class \[▸ 2266\]](#)
- [TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.21.2.7 DynamicSymbol.OnReadValue Method

Handler function for the

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Object OnReadValue(
    int timeout
)
```

Parameters

timeout	Type: System.Int32 The timeout.
---------	------------------------------------

Return Value

Type: Object
 System.Object.

Exceptions

Exception	Condition
CannotAccessVirtualSymbolException [▸ 2103]	

Reference

- [DynamicSymbol Class \[▸ 2266\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.2.8 DynamicSymbol.OnReadValueAsync Method

Handler function reading the [DynamicSymbols \[► 2266\]](#) value asynchronously.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<ResultReadValueAccess> OnReadValueAsync (
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultReadValueAccess \[► 3209\]](#).

An asynchronous task returning the [ResultReadValueAccess \[► 3209\]](#) as result.

Exceptions

Exception	Condition
CannotAccessVirtualSymbolException [► 2103]	

Reference

[DynamicSymbol Class \[► 2266\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.2.9 DynamicSymbol.OnSetInstanceName Method

Sets a new InstanceName InstancePath

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual void OnSetInstanceName (
    string instanceName
)
```

Parameters

instanceName	Type: System.String Instance name.
--------------	---------------------------------------

Reference

[DynamicSymbol Class \[► 2266\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.2.10 DynamicSymbol.OnTryReadValue Method

Handler function for the

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual int OnTryReadValue(
    int timeout,
    out Object?? value
)
```

Parameters

timeout	Type: System.Int32 The timeout.
value	Type: System.Object. The value.

Return Value

Type: Int32
The error Code.

Exceptions

Exception	Condition
CannotAccessVirtualSymbolException [► 2103]	

Reference

[DynamicSymbol Class \[► 2266\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.2.11 DynamicSymbol.OnTryWriteValue Method

Handler Function for writing value.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual int OnTryWriteValue(
    Object value,
    int timeout
)
```

Parameters

value	Type: System.Object The value.
timeout	Type: System.Int32 The timeout.

Return Value

Type: Int32

Exceptions

Exception	Condition
CannotAccessVirtualSymbolException [▶ 2103]	

Reference[DynamicSymbol Class](#) [▶ 2266][TwinCAT.TypeSystem Namespace](#) [▶ 2083]**6.11.21.2.12 DynamicSymbol.OnWriteRawValue Method**

Handler function for reading symbols raw value.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
protected virtual void OnWriteRawValue(
    byte[] rawValue,
    int timeout
)
```

Parameters

rawValue	Type: System.Byte. The value as byte array.
timeout	Type: System.Int32 The timeout.

Reference[DynamicSymbol Class](#) [▶ 2266][TwinCAT.TypeSystem Namespace](#) [▶ 2083]**6.11.21.2.13 DynamicSymbol.OnWriteRawValueAsync Method**Handler function for writing the raw [DynamicSymbol](#) [▶ 2266] value.**Namespace:** [TwinCAT.TypeSystem](#) [▶ 2083]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<ResultWriteAccess> OnWriteRawValueAsync (
    byte[] rawValue,
    CancellationToken cancel
)
```

Parameters

rawValue	Type: .System.Byte. The raw value to write.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWriteAccess](#) [▶ 3217].
An asynchronous task object that represents the 'OnWriteRawValue' operation and returns a [ResultWriteAccess](#) [▶ 3217] as result.

Reference

- [DynamicSymbol Class](#) [▶ 2266]
- [TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.21.2.14 DynamicSymbol.OnWriteValue Method

Handler Function for writing value.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual void OnWriteValue (
    Object value,
    int timeout
)
```

Parameters

value	Type: System.Object The value.
timeout	Type: System.Int32 The timeout.

Exceptions

Exception	Condition
CannotAccessVirtualSymbolException [▶ 2103]	

Reference

- [DynamicSymbol Class](#) [▶ 2266]

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.2.15 DynamicSymbol.OnWriteValueAsync Method

Handler Function for writing value.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected virtual Task<ResultWriteAccess> OnWriteValueAsync (
    Object value,
    CancellationToken cancel
)
```

Parameters

value	Type: System.Object The value.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWriteAccess \[► 3217\]](#).

Task<WriteValueResult>.

Exceptions

Exception	Condition
CannotAccessVirtualSymbolException [► 2103]	

Reference

[DynamicSymbol Class \[► 2266\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.2.16 DynamicSymbol.ReadAnyValue Method

Reads the value of this [Value \[► 2775\]](#) into a new created instance of the managed type

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object ReadAnyValue (
    Type managedType
)
```

Parameters

managedType	Type: System.Type The tp.
-------------	------------------------------

Return Value



Type: Object
Read value (System.Object).

Reference

- [DynamicSymbol Class \[▶ 2266\]](#)
- [TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)
- [IValueAnySymbol.WriteAnyValue\(Object\) \[▶ 2764\]](#)
- [IValueAnySymbol.UpdateAnyValue\(Object.\) \[▶ 2762\]](#)

6.11.21.2.17 DynamicSymbol.ReadRawValue Method

Overload List

	Name	Description
	ReadRawValue. [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value

Reference

- [DynamicSymbol Class \[▶ 2266\]](#)
- [TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.21.2.17.1 DynamicSymbol.ReadRawValue Method

Reads the raw value of the [IValueSymbol \[▶ 2775\]](#) (Ads Read / Write)

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public byte[] ReadRawValue()
```

Field Value

Type: .Byte.
The raw value.

Return Value

Type: .Byte.
System.Byte[].

Implements

[IValueRawSymbol.ReadRawValue. \[▶ 2770\]](#)

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[ReadRawValue Overload](#) [[▶ 2301](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.2.17.2 DynamicSymbol.ReadRawValue Method (Int32)

Reads the Symbols raw value

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public byte[] ReadRawValue(  
    int timeout  
)
```

Parameters

timeout	Type: System.Int32 The timeout in ms.
---------	--

Field Value

Type: .Byte.

The raw value in bytes.

Return Value

Type: .Byte.

System.Byte[].

Implements

[IValueRawSymbol.ReadRawValue\(Int32\)](#) [[▶ 2771](#)]

Remarks

A negative timeout indicates that the Default Timeout for the communication will be used. 0 means timeout is switched off.

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[ReadRawValue Overload](#) [[▶ 2301](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.2.18 DynamicSymbol.ReadRawValueAsync Method

Read raw value as an asynchronous operation.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultReadRawAccess> ReadRawValueAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Field Value

Type: Task.ResultReadRawAccess [[▶ 3206](#)].
The raw value.

Return Value

Type: Task.ResultReadRawAccess [[▶ 3206](#)].
System.Byte[].

Implements

[IValueRawSymbol.ReadRawValueAsync\(CancellationToken\)](#) [[▶ 2772](#)]



Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.2.19 DynamicSymbol.ReadValue Method

Overload List

	Name	Description
	ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266].
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266].

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.2.19.1 DynamicSymbol.ReadValue Method

Reads the value of this [DynamicSymbol](#) [[▶ 2266](#)].

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object ReadValue()
```

Return Value

Type: Object
System.Object.

Implements

[IValueSymbol.ReadValue. \[▸ 2784\]](#)

Remarks**Examples****Dynamic Read access**

```
namespace Sample
{
    using System;
    using System.Diagnostics;
    using System.Threading;
    using TwinCAT;
    using TwinCAT.Ads;
    using TwinCAT.Ads.TypeSystem;
    using TwinCAT.Ads.ValueAccess;
    using TwinCAT.TypeSystem;
    using TwinCAT.TypeSystem.Generic;
    using TwinCAT.ValueAccess;

    class SymbolBrowserProgramV2DynamicTree
    {
        #region CODE_SAMPLE_SIMPLEDYNAMIC
        /// <summary>
        /// Defines the entry point of the application.
        /// </summary>
        /// <param name="args">The arguments.</param>
        static async void Main(string[] args)
        {
            // Get the AdsAddress from command-line arguments
            AmsAddress address = ArgParser.Parse(args);

            CancellationTokenSource cancelSource = new CancellationTokenSource();
            CancellationToken cancel = cancelSource.Token;

            using (AdsClient client = new AdsClient())
            {
                // Connect to the target device
                client.Connect(address);

                // Usage of "dynamic" Type and Symbols (>= .NET4 only)
                SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.DynamicTree);
                IAdsSymbolLoader dynLoader = (IAdsSymbolLoader)SymbolLoaderFactory.Create(client, settings);

            #endregion

            // Set the Default setting for Notifications
            dynLoader.DefaultNotificationSettings = new NotificationSettings(AdsTransMode.OnChange, 200,
2000);

            // Get the Symbols (Dynamic Symbols)
            var resultSymbols = await ((IDynamicSymbolLoader)dynLoader).GetDynamicSymbolsAsync(cancel);

            dynamic dynamicSymbols = resultSymbols.Symbols;
            dynamic adsPort = dynamicSymbols.TwinCAT_SystemInfoVarList._AppInfo.AdsPort;

            #region CODE_SAMPLE_SIMPLEDYNAMIC

            // Access Main Symbol with Dynamic Language Runtime support (DLR)
            // Dynamically created property "Main"
            //dynamic symMain = dynamicSymbols.Main;

            // Main is an 'VirtualSymbol' / Organizational unit that doesn't have a value
            // Calling ReadValue is not allowed
            //bool test = symMain.HasValue;
            //dynamic invalid = symMain.ReadValue();
        }
    }
}
```



```

        //Reading TaskInfo Value
        //
With calling ReadValueAsync() a 'snapshot' of the Symbols Instance is taken (reading async)
        ResultReadValueAccess resultRead = await dynamicSymbols.TwinCAT_SystemInfoVarList._TaskInfo.
ReadValueAsync(cancel);
        dynamic vTaskInfoArray = resultRead.Value;

        // Getting the Snapshot time in UTC format
        DateTimeOffset timeStamp1 = vTaskInfoArray.TimeStamp;

        // Getting TaskInfo Symbol for Task 1
        dynamic symTaskInfo1 = dynamicSymbols.TwinCAT_SystemInfoVarList._TaskInfo[1];

        // Getting CycleCount Symbol
        dynamic symCycleCount = symTaskInfo1.CycleCount;

        // Take Snapshot value of the ApplicationInfo struct
        resultRead = await dynamicSymbols.TwinCAT_SystemInfoVarList._AppInfo.ReadValueAsync(cancel);
        dynamic vAppInfo = resultRead.Value;

        // Get the UTC Timestamp of the snapshot
        DateTimeOffset timeStamp2 = vAppInfo.TimeStamp;

        // Access the ProjectName of the ApplicationInfo Snapshot (type-safe!)
        string projectNameValue = vAppInfo.ProjectName;

        // Reading the CycleCount Value
        resultRead = await symTaskInfo1.CycleCount.ReadValueAsync(cancel);    // Taking a Value Sna
psnot
        int cycleCountValue = (int)resultRead.Value;
        #endregion

        // Registering for dynamic "ValueChanged" events for the Values
        // Using Default Notification settings
        symCycleCount.ValueChanged += new EventHandler<ValueChangedEventArgs>(cycleCount_ValueChange
d);

        // Override default notification settings
        symTaskInfo1.NotificationSettings = new NotificationSettings(AdsTransMode.Cyclic, 500, 0);

        // Register for ValueChanged event.
        symTaskInfo1.ValueChanged += new EventHandler<ValueChangedEventArgs>(taskInfo1Value_ValueCha
nged); // Struct Type

        Thread.Sleep(10000); // Sleep main thread for 10 Seconds
    }
    Console.WriteLine("CycleCount Changed events received: {0}", _cycleCountEvents);
    Console.WriteLine("taskInfo1 Changed events received: {0}", _taskInfo1Events);

    Console.WriteLine("");
    Console.WriteLine("Press [Enter] for leave:");
    Console.ReadLine();
}

static object _notificationSynchronizer = new object();
static int _cycleCountEvents = 0;

/// <summary>
/// Handler function for the CycleCount ValueChanged event.
/// </summary>
/// <param name="sender">The sender.</param>
/// <param name="e">The e.</param>
static void cycleCount_ValueChanged(object sender, ValueChangedEventArgs e)
{
    lock(_notificationSynchronizer)
    {
        Interlocked.Increment(ref _cycleCountEvents);
        // val is a type safe value of int!
        dynamic val = e.Value;
        uint intVal = val;

        DateTimeOffset changedTime = e.DateTime.ToLocalTime(); // Convert UTC to local time
        Console.WriteLine("CycleCount changed to: {0}, TimeStamp: {1}", intVal, changedTime.ToString
("HH:mm:ss:fff"));
    }
}

static int _taskInfo1Events = 0;

```

```

/// <summary>
/// Handler function for the TaskInfo ValueChanged event.
/// </summary>
/// <param name="sender">The sender.</param>
/// <param name="e">The e.</param>
static void taskInfo1Value_ValueChanged(object sender, ValueChangedEventArgs e)
{
    lock (_notificationSynchronizer)
    {
        Interlocked.Increment(ref _taskInfo1Events);
        dynamic val = e.Value;
        DateTimeOffset changedTime = e.DateTime.ToLocalTime(); // Convert to local time

        // Val is a during Runtime created struct type and contains
        // the same Properties as related PLC object.
        int cycleTime = val.CycleTime;
        Console.WriteLine("TaskInfo1Value changed TimeStamp: {0}", changedTime.ToString("HH:mm:ss:fff"));
    }
}
}
}
}

```

Reference

[DynamicSymbol Class \[► 2266\]](#)

[ReadValue Overload \[► 2303\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.2.19.2 DynamicSymbol.ReadValue Method (Int32)

Reads the value of this [DynamicSymbol \[► 2266\]](#).

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object ReadValue(
    int timeout
)
```

Parameters

timeout	Type: System.Int32 The timeout in ms.
---------	--

Field Value

Type: Object
The value.

Return Value

Type: Object
System.Object.

Implements

[IValueSymbol.ReadValue\(Int32\) \[► 2785\]](#)

Remarks

Calling on primitive types, a call of this method will return the primitive value. On complex types (structures and arrays) it depends on the [ISymbolLoader \[▸ 2714\]](#) settings what will happen. In non dynamic modes: the raw byte Array will be returned, in dynamic mode: A Value will be created on the fly. A negative timeout indicates that the Default Timeout for the communication will be used. 0 means timeout is switched off.

Reference

[DynamicSymbol Class \[▸ 2266\]](#)

[ReadValue Overload \[▸ 2303\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.21.2.20 DynamicSymbol.ReadValueAsync Method

Reads the Value of the [IValueSymbol \[▸ 2775\]](#) asynchronously.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultReadValueAccess> ReadValueAsync (
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: Task.[ResultReadValueAccess \[▸ 3209\]](#).

A tasks that represents the asynchronous 'ReadValue' operation. The read result is stored in the [ResultReadValueAccess \[▸ 3209\]](#) return value and contains the [Value \[▸ 3213\]](#) and the [ErrorCode \[▸ 3202\]](#).

Implements

[IValueSymbol.ReadValueAsync\(CancellationToken\) \[▸ 2785\]](#)

Remarks

Calling on primitive types, a call of this method will return the primitive value. On complex types (structures and arrays) it depends on the [ISymbolLoader \[▸ 2714\]](#) settings what will happen. In non dynamic modes: the raw byte Array will be returned, in dynamic mode: A Value will be created on the fly.

Reference

[DynamicSymbol Class \[▸ 2266\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.21.2.21 DynamicSymbol.ToString Method

Returns a String that represents this instance.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```

Return Value

Type: String

A String that represents this instance.

Reference

[DynamicSymbol Class \[► 2266\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.2.22 DynamicSymbol.TryGetMember Method

Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TryGetMember(
    GetMemberBinder binder,
    out Object?? result
)
```

Parameters

binder	Type: System.Dynamic.GetMemberBinder Provides information about the object that called the dynamic operation. The binder.Name property provides the name of the member on which the dynamic operation is performed. For example, for the Console.WriteLine(sampleObject.SampleProperty) statement, where sampleObject is an instance of the class derived from the DynamicObject class, binder.Name returns "SampleProperty". The binder.IgnoreCase property specifies whether the member name is case-sensitive.
result	Type: System.Object. The result of the get operation. For example, if the method is called for a property, you can assign the property value to result.

Return Value

Type: Boolean

true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a run-time exception is thrown.)

Reference

[DynamicSymbol Class \[► 2266\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.2.23 DynamicSymbol.TryReadValue Method

Reads the Value of the [IValueSymbol](#) [[▶ 2775](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int TryReadValue(  
    int timeout,  
    out Object?? value  
)
```

Parameters

timeout	Type: System.Int32 The timeout in ms.
value	Type: System.Object. The symbol value.

Return Value

Type: Int32

The error code.

Implements

[IValueSymbol.TryReadValue\(Int32, Object.\)](#) [[▶ 2786](#)]

Remarks

Calling on primitive types, a call of this method will return the primitive value. On complex types (structures and arrays) it depends on the [ISymbolLoader](#) [[▶ 2714](#)] settings what will happen. In non dynamic modes: the raw byte Array will be returned, in dynamic mode: A Value will be created on the fly. A negative timeout indicates that the Default Timeout for the communication will be used. 0 means timeout is switched off.

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.2.24 DynamicSymbol.TryWriteValue Method

Writes the specified value to the [DynamicSymbol](#) [[▶ 2266](#)].

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int TryWriteValue(  
    Object value,  
    int timeout  
)
```

Parameters

value	Type: System.Object The value.
timeout	Type: System.Int32 The timeout in ms.

Return Value

Type: Int32
The error code.

Implements

[IValueSymbol.TryWriteValue\(Object, Int32\)](#) [► 2787]

Remarks

Calling on primitive types, a call of this method will directly write this Value. On complex types (structs and arrays) it depends on the [ISymbolLoader](#) [► 2714] settings what will happen. In non dynamic modes: Only byte Arrays (of correct size) can be written) in dynamic mode: A Value that represents the value will be accepted also. A negative timeout indicates that the Default Timeout for the communication will be used. 0 means timeout is switched off.

Examples**Dynamic Write access**

```
namespace Sample
{
    using System;
    using System.Diagnostics;
    using System.Threading;
    using TwinCAT;
    using TwinCAT.Ads;
    using TwinCAT.Ads.TypeSystem;
    using TwinCAT.Ads.ValueAccess;
    using TwinCAT.TypeSystem;
    using TwinCAT.TypeSystem.Generic;
    using TwinCAT.ValueAccess;

    class SymbolBrowserProgramV2DynamicTree
    {
        #region CODE_SAMPLE_SIMPLEDYNAMIC
        /// <summary>
        /// Defines the entry point of the application.
        /// </summary>
        /// <param name="args">The arguments.</param>
        static async void Main(string[] args)
        {
            // Get the AdsAddress from command-line arguments
            AmsAddress address = ArgParser.Parse(args);

            CancellationTokenSource cancelSource = new CancellationTokenSource();
            CancellationToken cancel = cancelSource.Token;

            using (AdsClient client = new AdsClient())
            {
                // Connect to the target device
                client.Connect(address);

                // Usage of "dynamic" Type and Symbols (>= .NET4 only)
                SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.DynamicTree);
                IAdsSymbolLoader dynLoader = (IAdsSymbolLoader)SymbolLoaderFactory.Create(client, settings);

            }

            #endregion

            // Set the Default setting for Notifications
            dynLoader.DefaultNotificationSettings = new NotificationSettings(AdsTransMode.OnChange, 200,
                2000);
        }
    }
}
```

```

// Get the Symbols (Dynamic Symbols)
var resultSymbols = await ((IDynamicSymbolLoader) dynLoader).GetDynamicSymbolsAsync(cancel);

dynamic dynamicSymbols = resultSymbols.Symbols;
dynamic adsPort = dynamicSymbols.TwinCAT_SystemInfoVarList._AppInfo.AdsPort;

#region CODE_SAMPLE_SIMPLEDYNAMIC

// Access Main Symbol with Dynamic Language Runtime support (DLR)
// Dynamically created property "Main"
//dynamic symMain = dynamicSymbols.Main;

// Main is an 'VirtualSymbol' / Organizational unit that doesn't have a value
// Calling ReadValue is not allowed
//bool test = symMain.HasValue;
//dynamic invalid = symMain.ReadValue();

//Reading TaskInfo Value
//
With calling ReadValueAsync() a 'snapshot' of the Symbols Instance is taken (reading async)
ResultReadValueAccess resultRead = await dynamicSymbols.TwinCAT_SystemInfoVarList._TaskInfo.
ReadValueAsync(cancel);
dynamic vTaskInfoArray = resultRead.Value;

// Getting the Snapshot time in UTC format
DateTimeOffset timeStamp1 = vTaskInfoArray.TimeStamp;

// Getting TaskInfo Symbol for Task 1
dynamic symTaskInfo1 = dynamicSymbols.TwinCAT_SystemInfoVarList._TaskInfo[1];

// Getting CycleCount Symbol
dynamic symCycleCount = symTaskInfo1.CycleCount;

// Take Snapshot value of the ApplicationInfo struct
resultRead = await dynamicSymbols.TwinCAT_SystemInfoVarList._AppInfo.ReadValueAsync(cancel);
dynamic vAppInfo = resultRead.Value;

// Get the UTC Timestamp of the snapshot
DateTimeOffset timeStamp2 = vAppInfo.TimeStamp;

// Access the ProjectName of the ApplicationInfo Snapshot (type-safe!)
string projectNameValue = vAppInfo.ProjectName;

// Reading the CycleCount Value
resultRead = await symTaskInfo1.CycleCount.ReadValueAsync(cancel); // Taking a Value Snapshot

int cycleCountValue = (int)resultRead.Value;
#endregion

// Registering for dynamic "ValueChanged" events for the Values
// Using Default Notification settings
symCycleCount.ValueChanged += new EventHandler<ValueChangedEventArgs>(cycleCount_ValueChanged);

// Override default notification settings
symTaskInfo1.NotificationSettings = new NotificationSettings(AdsTransMode.Cyclic, 500, 0);

// Register for ValueChanged event.
symTaskInfo1.ValueChanged += new EventHandler<ValueChangedEventArgs>(taskInfo1Value_ValueChanged); // Struct Type

Thread.Sleep(10000); // Sleep main thread for 10 Seconds
}
Console.WriteLine("CycleCount Changed events received: {0}", _cycleCountEvents);
Console.WriteLine("taskInfo1 Changed events received: {0}", _taskInfo1Events);

Console.WriteLine("");
Console.WriteLine("Press [Enter] for leave:");
Console.ReadLine();
}

static object _notificationSynchronizer = new object();
static int _cycleCountEvents = 0;

/// <summary>
/// Handler function for the CycleCount ValueChanged event.
/// </summary>
/// <param name="sender">The sender.</param>

```

```
/// <param name="e">The e.</param>
static void cycleCount_ValueChanged(object sender, ValueChangedEventArgs e)
{
    lock(_notificationSynchronizer)
    {
        Interlocked.Increment(ref _cycleCountEvents);
        // val is a type safe value of int!
        dynamic val = e.Value;
        uint intVal = val;

        DateTimeOffset changedTime = e.DateTime.ToLocalTime(); // Convert UTC to local time
        Console.WriteLine("CycleCount changed to: {0}, TimeStamp: {1}", intVal, changedTime.ToString(
"HH:mm:ss:fff"));
    }
}

static int _taskInfoEvents = 0;

/// <summary>
/// Handler function for the TaskInfo ValueChanged event.
/// </summary>
/// <param name="sender">The sender.</param>
/// <param name="e">The e.</param>
static void taskInfoValue_ValueChanged(object sender, ValueChangedEventArgs e)
{
    lock (_notificationSynchronizer)
    {
        Interlocked.Increment(ref _taskInfoEvents);
        dynamic val = e.Value;
        DateTimeOffset changedTime = e.DateTime.ToLocalTime(); // Convert to local time

        // Val is a during Runtime created struct type and contains
        // the same Properties as related PLC object.
        int cycleTime = val.CycleTime;
        Console.WriteLine("TaskInfoValue changed TimeStamp: {0}", changedTime.ToString("HH:mm:ss:ff
f"));
    }
}
}
```

Reference

[DynamicSymbol Class](#) [► 2266]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.21.2.25 DynamicSymbol.UpdateAnyValue Method

Reads the value of this [Value](#) [► 2775] into the specified managed value.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void UpdateAnyValue(
    ref Object valueObject
)
```

Parameters

valueObject	Type: System.Object. The managed object.
-------------	---

Return Value

Type:
Read value (System.Object).

Reference

- [DynamicSymbol Class \[▶ 2266\]](#)
- [TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)
- [IValueAnySymbol.ReadAnyValue\(Type\) \[▶ 2758\]](#)
- [IValueAnySymbol.WriteAnyValue\(Object\) \[▶ 2764\]](#)

6.11.21.2.26 DynamicSymbol.WriteAnyValue Method

Writes the value represented by the managed value to this [Value \[▶ 2775\]](#)

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteAnyValue(
    Object managedValue
)
```

Parameters



managedValue	Type: System.Object The managed value.
--------------	---

Reference

- [DynamicSymbol Class \[▶ 2266\]](#)
- [TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)
- [DynamicSymbol.ReadAnyValue\(Type\) \[▶ 2300\]](#)
- [DynamicSymbol.UpdateAnyValue\(Object.\) \[▶ 2312\]](#)

6.11.21.2.27 DynamicSymbol.WriteRawValue Method

Overload List

	Name	Description
	WriteRawValue(Byte e.) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	WriteRawValue(Byte e., Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)

Reference

- [DynamicSymbol Class \[▶ 2266\]](#)
- [TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.21.2.27.1 DynamicSymbol.WriteRawValue Method (.Byte.)

Writes the raw value of the [IValueSymbol](#) [[▶ 2775](#)] (Ads Read / Write)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteRawValue(
    byte[] rawValue
)
```

Parameters

rawValue	Type: .System.Byte. The value as byte array.
----------	---

Implements

[IValueRawSymbol.WriteRawValue\(Byte.\)](#) [[▶ 2772](#)]

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[WriteRawValue Overload](#) [[▶ 2313](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.2.27.2 DynamicSymbol.WriteRawValue Method (.Byte., Int32)

Writes the raw value of the [IValueSymbol](#) [[▶ 2775](#)] (Ads Read / Write)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteRawValue(
    byte[] rawValue,
    int timeout
)
```

Parameters

rawValue	Type: .System.Byte. The value as byte array.
timeout	Type: System.Int32 The timeout.

Field Value

Type:
The value.

Implements

[IValueRawSymbol.WriteRawValue\(Byte, Int32\) \[► 2773\]](#)

Remarks

A negative timeout indicates that the Default Timeout for the communication will be used. 0 means timeout is switched off.

Reference

[DynamicSymbol Class \[► 2266\]](#)

[WriteRawValue Overload \[► 2313\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.2.28 DynamicSymbol.WriteRawValueAsync Method

Writes the raw value of the [IValueSymbol \[► 2775\]](#) (Ads Read / Write)

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultWriteAccess> WriteRawValueAsync (
    byte[] rawValue,
    CancellationToken cancel
)
```

Parameters

rawValue	Type: <code>.System.Byte</code> . The value as byte array.
cancel	Type: <code>System.Threading.CancellationToken</code> The cancellation token.

Return Value

Type: `Task.ResultWriteAccess [► 3217]`.

A task that represents the asynchronous read operation. The [ResultRead \[► 1143\]](#) parameter contains the total number of bytes read into the buffer ([ReadBytes \[► 1145\]](#)) and the [ErrorCode \[► 1120\]](#) after execution..

Implements

[IValueRawSymbol.WriteRawValueAsync\(Byte, CancellationToken\) \[► 2774\]](#)




Reference

[DynamicSymbol Class \[► 2266\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.2.29 DynamicSymbol.WriteValue Method

Overload List

	Name	Description
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266].
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266].

Reference

[DynamicSymbol Class](#) [▶ 2266]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.21.2.29.1 DynamicSymbol.WriteValue Method (Object)

Writes the specified value to the [DynamicSymbol](#) [▶ 2266].

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteValue(
    Object value
)
```

Parameters

value	Type: System.Object The value.
-------	-----------------------------------

Implements

[IValueSymbol.WriteValue\(Object\)](#) [▶ 2788]

Remarks

Examples

Dynamic Write access

```
namespace Sample
{
    using System;
    using System.Diagnostics;
    using System.Threading;
    using TwinCAT;
    using TwinCAT.Ads;
    using TwinCAT.Ads.TypeSystem;
    using TwinCAT.Ads.ValueAccess;
    using TwinCAT.TypeSystem;
    using TwinCAT.TypeSystem.Generic;
    using TwinCAT.ValueAccess;

    class SymbolBrowserProgramV2DynamicTree
    {
```

```

#region CODE_SAMPLE_SIMPLEDYNAMIC
/// <summary>
/// Defines the entry point of the application.
/// </summary>
/// <param name="args">The arguments.</param>
static async void Main(string[] args)
{
    // Get the AdsAddress from command-line arguments
    AmsAddress address = ArgParser.Parse(args);

    CancellationTokenSource cancelSource = new CancellationTokenSource();
    CancellationToken cancel = cancelSource.Token;

    using (AdsClient client = new AdsClient())
    {
        // Connect to the target device
        client.Connect(address);

        // Usage of "dynamic" Type and Symbols (>= .NET4 only)
        SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.DynamicTree);
        IAdsSymbolLoader dynLoader = (IAdsSymbolLoader)SymbolLoaderFactory.Create(client, settings);

#endregion

    // Set the Default setting for Notifications
    dynLoader.DefaultNotificationSettings = new NotificationSettings(AdsTransMode.OnChange, 200,
2000);

    // Get the Symbols (Dynamic Symbols)
    var resultSymbols = await ((IDynamicSymbolLoader)dynLoader).GetDynamicSymbolsAsync(cancel);

    dynamic dynamicSymbols = resultSymbols.Symbols;
    dynamic adsPort = dynamicSymbols.TwinCAT_SystemInfoVarList._AppInfo.AdsPort;

#region CODE_SAMPLE_SIMPLEDYNAMIC

    // Access Main Symbol with Dynamic Language Runtime support (DLR)
    // Dynamically created property "Main"
    //dynamic symMain = dynamicSymbols.Main;

    // Main is an 'VirtualSymbol' / Organizational unit that doesn't have a value
    // Calling ReadValue is not allowed
    //bool test = symMain.HasValue;
    //dynamic invalid = symMain.ReadValue();

    //Reading TaskInfo Value
    //
    With calling ReadValueAsync() a 'snapshot' of the Symbols Instance is taken (reading async)
    ResultReadValueAccess resultRead = await dynamicSymbols.TwinCAT_SystemInfoVarList._TaskInfo.
ReadValueAsync(cancel);
    dynamic vTaskInfoArray = resultRead.Value;

    // Getting the Snapshot time in UTC format
    DateTimeOffset timeStamp1 = vTaskInfoArray.TimeStamp;

    // Getting TaskInfo Symbol for Task 1
    dynamic symTaskInfo1 = dynamicSymbols.TwinCAT_SystemInfoVarList._TaskInfo[1];

    // Getting CycleCount Symbol
    dynamic symCycleCount = symTaskInfo1.CycleCount;

    // Take Snapshot value of the ApplicationInfo struct
    resultRead = await dynamicSymbols.TwinCAT_SystemInfoVarList._AppInfo.ReadValueAsync(cancel);
    dynamic vAppInfo = resultRead.Value;

    // Get the UTC Timestamp of the snapshot
    DateTimeOffset timeStamp2 = vAppInfo.TimeStamp;

    // Access the ProjectName of the ApplicationInfo Snapshot (type-safe!)
    string projectNameValue = vAppInfo.ProjectName;

    // Reading the CycleCount Value
    resultRead = await symTaskInfo1.CycleCount.ReadValueAsync(cancel);    // Taking a Value Sna
pshot
    int cycleCountValue = (int)resultRead.Value;
#endregion

    // Registering for dynamic "ValueChanged" events for the Values
    // Using Default Notification settings

```

```

symCycleCount.ValueChanged += new EventHandler<ValueChangedEventArgs>(cycleCount_ValueChanged);

// Override default notification settings
symTaskInfo1.NotificationSettings = new NotificationSettings(AdsTransMode.Cyclic, 500, 0);

// Register for ValueChanged event.
symTaskInfo1.ValueChanged += new EventHandler<ValueChangedEventArgs>(taskInfo1Value_ValueChanged); // Struct Type

Thread.Sleep(10000); // Sleep main thread for 10 Seconds
}
Console.WriteLine("CycleCount Changed events received: {0}", _cycleCountEvents);
Console.WriteLine("taskInfo1 Changed events received: {0}", _taskInfo1Events);

Console.WriteLine("");
Console.WriteLine("Press [Enter] for leave:");
Console.ReadLine();
}

static object _notificationSynchronizer = new object();
static int _cycleCountEvents = 0;

/// <summary>
/// Handler function for the CycleCount ValueChanged event.
/// </summary>
/// <param name="sender">The sender.</param>
/// <param name="e">The e.</param>
static void cycleCount_ValueChanged(object sender, ValueChangedEventArgs e)
{
    lock(_notificationSynchronizer)
    {
        Interlocked.Increment(ref _cycleCountEvents);
        // val is a type safe value of int!
        dynamic val = e.Value;
        uint intVal = val;

        DateTimeOffset changedTime = e.DateTime.ToLocalTime(); // Convert UTC to local time
        Console.WriteLine("CycleCount changed to: {0}, TimeStamp: {1}", intVal, changedTime.ToString(
("HH:mm:ss:fff"));
    }
}

static int _taskInfo1Events = 0;

/// <summary>
/// Handler function for the TaskInfo ValueChanged event.
/// </summary>
/// <param name="sender">The sender.</param>
/// <param name="e">The e.</param>
static void taskInfo1Value_ValueChanged(object sender, ValueChangedEventArgs e)
{
    lock (_notificationSynchronizer)
    {
        Interlocked.Increment(ref _taskInfo1Events);
        dynamic val = e.Value;
        DateTimeOffset changedTime = e.DateTime.ToLocalTime(); // Convert to local time

        // Val is a during Runtime created struct type and contains
        // the same Properties as related PLC object.
        int cycleTime = val.CycleTime;
        Console.WriteLine("TaskInfo1Value changed TimeStamp: {0}", changedTime.ToString("HH:mm:ss:fff"));
    }
}
}

```

Reference

[DynamicSymbol Class \[► 2266\]](#)

[WriteValue Overload \[► 2316\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.2.29.2 DynamicSymbol.WriteValue Method (Object, Int32)

Writes the specified value to the [DynamicSymbol](#) [[▶ 2266](#)].

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void WriteValue(
    Object value,
    int timeout
)
```

Parameters

value	Type: System.Object The value.
timeout	Type: System.Int32 The timeout in ms.

Implements

[IValueSymbol.WriteValue\(Object, Int32\)](#) [[▶ 2788](#)]

Remarks

Calling on primitive types, a call of this method will directly write this Value. On complex types (structs and arrays) it depends on the [ISymbolLoader](#) [[▶ 2714](#)] settings what will happen. In non dynamic modes: Only byte Arrays (of correct size) can be written) in dynamic mode: A Value that represents the value will be accepted also. A negative timeout indicates that the Default Timeout for the communication will be used. 0 means timeout is switched off.

Examples

Dynamic Write access

```
namespace Sample
{
    using System;
    using System.Diagnostics;
    using System.Threading;
    using TwinCAT;
    using TwinCAT.Ads;
    using TwinCAT.Ads.TypeSystem;
    using TwinCAT.Ads.ValueAccess;
    using TwinCAT.TypeSystem;
    using TwinCAT.TypeSystem.Generic;
    using TwinCAT.ValueAccess;

    class SymbolBrowserProgramV2DynamicTree
    {
        #region CODE_SAMPLE_SIMPLEDYNAMIC
        /// <summary>
        /// Defines the entry point of the application.
        /// </summary>
        /// <param name="args">The arguments.</param>
        static async void Main(string[] args)
        {
            // Get the AdsAddress from command-line arguments
            AmsAddress address = ArgParser.Parse(args);

            CancellationTokenSource cancelSource = new CancellationTokenSource();
            CancellationToken cancel = cancelSource.Token;

            using (AdsClient client = new AdsClient())
```

```

    {
        // Connect to the target device
        client.Connect(address);

        // Usage of "dynamic" Type and Symbols (>= .NET4 only)
        SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.DynamicTree);
        IAdsSymbolLoader dynLoader = (IAdsSymbolLoader)SymbolLoaderFactory.Create(client, settings);

        #endregion

        // Set the Default setting for Notifications
        dynLoader.DefaultNotificationSettings = new NotificationSettings(AdsTransMode.OnChange, 200,
2000);

        // Get the Symbols (Dynamic Symbols)
        var resultSymbols = await ((IDynamicSymbolLoader)dynLoader).GetDynamicSymbolsAsync(cancel);

        dynamic dynamicSymbols = resultSymbols.Symbols;
        dynamic adsPort = dynamicSymbols.TwinCAT_SystemInfoVarList._AppInfo.AdsPort;

        #region CODE_SAMPLE_SIMPLEDYNAMIC

        // Access Main Symbol with Dynamic Language Runtime support (DLR)
        // Dynamically created property "Main"
        //dynamic symMain = dynamicSymbols.Main;

        // Main is an 'VirtualSymbol' / Organizational unit that doesn't have a value
        // Calling ReadValue is not allowed
        //bool test = symMain.HasValue;
        //dynamic invalid = symMain.ReadValue();

        //Reading TaskInfo Value
        //
        With calling ReadValueAsync() a 'snapshot' of the Symbols Instance is taken (reading async)
        ResultReadValueAccess resultRead = await dynamicSymbols.TwinCAT_SystemInfoVarList._TaskInfo.
ReadValueAsync(cancel);
        dynamic vTaskInfoArray = resultRead.Value;

        // Getting the Snapshot time in UTC format
        DateTimeOffset timeStamp1 = vTaskInfoArray.TimeStamp;

        // Getting TaskInfo Symbol for Task 1
        dynamic symTaskInfo1 = dynamicSymbols.TwinCAT_SystemInfoVarList._TaskInfo[1];

        // Getting CycleCount Symbol
        dynamic symCycleCount = symTaskInfo1.CycleCount;

        // Take Snapshot value of the ApplicationInfo struct
        resultRead = await dynamicSymbols.TwinCAT_SystemInfoVarList._AppInfo.ReadValueAsync(cancel);
        dynamic vAppInfo = resultRead.Value;

        // Get the UTC Timestamp of the snapshot
        DateTimeOffset timeStamp2 = vAppInfo.TimeStamp;

        // Access the ProjectName of the ApplicationInfo Snapshot (type-safe!)
        string projectNameValue = vAppInfo.ProjectName;

        // Reading the CycleCount Value
        resultRead = await symTaskInfo1.CycleCount.ReadValueAsync(cancel);    // Taking a Value Sna
psshot
        int cycleCountValue = (int)resultRead.Value;
        #endregion

        // Registering for dynamic "ValueChanged" events for the Values
        // Using Default Notification settings
        symCycleCount.ValueChanged += new EventHandler<ValueChangedEventArgs>(cycleCount_ValueChange
d);

        // Override default notification settings
        symTaskInfo1.NotificationSettings = new NotificationSettings(AdsTransMode.Cyclic, 500, 0);

        // Register for ValueChanged event.
        symTaskInfo1.ValueChanged += new EventHandler<ValueChangedEventArgs>(taskInfo1Value_ValueCha
nged); // Struct Type

        Thread.Sleep(10000); // Sleep main thread for 10 Seconds
    }
    Console.WriteLine("CycleCount Changed events received: {0}", _cycleCountEvents);
    Console.WriteLine("taskInfo1 Changed events received: {0}", _taskInfo1Events);

```



```

        Console.WriteLine("");
        Console.WriteLine("Press [Enter] for leave:");
        Console.ReadLine();
    }

    static object _notificationSynchronizer = new object();
    static int _cycleCountEvents = 0;

    /// <summary>
    /// Handler function for the CycleCount ValueChanged event.
    /// </summary>
    /// <param name="sender">The sender.</param>
    /// <param name="e">The e.</param>
    static void cycleCount_ValueChanged(object sender, ValueChangedEventArgs e)
    {
        lock(_notificationSynchronizer)
        {
            Interlocked.Increment(ref _cycleCountEvents);
            // val is a type safe value of int!
            dynamic val = e.Value;
            uint intVal = val;

            DateTimeOffset changedTime = e.DateTime.ToLocalTime(); // Convert UTC to local time
            Console.WriteLine("CycleCount changed to: {0}, TimeStamp: {1}", intVal, changedTime.ToString(
("HH:mm:ss:fff"));
        }
    }

    static int _taskInfo1Events = 0;

    /// <summary>
    /// Handler function for the TaskInfo ValueChanged event.
    /// </summary>
    /// <param name="sender">The sender.</param>
    /// <param name="e">The e.</param>
    static void taskInfo1Value_ValueChanged(object sender, ValueChangedEventArgs e)
    {
        lock (_notificationSynchronizer)
        {
            Interlocked.Increment(ref _taskInfo1Events);
            dynamic val = e.Value;
            DateTimeOffset changedTime = e.DateTime.ToLocalTime(); // Convert to local time

            // Val is a during Runtime created struct type and contains
            // the same Properties as related PLC object.
            int cycleTime = val.CycleTime;
            Console.WriteLine("TaskInfo1Value changed TimeStamp: {0}", changedTime.ToString("HH:mm:ss:ff
f"));
        }
    }
}

```

Reference

[DynamicSymbol Class \[► 2266\]](#)

[WriteValue Overload \[► 2316\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.2.30 DynamicSymbol.WriteValueAsync Method

Writes the Value of the [IValueSymbol \[► 2775\]](#)

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultWriteAccess> WriteValueAsync (
    Object value,
    CancellationToken cancel
)
```

Parameters

value	Type: System.Object The value.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWriteAccess](#) [[▶ 3217](#)].

A tasks that represents the asynchronous 'ReadValue' operation. The read result is stored in the [ResultWriteAccess](#) [[▶ 3217](#)] return value and contains the [ErrorCode](#) [[▶ 3202](#)].

Implements

[IValueSymbol.WriteValueAsync\(Object, CancellationToken\)](#) [[▶ 2789](#)]

Remarks

Calling on primitive types, a call of this method will directly write this Value. On complex types (structs and arrays) it depends on the [ISymbolLoader](#) [[▶ 2714](#)] settings what will happen. In non dynamic modes: Only byte Arrays (of correct size) can be written) in dynamic mode: A Value that represents the value will be accepted also.

Reference



[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.3 DynamicSymbol Events

The [DynamicSymbol](#) [[▶ 2266](#)] type exposes the following members.

Events

	Name	Description
	RawValueChanged [▶ 2323]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed.
	ValueChanged [▶ 2323]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed.

Reference

[DynamicSymbol Class](#) [[▶ 2266](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.21.3.1 DynamicSymbol.RawValueChanged Event

Occurs when the RawValue of the [IValueSymbol](#) [▸ 2775] has changed.

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<RawValueChangedEventArgs> RawValueChanged
```

Value

Type: [System.EventHandler.RawValueChangedEventArgs](#) [▸ 2813].

Implements

[IValueRawSymbol.RawValueChanged](#) [▸ 2774]

Reference

[DynamicSymbol Class](#) [▸ 2266]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.21.3.2 DynamicSymbol.ValueChanged Event

Occurs when the (Primitive) value of the [IValueSymbol](#) [▸ 2775] has changed.

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public event EventHandler<ValueChangedEventArgs> ValueChanged
```

Value

Type: [System.EventHandler.ValueChangedEventArgs](#) [▸ 2971].

Implements

[IValueSymbol.ValueChanged](#) [▸ 2790]

Remarks

Examples

Use Dynamic Notifications

```
using System;
using System.Diagnostics;
using System.Threading;
using TwinCAT;
using TwinCAT.Ads;
using TwinCAT.Ads.TypeSystem;
using TwinCAT.Ads.ValueAccess;
using TwinCAT.TypeSystem;
```

```

using TwinCAT.TypeSystem.Generic;
using TwinCAT.ValueAccess;

namespace Sample
{
    class SymbolBrowserV2Notifications
    {
        /// <summary>
        /// Defines the entry point of the application.
        /// </summary>
        /// <param name="args">The arguments.</param>
        static void Main(string[] args)
        {
            // Parse the Command Line Parameters.
            AmsAddress address = ArgParser.Parse(args);

            #region DEFAULTNOTIFICATON_SAMPLE

            // Create AdsClient object
            using (AdsClient client = new AdsClient())
            {
                // No automatic Synchronization (necessary for Console applications without message loop)
                //client.Synchronize = false;

                // Connect to client
                client.Connect(address);

                // Usage of 'dynamic' type/symbol loader
                SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.DynamicTree, ValueAccessMode.IndexGroupOffsetPreferred);
                IAdsSymbolLoader dynLoader = (IAdsSymbolLoader)SymbolLoaderFactory.Create(client, settings);

                // Set the DefaultNotification Properties
                dynLoader.DefaultNotificationSettings = new NotificationSettings(AdsTransMode.ClientOnChange, 200, 2000);

                // Determine the symbols
                dynamic dynamicSymbols = ((IDynamicSymbolLoader)dynLoader).SymbolsDynamic;

                // Task 1 Symbol (build in symbol)
                dynamic task1Symbol = dynamicSymbols._TaskInfo[1];

                // CycleCount Symbol
                dynamic cycleCountSymbol = task1Symbol.CycleCount;

                // Override Notification Setting for Cycle Count Symbol
                cycleCountSymbol.NotificationSettings = new NotificationSettings(AdsTransMode.OnChange, 250, 0);

                // Register Dynamic Value Changed event.
                cycleCountSymbol.ValueChanged += new EventHandler<ValueChangedEventArgs>(cycleCount_ValueChanged);
            }
            #endregion

            // Sleep main thread to receive notifications
            Thread.Sleep(10000);
            #region DEFAULTNOTIFICATON_SAMPLE
            }
            #endregion

            Console.WriteLine("CycleCount Changed events received: {0}", _cycleCountEvents);
            Console.WriteLine("");
            Console.WriteLine("Press [Enter] for leave:");
            Console.ReadLine();
        }

        /// <summary>
        /// The cycle count event counter
        /// </summary>
        static int _cycleCountEvents = 0;

        /// <summary>
        /// Handler function for CycleCount changed events.
        /// </summary>
        /// <param name="sender">Event sender.</param>
        /// <param name="args">Event arguments.</param>
        static void cycleCount_ValueChanged(object sender, ValueChangedEventArgs args)
        {
            Interlocked.Increment(ref _cycleCountEvents);

            // Use Value as dynamic (type safe: INT) object.
            dynamic val = args.Value;

```

```

        int intVal = val;

        DateTimeOffset changedTime = args.DateTime.ToLocalTime(); // Convert UTC to local time
        Console.WriteLine("CycleCount changed to: {0}, TimeStamp: {1}", intVal, changedTime.ToString("HH:mm:ss:fff"));
    }
}

```

Reference





[DynamicSymbol Class \[► 2266\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.4 DynamicSymbol Operators

The [DynamicSymbol \[► 2266\]](#) type exposes the following members.

Operators

	Name	Description
 	Equality [► 2325]	Operator==
 	Inequality [► 2326]	Implements the != operator.

Reference

[DynamicSymbol Class \[► 2266\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.4.1 DynamicSymbol.Equality Operator

Operator==

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public static bool operator ==(
    DynamicSymbol? o1,
    DynamicSymbol? o2
)

```

Parameters

o1	Type: TwinCAT.TypeSystem.DynamicSymbol [► 2266] The o1.
o2	Type: TwinCAT.TypeSystem.DynamicSymbol [► 2266] The o2.

Return Value

Type: Boolean
The result of the operator.

Reference

[DynamicSymbol Class \[► 2266\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.4.2 DynamicSymbol.Inequality Operator

Implements the != operator.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static bool operator !=(
    DynamicSymbol? o1,
    DynamicSymbol? o2
)
```

Parameters

o1	Type: TwinCAT.TypeSystem.DynamicSymbol [► 2266] The o1.
o2	Type: TwinCAT.TypeSystem.DynamicSymbol [► 2266] The o2.

Return Value

Type: Boolean
The result of the operator.

Reference


[DynamicSymbol Class \[► 2266\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.5 DynamicSymbol Fields

The [DynamicSymbol \[► 2266\]](#) type exposes the following members.

Fields

	Name	Description
	syncObject [► 2327]	Synchronization object

Reference

[DynamicSymbol Class \[► 2266\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.21.5.1 DynamicSymbol.syncObject Field

Synchronization object

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected Object syncObject
```

Field Value

Type: Object

Reference

[DynamicSymbol Class \[▸ 2266\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.22 DynamicSymbolsCollection Class

Dynamic (Expandable) Symbols collection.

Inheritance Hierarchy

System.Object
 System.Dynamic.DynamicObject
 TwinCAT.TypeSystem.DynamicSymbolsCollection

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229



Syntax

C#



```
public sealed class DynamicSymbolsCollection : DynamicObject,
    IDynamicSymbolsCollection, IDynamicMetaObjectProvider, IEnumerable<ISymbol>, IEnumerable
```

The DynamicSymbolsCollection type exposes the following members.















Constructors







	Name	Description
	DynamicSymbolsCollection(IEnumerable<ISymbol>) [▸ 2330]	Initializes a new instance of the DynamicSymbolsCollection class (for internal use only)
	DynamicSymbolsCollection(SymbolCollection.ISymbol) [▸ 2330]	Initializes a new instance of the DynamicSymbolsCollection class (for internal use only)

Properties

	Name	Description
	Empty [▶ 2332]	Gets an empty collection.
	Item [▶ 2331]	Gets the DynamicSymbol [▶ 2266] with the specified name.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetDynamicMemberNames [▶ 2334]	Returns the enumeration of all dynamic member names. (Overrides DynamicObject.GetDynamicMemberNames..)
	GetEnumerator [▶ 2334]	Gets the enumerator.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject .)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryGetIndex	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicObject .)
	TryGetInstance [▶ 2335]	Tries to get the Instance [▶ 2549]. of the specified path.

	Name	Description
	TryGetMember [▶ 2335]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Overrides DynamicObject.TryGetMember(GetMemberBinder, Object.))
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject.)
	TryInvokeMember	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicObject.)
	TrySetIndex	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicObject.)
	TrySetMember	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicObject.)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)

Remarks



The DynamicSymbolsCollection collection adds dynamically its child Symbols as Members (for access like "Main.Symbol")

Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.22.1 DynamicSymbolsCollection Constructor

Overload List

	Name	Description
	DynamicSymbolsCollection(IEnumerable.ISymbol.) [▶ 2330]	Initializes a new instance of the DynamicSymbolsCollection [▶ 2327] class (for internal use only)
	DynamicSymbolsCollection(SymbolCollection.ISymbol.) [▶ 2330]	Initializes a new instance of the DynamicSymbolsCollection [▶ 2327] class (for internal use only)

Reference

[DynamicSymbolsCollection Class](#) [▶ 2327]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.22.1.1 DynamicSymbolsCollection Constructor (IEnumerable.ISymbol.)

Initializes a new instance of the [DynamicSymbolsCollection](#) [► 2327] class (for internal use only)

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DynamicSymbolsCollection(
    IEnumerable<ISymbol> symbols
)
```

Parameters

symbols	Type: System.Collections.Generic.IEnumerable.ISymbol [► 2691]. The symbols.
---------	--

Exceptions

Exception	Condition
ArgumentNullException	symbols

Reference

[DynamicSymbolsCollection Class](#) [► 2327]

[DynamicSymbolsCollection Overload](#) [► 2329]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.22.1.2 DynamicSymbolsCollection Constructor (SymbolCollection.ISymbol.)

Initializes a new instance of the [DynamicSymbolsCollection](#) [► 2327] class (for internal use only)

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DynamicSymbolsCollection(
    SymbolCollection<ISymbol> symbols
)
```

Parameters

symbols	Type: TwinCAT.TypeSystem.Generic.SymbolCollection [► 3165]. ISymbol [► 2691]. The symbols.
---------	---

Exceptions

Exception	Condition
ArgumentNullException	symbols

Reference

[DynamicSymbolsCollection Class](#) [► 2327]



[DynamicSymbolsCollection Overload \[▶ 2329\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.22.2 DynamicSymbolsCollection Properties

The [DynamicSymbolsCollection \[▶ 2327\]](#) type exposes the following members.

Properties

	Name	Description
	Empty [▶ 2332]	Gets an empty collection.
	Item [▶ 2331]	Gets the DynamicSymbol [▶ 2266] with the specified name.

Reference

[DynamicSymbolsCollection Class \[▶ 2327\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.22.2.1 DynamicSymbolsCollection.Item Property

Gets the [DynamicSymbol \[▶ 2266\]](#) with the specified name.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DynamicSymbol this[
    string name
] { get; }
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: [DynamicSymbol \[▶ 2266\]](#)
DynamicSymbol.

Exceptions

Exception	Condition
KeyNotFoundException	Symbol name not found in DynamicSymbols collection!

Reference

[DynamicSymbolsCollection Class \[▶ 2327\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.22.2 DynamicSymbolsCollection.Empty Property

Gets an empty collection.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static DynamicSymbolsCollection Empty { get; }
```

Return Value

Type: [DynamicSymbolsCollection \[▸ 2327\]](#)

DynamicSymbolsCollection.

Reference










[DynamicSymbolsCollection Class \[▸ 2327\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.22.3 DynamicSymbolsCollection Methods

The [DynamicSymbolsCollection \[▸ 2327\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetDynamicMemberNames [▸ 2334]	Returns the enumeration of all dynamic member names. (Overrides DynamicObject.GetDynamicMemberNames..)
	GetEnumerator [▸ 2334]	Gets the enumerator.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject .)

	Name	Description
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetIndex	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicObject.)
	TryGetInstance [▶ 2335]	Tries to get the Instance [▶ 2549] . of the specified path.
	TryGetMember [▶ 2335]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Overrides DynamicObject.TryGetMember(GetMemberBinder, Object.).)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject.)
	TryInvokeMember	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicObject.)
	TrySetIndex	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicObject.)
	TrySetMember	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicObject.)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)

Reference

[DynamicSymbolsCollection Class \[▶ 2327\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.22.3.1 DynamicSymbolsCollection.Empty Method

Gets an empty collection.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public static DynamicSymbolsCollection Empty()
```

Return Value

Type: [DynamicSymbolsCollection](#) [▶ 2327]
DynamicSymbolsCollection.

Reference

[DynamicSymbolsCollection Class](#) [▶ 2327]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.22.3.2 DynamicSymbolsCollection.GetDynamicMemberNames Method

Returns the enumeration of all dynamic member names.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override IEnumerable<string> GetDynamicMemberNames ()
```

Return Value

Type: [IEnumerable.String](#).
A sequence that contains dynamic member names.

Reference

[DynamicSymbolsCollection Class](#) [▶ 2327]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.22.3.3 DynamicSymbolsCollection.GetEnumerator Method

Gets the enumerator.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IEnumerator<ISymbol> GetEnumerator ()
```

Return Value

Type: [IEnumerator.ISymbol](#) [▶ 2691].
A [IEnumerator.T](#). that can be used to iterate through the collection.

Implements

[IEnumerable.T](#).GetEnumerator.

Reference[DynamicSymbolsCollection Class \[► 2327\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.22.3.4 DynamicSymbolsCollection.TryGetInstance Method**

Tries to get the [Instance \[► 2549\]](#). of the specified path.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public bool TryGetInstance(
    string instanceSpecifier,
    out ISymbol?? symbol
)
```

Parameters

instanceSpecifier	Type: System.String The instance path or Instance Name (dependent of Mode [► 3111] setting)
symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] . The symbol.

Return Value

Type: Boolean

true if the [Instance \[► 2549\]](#) is found; otherwise, false

Exceptions

Exception	Condition
ArgumentNullException	instancePath
ArgumentException	

Reference[DynamicSymbolsCollection Class \[► 2327\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.22.3.5 DynamicSymbolsCollection.TryGetMember Method**

Provides the implementation for operations that get member values. Classes derived from the [DynamicObject](#) class can override this method to specify dynamic behavior for operations such as getting a value for a property.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TryGetMember(
    GetMemberBinder binder,
    out Object?? result
)
```

Parameters

binder	Type: System.Dynamic.GetMemberBinder Provides information about the object that called the dynamic operation. The binder.Name property provides the name of the member on which the dynamic operation is performed. For example, for the Console.WriteLine(sampleObject.SampleProperty) statement, where sampleObject is an instance of the class derived from the DynamicObject class, binder.Name returns "SampleProperty". The binder.IgnoreCase property specifies whether the member name is case-sensitive.
result	Type: System.Object. The result of the get operation. For example, if the method is called for a property, you can assign the property value to result.

Return Value

Type: Boolean

true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a run-time exception is thrown.)

Reference

[DynamicSymbolsCollection Class \[► 2327\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.23 DynamicUnionInstance Class

Dynamic union instance

Inheritance Hierarchy

System.Object
 System.Dynamic.DynamicObject
 TwinCAT.TypeSystem.DynamicSymbol [► 2266]
 TwinCAT.TypeSystem.DynamicUnionInstance

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229


Syntax

C#







```
public sealed class DynamicUnionInstance : DynamicSymbol,
    IUnionInstance, ISymbol, IAttributedInstance, IInstance, IBitSize
```

The DynamicUnionInstance type exposes the following members.















Properties

	Name	Description
	InnerSymbol [► 2274]	Gets the inner symbol of this DynamicSymbol [► 2266] (Inherited from DynamicSymbol [► 2266] .)







	Name	Description
	AccessRights [▶ 2274]	Gets the access rights. (Inherited from DynamicSymbol [▶ 2266].)
	Attributes [▶ 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [▶ 2266].)
	BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from DynamicSymbol [▶ 2266].)
	ByteSize [▶ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [▶ 2266].)
	Category [▶ 2277]	Gets the category. (Inherited from DynamicSymbol [▶ 2266].)
	Comment [▶ 2277]	Gets the comment of the IInstance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266].)
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	FieldInstances [▶ 2342]	Gets the member instances of the Struct Instance [▶ 2666].
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances , what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266].)
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266].)
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DynamicSymbol [▶ 2266].)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from DynamicSymbol [▶ 2266].)
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266].)
	IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266].)
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266].)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266].)
	IsReference [▶ 2284]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
	Size [▶ 2286]	Gets the size of the Instance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)



Methods

	Name	Description
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)
	GetDynamicMemberNames [▶ 2345]	Returns the enumeration of all dynamic member names. (Overrides DynamicSymbol.GetDynamicMemberNames . [▶ 2293].)
	GetHashCode [▶ 2293]	Gets the HashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue . [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValueAsync [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue . [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)

	Name	Description
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject.)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetIndex	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicObject.)
	TryGetMember [▶ 2345]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Overrides DynamicSymbol.TryGetMember(GetMemberBinder, Object.) [▶ 2308].)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject.)
	TryInvokeMember	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicObject.)
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	TrySetIndex	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicObject.)
	TrySetMember [▶ 2346]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Overrides DynamicObject.TrySetMember(SetMemberBinder, Object.) .)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
 	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from DynamicSymbol [▶ 2266].)
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte. e.) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	WriteRawValue(Byte, Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266] .)
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266] .)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266] . (Inherited from DynamicSymbol [▶ 2266] .)
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266] . (Inherited from DynamicSymbol [▶ 2266] .)
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266] .)

Events

	Name	Description
	RawValueChanged [▶ 2323]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266] .)
	ValueChanged [▶ 2323]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266] .)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021] .)







Reference




[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)


6.11.23.1 DynamicUnionInstance Properties

The [DynamicUnionInstance \[▶ 2336\]](#) type exposes the following members.

Properties

	Name	Description
	InnerSymbol [▶ 2274]	Gets the inner symbol of this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266] .)
	AccessRights [▶ 2274]	Gets the access rights. (Inherited from DynamicSymbol [▶ 2266] .)
	Attributes [▶ 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [▶ 2266] .)
	BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from DynamicSymbol [▶ 2266] .)
	ByteSize [▶ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [▶ 2266] .)
	Category [▶ 2277]	Gets the category. (Inherited from DynamicSymbol [▶ 2266] .)

	Name	Description
	Comment [▶ 2277]	Gets the comment of the IInstance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266].)
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	FieldInstances [▶ 2342]	Gets the member instances of the Struct Instance [▶ 2666].
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances , what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266].)
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266].)
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DynamicSymbol [▶ 2266].)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from DynamicSymbol [▶ 2266].)
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266].)
	IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266].)
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266].)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266].)
	IsReference [▶ 2284]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
	Size [▶ 2286]	Gets the size of the IInstance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)

Reference

[DynamicUnionInstance Class](#) [▶ 2336]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.23.1.1 DynamicUnionInstance.FieldInstances Property

Gets the member instances of the [Struct Instance](#) [▶ 2666].

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISymbolCollection<ISymbol> FieldInstances { get; }
```

Property Value

Type: [ISymbolCollection](#) [▶ 2700].[ISymbol](#) [▶ 2691].

The member instances.

Implements

[IUnionInstance.FieldInstances](#) [▶ 2740]

Reference





[DynamicUnionInstance Class](#) [▶ 2336]



















[TwinCAT.TypeSystem Namespace](#) [▶ 2083]
















6.11.23.2 DynamicUnionInstance Methods

The [DynamicUnionInstance](#) [▶ 2336] type exposes the following members.

Methods

	Name	Description
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)
	GetDynamicMemberNames [▶ 2345]	Returns the enumeration of all dynamic member names. (Overrides DynamicSymbol.GetDynamicMemberNames . [▶ 2293].)
	GetHashCode [▶ 2293]	Gets the HashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)

	Name	Description
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue. [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValueAsyn c [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject.)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject.)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetIndex	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicObject.)
	TryGetMember [▶ 2345]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Overrides DynamicSymbol.TryGetMember(GetMemberBinder, Object.) [▶ 2308].)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject.)
	TryInvokeMember	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicObject.)

	Name	Description
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	TrySetIndex	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicObject .)
	TrySetMember [▶ 2346]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Overrides DynamicObject.TrySetMember(SetMemberBinder, Object) .)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject .)
 	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from DynamicSymbol [▶ 2266].)
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte, Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[DynamicUnionInstance Class](#) [▶ 2336]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.23.2.1 DynamicUnionInstance.GetDynamicMemberNames Method

Returns the enumeration of all dynamic member names.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override IEnumerable<string> GetDynamicMemberNames ()
```

Return Value

Type: [IEnumerable.String](#).

A sequence that contains dynamic member names.

Reference

[DynamicUnionInstance Class](#) [[▶ 2336](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.23.2.2 DynamicUnionInstance.TryGetMember Method

Provides the implementation for operations that get member values. Classes derived from the [DynamicObject](#) class can override this method to specify dynamic behavior for operations such as getting a value for a property.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TryGetMember (
    GetMemberBinder binder,
    out Object?? result
)
```

Parameters

binder	Type: System.Dynamic.GetMemberBinder Provides information about the object that called the dynamic operation. The <code>binder.Name</code> property provides the name of the member on which the dynamic operation is performed. For example, for the <code>Console.WriteLine(sampleObject.SampleProperty)</code> statement, where <code>sampleObject</code> is an instance of the class derived from the DynamicObject class, <code>binder.Name</code> returns "SampleProperty". The <code>binder.IgnoreCase</code> property specifies whether the member name is case-sensitive.
result	Type: System.Object . The result of the get operation. For example, if the method is called for a property, you can assign the property value to <code>result</code> .

Return Value

Type: [Boolean](#)

true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a run-time exception is thrown.)

Reference

[DynamicUnionInstance Class \[► 2336\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.23.2.3 DynamicUnionInstance.TrySetMember Method

Provides the implementation for operations that set member values. Classes derived from the `DynamicObject` class can override this method to specify dynamic behavior for operations such as setting a value for a property.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TrySetMember(
    SetMemberBinder binder,
    Object? value
)
```

Parameters

binder	Type: <code>System.Dynamic.SetMemberBinder</code> Provides information about the object that called the dynamic operation. The <code>binder.Name</code> property provides the name of the member to which the value is being assigned. For example, for the statement <code>sampleObject.SampleProperty = "Test"</code> , where <code>sampleObject</code> is an instance of the class derived from the <code>DynamicObject</code> class, <code>binder.Name</code> returns "SampleProperty". The <code>binder.IgnoreCase</code> property specifies whether the member name is case-sensitive.
value	Type: <code>System.Object</code> The value to set to the member. For example, for <code>sampleObject.SampleProperty = "Test"</code> , where <code>sampleObject</code> is an instance of the class derived from the <code>DynamicObject</code> class, the value is "Test".

Return Value

Type: `Boolean`

true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a language-specific run-time exception is thrown.)

Reference


[DynamicUnionInstance Class \[► 2336\]](#)


[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.23.3 DynamicUnionInstance Events

The [DynamicUnionInstance \[► 2336\]](#) type exposes the following members.

Events

	Name	Description
	RawValueChanged [► 2323]	Occurs when the RawValue of the IValueSymbol [► 2775] has changed. (Inherited from DynamicSymbol [► 2266] .)

	Name	Description
	ValueChanged [▶ 2323]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)

Reference

[DynamicUnionInstance Class](#) [▶ 2336]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.24 DynamicValue Class

Dynamic value (uses [RuntimeBinding](#) for [ISymbol](#) [▶ 2691] value reading / writing).

Inheritance Hierarchy

- System.Object
- System.Dynamic.DynamicObject
- TwinCAT.TypeSystem.DynamicValue
 - [TwinCAT.TypeSystem.DynamicArrayValue](#) [▶ 3042]
 - [TwinCAT.TypeSystem.DynamicEnumValue](#) [▶ 3048]
 - [TwinCAT.TypeSystem.DynamicPointerValue](#) [▶ 2196]
 - [TwinCAT.TypeSystem.DynamicReferenceValue](#) [▶ 2217]

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229











Syntax

C#




















```
public class DynamicValue : DynamicObject,
    IDynamicValue, IDynamicMetaObjectProvider, IValue, IStructValue, IArrayValue
```
















The [DynamicValue](#) type exposes the following members.

Properties

	Name	Description
	Age [▶ 2351]	Gets the age of the value (last successful read of the value)
	CachedRaw [▶ 2351]	Gets the cached Raw internal Data.
	DataType [▶ 2352]	Gets the data type bound to this IValue [▶ 2745]
	IsPrimitive [▶ 2353]	Gets a value indicating whether this IValue [▶ 2745] is a primitive value.
	ParentValue [▶ 2356]	Gets the parent value.
	ResolvedType [▶ 2353]	Gets the resolved type.
	Symbol [▶ 2353]	Gets the symbol that is bound to this value.
	TimeStamp [▶ 2354]	Gets the Time stamp of the last successful read of the Value.
	UpdateMode [▶ 2354]	Gets / Sets the update mode
	ValueFactory [▶ 2355]	The value factory

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetDynamicMemberNames [▶ 2358]	Returns the enumeration of all dynamic member names. (Overrides DynamicObject.GetDynamicMemberNames..)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Read [▶ 2359]	Reads the value (via ADS)
	ReadAsync [▶ 2359]	read as an asynchronous operation.
	ResolveValue [▶ 2360]	Resolves the Value object to its primitive value.
	setMemberValue [▶ 2374]	Writes the specified member element.
	ToString [▶ 2361]	Returns a String that represents this instance. (Overrides Object.ToString..)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject.)
	TryConvert [▶ 2361]	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Overrides DynamicObject.TryConvert(ConvertBinder, Object..).)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetArrayElementValues [▶ 2362]	Returns Array Element values.
	TryGetIndex [▶ 2363]	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Overrides DynamicObject.TryGetIndex(GetIndexBinder, .Object., Object..).)

	Name	Description
	TryGetIndexValue(Int32, Object.) [▶ 2364]	Reads the specified array element.
	TryGetIndexValue(Object, Object.) [▶ 2364]	Tries the get index value.
	TryGetMember [▶ 2365]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Overrides DynamicObject.TryGetMember(GetMemberBinder, Object.))
	TryGetMemberValue [▶ 2366]	Tries the get member value.
	TryInvoke [▶ 2366]	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Overrides DynamicObject.TryInvoke(InvokeBinder, .Object., Object.))
	TryInvokeMember [▶ 2367]	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Overrides DynamicObject.TryInvokeMember(InvokeMemberBinder, .Object., Object.))
	TryResolveValue [▶ 2368]	Tries to resolves the Value object to its primitive value.
	TrySetIndex [▶ 2369]	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Overrides DynamicObject.TrySetIndex(SetIndexBinder, .Object., Object.))
	TrySetIndexValue(Int32, Object) [▶ 2370]	Tries to set the indexed value on Arrays
	TrySetIndexValue(Object, Object) [▶ 2370]	Tries to set the indexed value on Arrays
	TrySetMember [▶ 2371]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Overrides DynamicObject.TrySetMember(SetMemberBinder, Object.))
	TrySetMemberValue [▶ 2372]	Tries to Set a Member/Property Value
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
	Write [▶ 2372]	Writes the value (via ADS)
	WriteAsync [▶ 2373]	write as an asynchronous operation.

Remarks

The DynamicValue adds dynamic run time behaviour to the [IValue \[▸ 2745\]](#)/[Value \[▸ 2745\]](#). That means e.g. for struct values that .NET Properties are on-the-fly defined and dispatched at runtime just like defined in the structs structs data type definition. Another example is the access of Array Element values through indexes.

Examples

Sample for the dynamic resolution of Symbols and reading values:

Dynamic Symbol access

```

/// <summary>
/// Defines the entry point of the application.
/// </summary>
/// <param name="args">The arguments.</param>
static async void Main(string[] args)
{
    // Get the AdsAddress from command-line arguments
    AmsAddress address = ArgParser.Parse(args);

    CancellationTokenSource cancelSource = new CancellationTokenSource();
    CancellationToken cancel = cancelSource.Token;

    using (AdsClient client = new AdsClient())
    {
        // Connect to the target device
        client.Connect(address);

        // Usage of "dynamic" Type and Symbols (>= .NET4 only)
        SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.DynamicTree);
        IAdsSymbolLoader dynLoader = (IAdsSymbolLoader)SymbolLoaderFactory.Create(client, settings);
    }
}

```

Reference




[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

[TwinCAT.TypeSystem.DynamicSymbol \[▸ 2266\]](#)

[System.Dynamic.DynamicObject](#)

[TwinCAT.TypeSystem.IValue \[▸ 2745\]](#)





Also see about this







-  [DynamicValue.ReadMember Method \[▸ 2360\]](#)
-  [DynamicValue.TrySetIndexValue Method \[▸ 2369\]](#)
-  [DynamicValue.WriteMember Method \[▸ 2373\]](#)

6.11.24.1 DynamicValue Properties

The [DynamicValue \[▸ 2347\]](#) type exposes the following members.

Properties

	Name	Description
	Age [▸ 2351]	Gets the age of the value (last successful read of the value)
	CachedRaw [▸ 2351]	Gets the cached Raw internal Data.
	DataType [▸ 2352]	Gets the data type bound to this IValue [▸ 2745]
	IsPrimitive [▸ 2353]	Gets a value indicating whether this IValue [▸ 2745] is a primitive value.

	Name	Description
	ParentValue [▶ 2356]	Gets the parent value.
	ResolvedType [▶ 2353]	Gets the resolved type.
	Symbol [▶ 2353]	Gets the symbol that is bound to this value.
	TimeStamp [▶ 2354]	Gets the Time stamp of the last successful read of the Value.
	UpdateMode [▶ 2354]	Gets / Sets the update mode
	ValueFactory [▶ 2355]	The value factory

Reference

[DynamicValue Class](#) [[▶ 2347](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.24.1.1 DynamicValue.Age Property

Gets the age of the value (last successful read of the value)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public TimeSpan Age { get; }
```

Property Value

Type: TimeSpan

The age.

Implements

[IValue.Age](#) [[▶ 2746](#)]

Exceptions

Exception	Condition
NotImplementedException	

Reference

[DynamicValue Class](#) [[▶ 2347](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

[DynamicValue.TimeStamp](#) [[▶ 2354](#)]

6.11.24.1.2 DynamicValue.CachedRaw Property

Gets the cached Raw internal Data.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyMemory<byte> CachedRaw { get; }
```

Property Value

Type: [ReadOnlyMemory.Byte](#).
The raw cached data.

Implements

[IValue.CachedRaw](#) [▶ 2746]

Reference

[DynamicValue Class](#) [▶ 2347]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.24.1.3 DynamicValue.DataType Property

Gets the data type bound to this [IValue](#) [▶ 2745]

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataType? DataType { get; }
```

Property Value

Type: [IDataType](#) [▶ 2475]
The type of the data.

Implements

[IValue.DataType](#) [▶ 2747]

Exceptions

Exception	Condition
NotImplementedException	

Reference

[DynamicValue Class](#) [▶ 2347]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.24.1.4 DynamicValue.IsPrimitive Property

Gets a value indicating whether this [IValue](#) [▸ 2745] is a primitive value.

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsPrimitive { get; }
```

Property Value

Type: Boolean

true if this instance is primitive; otherwise, false.

Implements

[IValue.IsPrimitive](#) [▸ 2747]

Reference

[DynamicValue Class](#) [▸ 2347]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.24.1.5 DynamicValue.ResolvedType Property

Gets the resolved type.

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected IDataType? ResolvedType { get; }
```

Property Value

Type: [IDataType](#) [▸ 2475]

Resolved type.

Reference

[DynamicValue Class](#) [▸ 2347]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.24.1.6 DynamicValue.Symbol Property

Gets the symbol that is bound to this value.

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISymbol Symbol { get; }
```

Property Value

Type: [ISymbol](#) [[▶ 2691](#)]

The symbol.

Implements

[IValue.Symbol](#) [[▶ 2748](#)]

Reference

[DynamicValue Class](#) [[▶ 2347](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.24.1.7 DynamicValue.TimeStamp Property

Gets the Time stamp of the last successful read of the Value.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTimeOffset TimeStamp { get; }
```

Property Value

Type: DateTimeOffset

The read time stamp.

Implements

[IValue.TimeStamp](#) [[▶ 2748](#)]

Reference

[DynamicValue Class](#) [[▶ 2347](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.24.1.8 DynamicValue.UpdateMode Property

Gets / Sets the update mode

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ValueUpdateMode UpdateMode { get; set; }
```

Property Value

Type: [ValueUpdateMode](#) [▶ 3229]

The update mode.

Implements

[IValue.UpdateMode](#) [▶ 2748]

Remarks

The default value is initialized by the creating Value Factory.

Mode	Description
Immediately [▶ 3229]	Writes the values of this DynamicValue [▶ 2347]"/> instantly when setting its value or the value of its child members/elements.
Triggered [▶ 3229]	Caches internally the value of this DynamicValue [▶ 2347]"/> until the Write . [▶ 2372] method is called. This reduces ADS roundtrips, if one or more member/element values should be changed. Furthermore the write on the destination system happens consistently in one ADS Write operation, which could be important for dependent properties/members/elements.




Reference

[DynamicValue Class](#) [▶ 2347]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

[TwinCAT.ValueAccess.ValueUpdateMode](#) [▶ 3229]

Also see about this

-  [ValueUpdateMode Enumeration](#) [▶ 159]
-  [ValueUpdateMode Enumeration](#) [▶ 159]
-  [ValueUpdateMode Enumeration](#) [▶ 159]

6.11.24.1.9 DynamicValue.ValueFactory Property

The value factory

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public IAccessorValueFactory ValueFactory { get; }
```

Property Value

Type: [IAccessorValueFactory](#) [▶ 3192]

Implements

[IDynamicValue.ValueFactory](#) [▶ 2506]

Reference

[DynamicValue Class](#) [► 2347]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.24.1.10 DynamicValue.ParentValue Property

Gets the parent value.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDynamicValue? ParentValue { get; }
```

Property Value

Type: [IDynamicValue](#) [► 2504]

The parent value.

Implements

[IDynamicValue.ParentValue](#) [► 2506]

Reference







[DynamicValue Class](#) [► 2347]



















[TwinCAT.TypeSystem Namespace](#) [► 2083]











6.11.24.2 DynamicValue Methods

The [DynamicValue](#) [► 2347] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetDynamicMemberNames [► 2358]	Returns the enumeration of all dynamic member names. (Overrides DynamicObject.GetDynamicMemberNames..)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object.)

	Name	Description
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Read [▶ 2359]	Reads the value (via ADS)
	ReadAsync [▶ 2359]	read as an asynchronous operation.
	ResolveValue [▶ 2360]	Resolves the Value object to its primitive value.
	setMemberValue [▶ 2374]	Writes the specified member element.
	ToString [▶ 2361]	Returns a String that represents this instance. (Overrides Object.ToString..)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject.)
	TryConvert [▶ 2361]	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Overrides DynamicObject.TryConvert(ConvertBinder, Object..))
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetArrayElement Values [▶ 2362]	Returns Array Element values.
	TryGetIndex [▶ 2363]	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Overrides DynamicObject.TryGetIndex(GetIndexBinder, .Object., Object..))
	TryGetIndexValue(.Int32., Object.) [▶ 2364]	Reads the specified array element.
	TryGetIndexValue(.Object., Object.) [▶ 2364]	Tries the get index value.
	TryGetMember [▶ 2365]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Overrides DynamicObject.TryGetMember(GetMemberBinder, Object..))
	TryGetMemberValue [▶ 2366]	Tries the get member value.
	TryInvoke [▶ 2366]	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Overrides DynamicObject.TryInvoke(InvokeBinder, .Object., Object..))




	Name	Description
	TryInvokeMember [▶ 2367]	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Overrides DynamicObject.TryInvokeMember(InvokeMemberBinder, .Object., Object).)
	TryResolveValue [▶ 2368]	Tries to resolves the Value object to its primitive value.
	TrySetIndex [▶ 2369]	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Overrides DynamicObject.TrySetIndex(SetIndexBinder, .Object., Object).)
	TrySetIndexValue(Int32, Object) [▶ 2370]	Tries to set the indexed value on Arrays
	TrySetIndexValue(Object, Object) [▶ 2370]	Tries to set the indexed value on Arrays
	TrySetMember [▶ 2371]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Overrides DynamicObject.TrySetMember(SetMemberBinder, Object).)
	TrySetMemberValue [▶ 2372]	Tries to Set a Member/Property Value
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
	Write [▶ 2372]	Writes the value (via ADS)
	WriteAsync [▶ 2373]	write as an asynchronous operation.

Reference

[DynamicValue Class](#) [▶ 2347]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

Also see about this

-  [DynamicValue.ReadMember Method](#) [▶ 2360]
-  [DynamicValue.TrySetIndexValue Method](#) [▶ 2369]
-  [DynamicValue.WriteMember Method](#) [▶ 2373]

6.11.24.2.1 DynamicValue.GetDynamicMemberNames Method

Returns the enumeration of all dynamic member names.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override IEnumerable<string> GetDynamicMemberNames ()
```

Return Value

Type: `IEnumerable.String`.
A sequence that contains dynamic member names.

Reference

[DynamicValue Class](#) [► 2347]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.24.2.2 DynamicValue.Read Method

Reads the value (via ADS)

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Read()
```

Implements

[IValue.Read.](#) [► 2749]

Exceptions

Exception	Condition
SymbolException [► 2933]	

Reference

[DynamicValue Class](#) [► 2347]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.24.2.3 DynamicValue.ReadAsync Method

read as an asynchronous operation.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultAccess> ReadAsync(  
    CancellationToken cancel  
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultAccess](#) [[▶ 3197](#)].
[Task<ReadValueResult>](#).

Implements

[IValue.ReadAsync\(CancellationToken\)](#) [[▶ 2750](#)]

Reference

[DynamicValue Class](#) [[▶ 2347](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.24.2.4 DynamicValue.ReadMember Method

Reads the specified member element.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
protected virtual Object ReadMember(  
    ISymbol memberInstance  
)
```

Parameters

memberInstance Type: [TwinCAT.TypeSystem.ISymbol](#) [[▶ 2691](#)]
The member instance.

Return Value

Type: [Object](#)

Reference

[DynamicValue Class](#) [[▶ 2347](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.24.2.5 DynamicValue.ResolveValue Method

Resolves the Value object to its primitive value.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object ResolveValue(  
    bool resolveEnumToPrimitive  
)
```

Parameters

resolveEnumToPrimitive	Type: System.Boolean if set to true EnumValue [2524]s are resolved to their primitives also.
------------------------	---

Return Value

Type: Object
System.Object.

Implements

[IValue.ResolveValue\(Boolean\)](#) [[2750](#)]

Remarks

If the value is not primitive, this method returns the [IValue](#) [[2745](#)] itself.

Reference

[DynamicValue Class](#) [[2347](#)]

[TwinCAT.TypeSystem Namespace](#) [[2083](#)]

6.11.24.2.6 DynamicValue.ToString Method

Returns a String that represents this instance.

Namespace: [TwinCAT.TypeSystem](#) [[2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```

Return Value

Type: String
A String that represents this instance.

Reference

[DynamicValue Class](#) [[2347](#)]

[TwinCAT.TypeSystem Namespace](#) [[2083](#)]

6.11.24.2.7 DynamicValue.TryConvert Method

Provides implementation for type conversion operations. Classes derived from the `DynamicObject` class can override this method to specify dynamic behavior for operations that convert an object from one type to another.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TryConvert(
    ConvertBinder binder,
    out Object?? result
)
```

Parameters

binder	Type: System.Dynamic.ConvertBinder Provides information about the conversion operation. The binder.Type property provides the type to which the object must be converted. For example, for the statement (String)sampleObject in C# (CType(sampleObject, Type) in Visual Basic), where sampleObject is an instance of the class derived from the DynamicObject class, binder.Type returns the String type. The binder.Explicit property provides information about the kind of conversion that occurs. It returns true for explicit conversion and false for implicit conversion.
result	Type: System.Object. The result of the type conversion operation.

Return Value

Type: Boolean

true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a language-specific run-time exception is thrown.)

Reference

[DynamicValue Class](#) [[▶ 2347](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.24.2.8 DynamicValue.TryGetArrayElementValues Method

Returns Array Element values.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetArrayElementValues(
    out IEnumerable<Object>? elementValues
)
```

Parameters

elementValues	Type: System.Collections.Generic.IEnumerable.Object.. The element values.
---------------	--

Return Value

Type: Boolean

true if the values are returned, false otherwise.

Implements

[IArrayValue.TryGetArrayElementValues\(IEnumerable.Object..\) \[▸ 2468\]](#)

Reference

[DynamicValue Class \[▸ 2347\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.24.2.9 DynamicValue.TryGetIndex Method

Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TryGetIndex(
    GetIndexBinder binder,
    Object[] indexes,
    out Object?? result
)
```

Parameters

binder	Type: System.Dynamic.GetIndexBinder Provides information about the operation.
indexes	Type: System.Object. The indexes that are used in the operation. For example, for the sampleObject[3] operation in C# (sampleObject(3) in Visual Basic), where sampleObject is derived from the DynamicObject class, indexes[0] is equal to 3.
result	Type: System.Object. The result of the index operation.

Return Value

Type: Boolean
true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a run-time exception is thrown.)


Reference


[DynamicValue Class \[▸ 2347\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.24.2.10 DynamicValue.TryGetIndexValue Method

Overload List

	Name	Description
	TryGetIndexValue(Int32, Object.) [▸ 2364]	Reads the specified array element.

	Name	Description
	TryGetIndexValue(Object, Object.) [▶ 2364]	Tries the get index value.

Reference

[DynamicValue Class](#) [▶ 2347]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.24.2.10.1 DynamicValue.TryGetIndexValue Method (.Int32., Object.)

Reads the specified array element.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetIndexValue(
    int[] indices,
    out Object?? value
)
```

Parameters

indices	Type: .System.Int32. The indices.
value	Type: System.Object. The value.

Return Value

Type: Boolean
System.Object.

Implements

[IArrayValue.TryGetIndexValue\(.Int32., Object.\)](#) [▶ 2468]

Reference

[DynamicValue Class](#) [▶ 2347]

[TryGetIndexValue Overload](#) [▶ 2363]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.24.2.10.2 DynamicValue.TryGetIndexValue Method (.Object., Object.)

Tries the get index value.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetIndexValue(
    Object[] indexes,
    out Object?? result
)
```

Parameters

indexes	Type: .System.Object. The indexes.
result	Type: System.Object. The result.

Return Value

Type: Boolean
true if the value is returned, false otherwise.

Reference

[DynamicValue Class](#) [► 2347]

[TryGetIndexValue Overload](#) [► 2363]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.24.2.11 DynamicValue.TryGetMember Method

Provides the implementation for operations that get member values. Classes derived from the `DynamicObject` class can override this method to specify dynamic behavior for operations such as getting a value for a property.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TryGetMember(
    GetMemberBinder binder,
    out Object?? result
)
```

Parameters

binder	Type: System.Dynamic.GetMemberBinder Provides information about the object that called the dynamic operation. The <code>binder.Name</code> property provides the name of the member on which the dynamic operation is performed. For example, for the <code>Console.WriteLine(sampleObject.SampleProperty)</code> statement, where <code>sampleObject</code> is an instance of the class derived from the <code>DynamicObject</code> class, <code>binder.Name</code> returns "SampleProperty". The <code>binder.IgnoreCase</code> property specifies whether the member name is case-sensitive.
result	Type: System.Object. The result of the get operation. For example, if the method is called for a property, you can assign the property value to <code>result</code> .

Return Value

Type: Boolean

true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a run-time exception is thrown.)

Reference

[DynamicValue Class](#) [► 2347]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.24.2.12 DynamicValue.TryGetMemberValue Method

Tries the get member value.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public virtual bool TryGetMemberValue(
    string name,
    out Object?? result
)
```

Parameters

name	Type: System.String The name.
result	Type: System.Object. The result.

Return Value

Type: Boolean

true if the member value could be determined, false otherwise.

Implements

[IStructValue.TryGetMemberValue\(String, Object.\)](#) [► 2680]

Exceptions

Exception	Condition
SymbolException [► 2933]	

Reference

[DynamicValue Class](#) [► 2347]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.24.2.13 DynamicValue.TryInvoke Method

Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TryInvoke(
    InvokeBinder binder,
    Object?[]? args,
    out Object?? result
)
```

Parameters

binder	Type: System.Dynamic.InvokeBinder Provides information about the invoke operation.
args	Type: .System.Object. The arguments that are passed to the object during the invoke operation. For example, for the sampleObject(100) operation, where sampleObject is derived from the DynamicObject class, args is equal to 100.
result	Type: System.Object. The result of the object invocation.

Return Value

Type: Boolean
 true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a language-specific run-time exception is thrown.)

Reference

- [DynamicValue Class \[▸ 2347\]](#)
- [TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.24.2.14 DynamicValue.TryInvokeMember Method

Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TryInvokeMember(
    InvokeMemberBinder binder,
    Object?[]? args,
    out Object?? result
)
```

Parameters

binder	Type: System.Dynamic.InvokeMemberBinder Provides information about the dynamic operation. The binder.Name property provides the name of the member on which the dynamic operation is performed. For example, for the statement sampleObject.SampleMethod(100), where sampleObject is an instance of the class derived from the DynamicObject class, binder.Name returns "SampleMethod". The binder.IgnoreCase property specifies whether the member name is case-sensitive.
--------	--

args	Type: .System.Object. The arguments that are passed to the object member during the invoke operation. For example, for the statement sampleObject.SampleMethod(100), where sampleObject is derived from the DynamicObject class, args[0] is equal to 100.
result	Type: System.Object. The result of the member invocation.

Return Value

Type: Boolean

true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a language-specific run-time exception is thrown.)

Reference

[DynamicValue Class](#) [► 2347]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.24.2.15 DynamicValue.TryResolveValue Method

Tries to resolves the Value object to its primitive value.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public bool TryResolveValue(
    bool resolveEnumToPrimitive,
    out Object?? value
)
```

Parameters

resolveEnumToPrimitive	Type: System.Boolean if set to true EnumValue [► 2524]s are resolved to their primitives also.
value	Type: System.Object. The value.

Return Value

Type: Boolean

true if value can be resolved, false otherwise.

Implements

[IValue.TryResolveValue\(Boolean, Object.\)](#) [► 2751]

Reference

[DynamicValue Class](#) [► 2347]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.24.2.16 DynamicValue.TrySetIndex Method

Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TrySetIndex(
    SetIndexBinder binder,
    Object[] indexes,
    Object? value
)
```

Parameters

binder	Type: System.Dynamic.SetIndexBinder Provides information about the operation.
indexes	Type: .System.Object. The indexes that are used in the operation. For example, for the sampleObject[3] = 10 operation in C# (sampleObject(3) = 10 in Visual Basic), where sampleObject is derived from the DynamicObject class, indexes[0] is equal to 3.
value	Type: System.Object The value to set to the object that has the specified index. For example, for the sampleObject[3] = 10 operation in C# (sampleObject(3) = 10 in Visual Basic), where sampleObject is derived from the DynamicObject class, value is equal to 10.

Return Value

Type: Boolean

true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a language-specific run-time exception is thrown.)

Reference

[DynamicValue Class](#) [[▶ 2347](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.24.2.17 DynamicValue.TrySetIndexValue Method

Overload List

	Name	Description
	TrySetIndexValue(In t32., Object) [▶ 2370]	Tries to set the indexed value on Arrays
	TrySetIndexValue(Object., Object) [▶ 2370]	Tries to set the indexed value on Arrays

Reference

[DynamicValue Class](#) [[▶ 2347](#)]

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

Also see about this

 [TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.24.2.17.1 DynamicValue.TrySetIndexValue Method (.Int32., Object)

Tries to set the indexed value on Arrays

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TrySetIndexValue(
    int[] indices,
    Object? value
)
```

Parameters

indices	Type: .System.Int32. The indices.
value	Type: System.Object The value.

Return Value

Type: Boolean

true if succeeded, false otherwise.

Exceptions

Exception	Condition
ArgumentNullException	indices

Reference

[DynamicValue Class \[► 2347\]](#)

[TrySetIndexValue Overload \[► 2369\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.24.2.17.2 DynamicValue.TrySetIndexValue Method (.Object., Object)

Tries to set the indexed value on Arrays

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TrySetIndexValue(
    Object[] indices,
    Object? value
)
```

Parameters

indices	Type: .System.Object. The indexes.
value	Type: System.Object The value.

Return Value

Type: Boolean
true if succeeded, false otherwise.

Implements

[IArrayValue.TrySetIndexValue\(.Object., Object\) \[▸ 2469\]](#)

Reference

[DynamicValue Class \[▸ 2347\]](#)

[TrySetIndexValue Overload \[▸ 2369\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.24.2.18 DynamicValue.TrySetMember Method

Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TrySetMember(
    SetMemberBinder binder,
    Object? value
)
```

Parameters

binder	Type: System.Dynamic.SetMemberBinder Provides information about the object that called the dynamic operation. The binder.Name property provides the name of the member to which the value is being assigned. For example, for the statement sampleObject.SampleProperty = "Test", where sampleObject is an instance of the class derived from the DynamicObject class, binder.Name returns "SampleProperty". The binder.IgnoreCase property specifies whether the member name is case-sensitive.
value	Type: System.Object The value to set to the member. For example, for sampleObject.SampleProperty = "Test", where sampleObject is an instance of the class derived from the DynamicObject class, the value is "Test".

Return Value

Type: Boolean
true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a language-specific run-time exception is thrown.)

Reference

[DynamicValue Class](#) [► 2347]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.24.2.19 DynamicValue.TrySetMemberValue Method

Tries to Set a Member/Property Value

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TrySetMemberValue(  
    string name,  
    Object? value  
)
```

Parameters

name	Type: System.String The name of the member
value	Type: System.Object The value.

Return Value

Type: Boolean
true if succeeded, otherwise false otherwise.

Implements

[IStructValue.TrySetMemberValue\(String, Object\)](#) [► 2681]

Reference

[DynamicValue Class](#) [► 2347]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.24.2.20 DynamicValue.Write Method

Writes the value (via ADS)

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Write()
```

Implements

[IValue.Write.](#) [► 2751]

Exceptions

Exception	Condition
SymbolException [▶ 2933]	

Reference

[DynamicValue Class](#) [[▶ 2347](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.24.2.21 DynamicValue.WriteAsync Method

write as an asynchronous operation.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Task<ResultWriteAccess> WriteAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultWriteAccess](#) [[▶ 3217](#)].

Task<WriteValueResult>.

Implements

[IValue.WriteAsync\(CancellationToken\)](#) [[▶ 2751](#)]

Reference

[DynamicValue Class](#) [[▶ 2347](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.24.2.22 DynamicValue.WriteMember Method

Writes the specified member element.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.116+a71ced3

Syntax**C#**

```
protected virtual void WriteMember(
    ISymbol memberInstance,
    Object? value
)
```

Parameters

memberInstance	Type: TwinCAT.TypeSystem.ISymbol [2691] The member instance.
value	Type: System.Object The value.

Reference[DynamicValue Class](#) [[2347](#)][TwinCAT.TypeSystem Namespace](#) [[2083](#)]**6.11.24.2.23 DynamicValue.setMemberValue Method**

Writes the specified member element.

Namespace: [TwinCAT.TypeSystem](#) [[2083](#)]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
protected virtual void setMemberValue(
    ISymbol memberInstance,
    Object? value
)
```

Parameters

memberInstance	Type: TwinCAT.TypeSystem.ISymbol [2691] The member instance.
value	Type: System.Object The value.

Reference[DynamicValue Class](#) [[2347](#)][TwinCAT.TypeSystem Namespace](#) [[2083](#)]**6.11.25 DynamicVirtualStructInstance Class**

Dynamic struct instance

Inheritance Hierarchy

System.Object
 System.Dynamic.DynamicObject
[TwinCAT.TypeSystem.DynamicSymbol](#) [[2266](#)]
[TwinCAT.TypeSystem.DynamicInterfaceInstance](#) [[2995](#)]
[TwinCAT.TypeSystem.DynamicStructInstance](#) [[2251](#)]
 TwinCAT.TypeSystem.DynamicVirtualStructInstance

Namespace: [TwinCAT.TypeSystem](#) [[2083](#)]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229













Syntax














C#

```
public sealed class DynamicVirtualStructInstance : DynamicStructInstance,
    IVirtualStructInstance, IStructInstance, IInterfaceInstance, ISymbol, IAttributedInstance,
    IInstance, IBitSize, IRpcCallableInstance
```









The DynamicVirtualStructInstance type exposes the following members.


















Properties




















	Name	Description
	InnerSymbol [▶ 2274]	Gets the inner symbol of this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266] .)
	AccessRights [▶ 2274]	Gets the access rights. (Inherited from DynamicSymbol [▶ 2266] .)
	Attributes [▶ 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [▶ 2266] .)
	BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from DynamicSymbol [▶ 2266] .)
	ByteSize [▶ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [▶ 2266] .)
	Category [▶ 2277]	Gets the category. (Inherited from DynamicSymbol [▶ 2266] .)
	Comment [▶ 2277]	Gets the comment of the IInstance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266] .)
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266] .)
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549] . (Inherited from DynamicSymbol [▶ 2266] .)
	HasRpcMethods [▶ 3001]	Gets a value indicating whether this instance has RPC methods (Inherited from DynamicInterfaceInstance [▶ 2995] .)
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances, what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266] .)
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266] .)
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266] .)
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266] .)
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DynamicSymbol [▶ 2266] .)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from DynamicSymbol [▶ 2266] .)
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266] .)
	IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266] .)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266] .)

	Name	Description
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266].)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266].)
	IsReference [▶ 2284]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)
	MemberInstances [▶ 3002]	Gets the member instances of the Struct Instance [▶ 2666]. (Inherited from DynamicInterfaceInstance [▶ 2995].)
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
	RpcMethods [▶ 3003]	Gets the Method descriptions for the IRpcCallableType [▶ 2624] (Inherited from DynamicInterfaceInstance [▶ 2995].)
	Size [▶ 2286]	Gets the size of the IInstance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)



Methods

	Name	Description
	AddMember [▶ 2383]	Adds a member instance.
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)
	GetDynamicMemberNames [▶ 3006]	Returns the enumeration of all dynamic member names. (Inherited from DynamicInterfaceInstance [▶ 2995].)
	GetHashCode [▶ 2293]	Gets the HashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
 	InvokeRpcMethodAsync [▶ 3007]	Invokes the specified RPC Method asynchronously (Inherited from DynamicInterfaceInstance [▶ 2995].)

	Name	Description
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue. [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValueAsync [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject .)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryGetIndex	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicObject .)
	TryGetMember [▶ 3008]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Inherited from DynamicInterfaceInstance [▶ 2995].)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject .)
	TryInvokeMember [▶ 3009]	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicInterfaceInstance [▶ 2995].)

	Name	Description
 	TryInvokeRpcMethod(String, Object, Object.) [▶ 3010]	Invokes the specified RPC Method (Inherited from DynamicInterfaceInstance [▶ 2995].)
 	TryInvokeRpcMethod(String, Object, Object, Object.) [▶ 3012]	Invokes the specified RPC Method (Inherited from DynamicInterfaceInstance [▶ 2995].)
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	TrySetIndex	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicObject .)
	TrySetMember [▶ 3013]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicInterfaceInstance [▶ 2995].)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject .)
 	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from DynamicSymbol [▶ 2266].)
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte.) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte, Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)

Events

	Name	Description
	RawValueChanged [▶ 2323]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)
	ValueChanged [▶ 2323]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)














Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.25.1 DynamicVirtualStructInstance Properties

The [DynamicVirtualStructInstance](#) [▶ 2374] type exposes the following members.

Properties

	Name	Description
	InnerSymbol [▶ 2274]	Gets the inner symbol of this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	AccessRights [▶ 2274]	Gets the access rights. (Inherited from DynamicSymbol [▶ 2266].)
	Attributes [▶ 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [▶ 2266].)
	BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from DynamicSymbol [▶ 2266].)
	ByteSize [▶ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [▶ 2266].)
	Category [▶ 2277]	Gets the category. (Inherited from DynamicSymbol [▶ 2266].)
	Comment [▶ 2277]	Gets the comment of the IInstance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266].)
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	HasRpcMethods [▶ 3001]	Gets a value indicating whether this instance has RPC methods (Inherited from DynamicInterfaceInstance [▶ 2995].)
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances, what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266].)
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full <code>DataType</code> but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266].)
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DynamicSymbol [▶ 2266].)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is a container type. (Inherited from DynamicSymbol [▶ 2266].)
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266].)
	IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266].)
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266].)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266].)
	IsReference [▶ 2284]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)
	MemberInstances [▶ 3002]	Gets the member instances of the Struct Instance [▶ 2666]. (Inherited from DynamicInterfaceInstance [▶ 2995].)
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
	RpcMethods [▶ 3003]	Gets the Method descriptions for the IRpcCallableType [▶ 2624] (Inherited from DynamicInterfaceInstance [▶ 2995].)
	Size [▶ 2286]	Gets the size of the IInstance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)

Reference





















[DynamicVirtualStructInstance Class](#) [[▶ 2374](#)]












[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]







6.11.25.2 DynamicVirtualStructInstance Methods

The [DynamicVirtualStructInstance](#) [[▶ 2374](#)] type exposes the following members.

Methods

	Name	Description
	AddMember [▶ 2383]	Adds an member instance.
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)
	GetDynamicMemberNames [▶ 3006]	Returns the enumeration of all dynamic member names. (Inherited from DynamicInterfaceInstance [▶ 2995].)
	GetHashCode [▶ 2293]	Gets the HashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
 	InvokeRpcMethodAsync [▶ 3007]	Invokes the specified RPC Method asynchronously (Inherited from DynamicInterfaceInstance [▶ 2995].)
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValueAsync [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject .)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)

	Name	Description
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetIndex	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicObject.)
	TryGetMember [▶ 3008]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Inherited from DynamicInterfaceInstance [▶ 2995].)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject.)
	TryInvokeMember [▶ 3009]	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicInterfaceInstance [▶ 2995].)
 	TryInvokeRpcMethod(String, Object, Object.) [▶ 3010]	Invokes the specified RPC Method (Inherited from DynamicInterfaceInstance [▶ 2995].)
 	TryInvokeRpcMethod(String, Object, Object, Object.) [▶ 3012]	Invokes the specified RPC Method (Inherited from DynamicInterfaceInstance [▶ 2995].)
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	TrySetIndex	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicObject.)
	TrySetMember [▶ 3013]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicInterfaceInstance [▶ 2995].)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
 	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from DynamicSymbol [▶ 2266].)
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte.) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	WriteRawValue(Byte, Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266] .)
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266] .)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266] . (Inherited from DynamicSymbol [▶ 2266] .)
 	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266] . (Inherited from DynamicSymbol [▶ 2266] .)
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266] .)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021] .)

Reference

[DynamicVirtualStructInstance Class \[▶ 2374\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.25.2.1 DynamicVirtualStructInstance.AddMember Method

Adds an member instance.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool AddMember(
    ISymbol memberInstance,
    IVirtualStructInstance parent
)
```

Parameters

memberInstance	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The member instance.
parent	Type: TwinCAT.TypeSystem.IVirtualStructInstance [▶ 2790] The parent struct instance. Usually the this pointer.

Return Value

Type: Boolean

Implements

[IVirtualStructInstance.AddMember\(ISymbol, IVirtualStructInstance\)](#) [[▶ 2794](#)]

Reference



[DynamicVirtualStructInstance Class](#) [[▶ 2374](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.25.3 DynamicVirtualStructInstance Events

The [DynamicVirtualStructInstance](#) [[▶ 2374](#)] type exposes the following members.

Events

	Name	Description
	RawValueChanged [▶ 2323]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)
	ValueChanged [▶ 2323]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)

Reference

[DynamicVirtualStructInstance Class](#) [[▶ 2374](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.26 EnumValue.T. Class

Enum Value

Inheritance Hierarchy

System.Object

TwinCAT.TypeSystem.EnumValue.T.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public class EnumValue<T> : IEnumValue
where T : IConvertible
```

Type Parameters






T	Enum base type (byte,sbyte,short,ushort,int,uint,long or ulong)
---	---

The EnumValue.T. type exposes the following members.









Constructors

	Name	Description
	EnumValue.T. [▶ 2390]	Initializes a new instance of the EnumValue.T. class.

Properties

	Name	Description
	ManagedBaseType [▶ 2386]	Gets the enumeration base type (sint,byte,short,ushort,int,uint,Int64,UInt64 supported)
	Name [▶ 2386]	Gets the name of the Enum Value
	Primitive [▶ 2387]	Gets the value.
	RawValue [▶ 2387]	Gets the raw value of the enumeration (as byte array)
	Size [▶ 2387]	Gets the size of the Enum value (in bytes)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Parse [▶ 2389]	Parse Enum Type string
	ToString [▶ 2389]	Returns a String that represents this instance. (Overrides Object.ToString..)
	TryParse [▶ 2390]	Parse EnumType string






Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.26.1 EnumValue.T. Properties

The [EnumValue.T.](#) [[▶ 2384](#)] generic type exposes the following members.

Properties

	Name	Description
	ManagedBaseType [▶ 2386]	Gets the enumeration base type (sint,byte,short,ushort,int,uint,Int64,UInt64 supported)
	Name [▶ 2386]	Gets the name of the Enum Value
	Primitive [▶ 2387]	Gets the value.
	RawValue [▶ 2387]	Gets the raw value of the enumeration (as byte array)
	Size [▶ 2387]	Gets the size of the Enum value (in bytes)

Reference

[EnumValue.T. Class \[► 2384\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.26.1.1 EnumValue.T..ManagedBaseType Property

Gets the enumeration base type (sint,byte,short,ushort,int,uint,Int64,UInt64 supported)

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Type ManagedBaseType { get; }
```

Property Value

Type: Type

The type of the base.

Implements

[IEnumValue.ManagedBaseType \[► 2525\]](#)

Reference

[EnumValue.T. Class \[► 2384\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.26.1.2 EnumValue.T..Name Property

Gets the name of the Enum Value

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Name { get; }
```

Property Value

Type: String

The name.

Implements

[IEnumValue.Name \[► 2525\]](#)

Reference

[EnumValue.T. Class \[► 2384\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.26.1.3 EnumValue.T..Primitive Property

Gets the value.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T Primitive { get; }
```

Property Value

Type: [T](#) [[▶ 2384](#)]

The value.

Reference

[EnumValue.T. Class](#) [[▶ 2384](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.26.1.4 EnumValue.T..RawValue Property

Gets the raw value of the enumeration (as byte array)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public byte[] RawValue { get; }
```

Property Value

Type: [.Byte](#).

The raw value.

Implements

[IEnumValue.RawValue](#) [[▶ 2526](#)]

Exceptions

Exception	Condition
NotSupportedException	Base type of enum is not allowed!

Reference

[EnumValue.T. Class](#) [[▶ 2384](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.26.1.5 EnumValue.T..Size Property

Gets the size of the Enum value (in bytes)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Size { get; }
```

Property Value

Type: Int32
The size.

Implements

[IEnumValue.Size \[▸ 2526\]](#)

Reference











[EnumValue.T. Class \[▸ 2384\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.26.2 EnumValue.T. Methods

The [EnumValue.T. \[▸ 2384\]](#) generic type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Parse [▸ 2389]	Parse Enum Type string
		
	ToString [▸ 2389]	Returns a String that represents this instance. (Overrides Object.ToString..)
	TryParse [▸ 2390]	Parse EnumType string
		

Reference

[EnumValue.T. Class \[▸ 2384\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.26.2.1 EnumValue.T..Parse Method

Parse Enum Type string

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static EnumValue<T> Parse(  
    IEnumType<T> type,  
    string str  
)
```

Parameters

type	Type: TwinCAT.TypeSystem.IEnumType [▶ 2516]. T [▶ 2384]. The type.
str	Type: System.String The string.

Return Value

Type: [EnumValue](#) [[▶ 2384](#)].[T](#) [[▶ 2384](#)].

EnumValue<T>.

Exceptions

Exception	Condition
FormatException	

Reference

[EnumValue.T. Class](#) [[▶ 2384](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.26.2.2 EnumValue.T..ToString Method

Returns a String that represents this instance.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string? ToString()
```

Return Value

Type: String

A String that represents this instance.

Reference

[EnumValue.T. Class](#) [[▶ 2384](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.26.2.3 EnumValue.T..TryParse Method

Parse EnumType string

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool TryParse(
    IEnumType<T> type,
    string str,
    out EnumValue<T>? value
)
```

Parameters

type	Type: TwinCAT.TypeSystem.IEnumType [▶ 2516]. T [▶ 2384]. The type.
str	Type: System.String The string.
value	Type: TwinCAT.TypeSystem.EnumValue [▶ 2384]. T [▶ 2384]. The value.

Return Value

Type: Boolean

true if the [EnumValue.T.](#) [[▶ 2384](#)] could be parsed, false otherwise.

Reference

[EnumValue.T. Class](#) [[▶ 2384](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.26.3 EnumValue.T. Constructor

Initializes a new instance of the [EnumValue.T.](#) [[▶ 2384](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public EnumValue(
    string name,
    T value,
    int valueSize
)
```

Parameters

name	Type: System.String The name.
value	Type: T [▶ 2384] The value.
valueSize	Type: System.Int32 Value size in bytes.

Reference

[EnumValue.T. Class \[▸ 2384\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.27 EnumValueCollection Class

Class EnumValueCollection.

Inheritance Hierarchy

System.Object
 TwinCAT.TypeSystem.EnumValueCollection

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229





Syntax

C#









```
public class EnumValueCollection : IList<IEnumValue>,
    ICollection<IEnumValue>, IEnumerable<IEnumValue>, IEnumerable,
    IEnumValueCollection, IEnumValueCollection<IEnumValue, IConvertible>
```

















The EnumValueCollection type exposes the following members.

Properties

	Name	Description
	Count [▸ 2393]	Gets the number of elements contained in the ICollection.T..
	IsReadOnly [▸ 2393]	Gets a value indicating whether the ICollection.T. is read-only.
	Item.Int32. [▸ 2394]	Gets or sets the element at the specified index.
	Item.String. [▸ 2395]	Gets or sets the element at the specified index.

Methods

	Name	Description
	Add [▸ 2396]	Adds an item to the ICollection.T..
	AsReadOnly [▸ 2397]	Gets as read only.
	Clear [▸ 2397]	Removes all items from the ICollection.T..
	Contains(IEnumValue) [▸ 2399]	Determines whether the ICollection.T. contains a specific value.
	Contains(Object) [▸ 2398]	Determines whether [contains] [the specified value].
	Contains(String) [▸ 2399]	Determines whether [contains] [the specified name].
	CopyTo [▸ 2400]	Copies the entire list.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)

	Name	Description
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 2400]	Returns an enumerator that iterates through the collection.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetNames [▶ 2401]	Gets the names.
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	GetValues [▶ 2401]	Gets the values.
	IndexOf [▶ 2402]	Determines the index of a specific item in the IList.T..
	Insert [▶ 2402]	Inserts an item to the IList.T. at the specified index.
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Parse [▶ 2403]	Parses the specified string to the Enum value.
	Remove [▶ 2404]	Removes the first occurrence of a specific object from the ICollection.T..
	RemoveAt [▶ 2404]	Removes the IList.T. item at the specified index.
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInfo [▶ 2405]	Tries the get information.
	TryParse(String, IConvertible.) [▶ 2406]	Parse the specified string to the enum value.
	TryParse(String, IEnumValue.) [▶ 2407]	Parse the specified string to the enum value.





Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.27.1 EnumValueCollection Properties

The [EnumValueCollection \[▶ 2391\]](#) type exposes the following members.

Properties

	Name	Description
	Count [▶ 2393]	Gets the number of elements contained in the ICollection.T..
	IsReadOnly [▶ 2393]	Gets a value indicating whether the ICollection.T. is read-only.
	Item.Int32. [▶ 2394]	Gets or sets the element at the specified index.
	Item.String. [▶ 2395]	Gets or sets the element at the specified index.

Reference

[EnumValueCollection Class \[▶ 2391\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.27.1.1 EnumValueCollection.Count Property

Gets the number of elements contained in the ICollection.T..

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Count { get; }
```

Property Value

Type: Int32

The count.

Implements

ICollection.T..Count

Reference

[EnumValueCollection Class \[► 2391\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.27.1.2 EnumValueCollection.IsReadOnly Property

Gets a value indicating whether the ICollection.T. is read-only.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsReadOnly { get; }
```

Property Value

Type: Boolean

true if this instance is read only; otherwise, false.

Implements

ICollection.T..IsReadOnly



Reference

[EnumValueCollection Class \[► 2391\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.27.1.3 EnumValueCollection.Item Property

Overload List

	Name	Description
	Item.Int32. [▸ 2394]	Gets or sets the element at the specified index.
	Item.String. [▸ 2395]	Gets or sets the element at the specified index.

Reference

[EnumValueCollection Class \[▸ 2391\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.27.1.3.1 EnumValueCollection.Item Property (Int32)

Gets or sets the element at the specified index.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IEnumValue this[
    int index
] { get; set; }
```

Parameters

index	Type: System.Int32 The index.
-------	----------------------------------

Return Value

Type: [IEnumValue \[▸ 2524\]](#)

EnumValue<T>.

Implements

IList.T..Item.Int32.

Exceptions

Exception	Condition
NotImplementedException	

Reference

[EnumValueCollection Class \[▸ 2391\]](#)

[Item Overload \[▸ 2394\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.27.1.3.2 EnumValueCollection.Item Property (String)

Gets or sets the element at the specified index.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IConvertible this[
    string name
] { get; }
```

Parameters

name	Type: System.String The name of the value
------	--

Return Value

Type: IConvertible
EnumValue<T>.

Implements

[IEnumValueCollection.TEnumValue, TValue..Item.String.](#) [[▶ 2530](#)]

Exceptions

Exception	Condition
NotImplementedException	
NotImplementedException	

Reference

[EnumValueCollection Class](#) [[▶ 2391](#)]





[Item Overload](#) [[▶ 2394](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.27.2 EnumValueCollection Methods

The [EnumValueCollection](#) [[▶ 2391](#)] type exposes the following members.

Methods

	Name	Description
	Add [▶ 2396]	Adds an item to the ICollection.T..
	AsReadOnly [▶ 2397]	Gets as read only.
	Clear [▶ 2397]	Removes all items from the ICollection.T..
	Contains(IEnumValu e) [▶ 2399]	Determines whether the ICollection.T. contains a specific value.

	Name	Description
	Contains(Object) [▶ 2398]	Determines whether [contains] [the specified value].
	Contains(String) [▶ 2399]	Determines whether [contains] [the specified name].
	CopyTo [▶ 2400]	Copies the entire list.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 2400]	Returns an enumerator that iterates through the collection.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetNames [▶ 2401]	Gets the names.
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	GetValues [▶ 2401]	Gets the values.
	IndexOf [▶ 2402]	Determines the index of a specific item in the IList.T..
	Insert [▶ 2402]	Inserts an item to the IList.T. at the specified index.
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Parse [▶ 2403]	Parses the specified string to the Enum value.
	Remove [▶ 2404]	Removes the first occurrence of a specific object from the ICollection.T..
	RemoveAt [▶ 2404]	Removes the IList.T. item at the specified index.
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInfo [▶ 2405]	Tries the get information.
	TryParse(String, IConvertible.) [▶ 2406]	Parse the specified string to the enum value.
	TryParse(String, IEnumValue.) [▶ 2407]	Parse the specified string to the enum value.

Reference

[EnumValueCollection Class](#) [▶ 2391]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.27.2.1 EnumValueCollection.Add Method

Adds an item to the ICollection.T..

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Add(  
    IEnumValue item  
)
```

Parameters

item	Type: TwinCAT.TypeSystem.IEnumValue [▸ 2524] The object to add to the ICollection.T..
------	--

Implements

ICollection.T..Add(T)

Reference

[EnumValueCollection Class](#) [▸ 2391]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.27.2.2 EnumValueCollection.AsReadOnly Method

Gets as read only.

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyEnumValueCollection AsReadOnly()
```

Field Value

Type: [ReadOnlyEnumValueCollection](#) [▸ 2824]

As read only.

Return Value

Type: [ReadOnlyEnumValueCollection](#) [▸ 2824]

ReadOnlyEnumValueCollection.

Reference

[EnumValueCollection Class](#) [▸ 2391]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.27.2.3 EnumValueCollection.Clear Method

Removes all items from the ICollection.T..

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Clear()
```

Implements

ICollection.T..Clear.




Reference

[EnumValueCollection Class \[▸ 2391\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.27.2.4 EnumValueCollection.Contains Method

Overload List

	Name	Description
	Contains(IEnumValue) [▸ 2399]	Determines whether the ICollection.T. contains a specific value.
	Contains(Object) [▸ 2398]	Determines whether [contains] [the specified value].
	Contains(String) [▸ 2399]	Determines whether [contains] [the specified name].

Reference

[EnumValueCollection Class \[▸ 2391\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.27.2.4.1 EnumValueCollection.Contains Method (Object)

Determines whether [contains] [the specified value].

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(
    Object value
)
```

Parameters

value	Type: System.Object The value.
-------	-----------------------------------

Return Value

Type: Boolean

true if [contains] [the specified value]; otherwise, false.

Reference

[EnumValueCollection Class \[▸ 2391\]](#)

[Contains Overload \[▸ 2398\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.27.2.4.2 EnumValueCollection.Contains Method (String)

Determines whether [contains] [the specified name].

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(  
    string name  
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: Boolean

true if [contains] [the specified name]; otherwise, false.

Implements

[IEnumValueCollection.TEnumValue, TValue..Contains\(String\) \[▸ 2532\]](#)

Reference

[EnumValueCollection Class \[▸ 2391\]](#)

[Contains Overload \[▸ 2398\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.27.2.4.3 EnumValueCollection.Contains Method (IEnumValue)

Determines whether the ICollection.T. contains a specific value.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(  
    IEnumValue item  
)
```

Parameters

item	Type: TwinCAT.TypeSystem.IEnumValue [▸ 2524] The object to locate in the ICollection.T. .
------	--

Return Value

Type: Boolean

true if item is found in the [ICollection.T.](#); otherwise, false.

Implements

[ICollection.T.](#).Contains(T)

Reference

[EnumValueCollection Class](#) [▸ 2391]

[Contains Overload](#) [▸ 2398]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.27.2.5 EnumValueCollection.CopyTo Method

Copies the entire list.

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public void CopyTo(
    IEnumValue[] array,
    int arrayIndex
)
```

Parameters

array	Type: .TwinCAT.TypeSystem.IEnumValue [▸ 2524]. The array.
arrayIndex	Type: System.Int32 Index of the array.

Implements

[ICollection.T.](#).CopyTo(.T., Int32)

Reference

[EnumValueCollection Class](#) [▸ 2391]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.27.2.6 EnumValueCollection.GetEnumerator Method

Returns an enumerator that iterates through the collection.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IEnumerator<IEnumValue> GetEnumerator()
```

Return Value

Type: [IEnumerator.IEnumValue](#) [► 2524].

A [IEnumerator.T](#). that can be used to iterate through the collection.

Implements

[IEnumerable.T](#)..GetEnumerator.

Reference

[EnumValueCollection Class](#) [► 2391]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.27.2.7 EnumValueCollection.GetNames Method

Gets the names.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string[] GetNames()
```

Return Value

Type: [.String](#).

[System.String](#)[].

Implements

[IEnumValueCollection.TEnumValue, TValue](#)..GetNames. [► 2533]

Reference

[EnumValueCollection Class](#) [► 2391]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.27.2.8 EnumValueCollection.GetValues Method

Gets the values.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IConvertible[] GetValues()
```

Return Value

Type: `IConvertible`.

`T[]`.

Implements

[IEnumerable<TEnumValue, TValue>.GetValues.](#) [[▶ 2533](#)]

Reference

[EnumValueCollection Class](#) [[▶ 2391](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.27.2.9 EnumValueCollection.IndexOf Method

Determines the index of a specific item in the `IList.T`.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int IndexOf(
    IEnumValue item
)
```

Parameters

item	Type: TwinCAT.TypeSystem.IEnumValue [▶ 2524] The object to locate in the <code>IList.T</code> .
------	--

Return Value

Type: `Int32`

The index of item if found in the list; otherwise, -1.

Implements

`IList.T.IndexOf(T)`

Reference

[EnumValueCollection Class](#) [[▶ 2391](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.27.2.10 EnumValueCollection.Insert Method

Inserts an item to the `IList.T` at the specified index.

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Insert(  
    int index,  
    IEnumValue item  
)
```

Parameters

index	Type: System.Int32 The zero-based index at which item should be inserted.
item	Type: TwinCAT.TypeSystem.IEnumValue [▸ 2524] The object to insert into the IList.T..

Implements

IList.T..Insert(Int32, T)

Reference

[EnumValueCollection Class](#) [[▸ 2391](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.27.2.11 EnumValueCollection.Parse Method

Parses the specified string to the Enum value.

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IConvertible Parse(  
    string name  
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: IConvertible

T.

Implements

[IEnumValueCollection.TEnumValue, TValue..Parse\(String\)](#) [[▸ 2533](#)]

Exceptions

Exception	Condition
ArgumentOutOfRangeException	name

Reference

[EnumValueCollection Class \[▸ 2391\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.27.2.12 EnumValueCollection.Remove Method

Removes the first occurrence of a specific object from the ICollection.T..

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public bool Remove(
    IEnumValue item
)
```

Parameters

item	Type: TwinCAT.TypeSystem.IEnumValue [▸ 2524] The object to remove from the ICollection.T..
------	---

Return Value

Type: Boolean

true if item was successfully removed from the ICollection.T.; otherwise, false. This method also returns false if item is not found in the original ICollection.T..

Implements

ICollection.T..Remove(T)

Exceptions

Exception	Condition
NotImplementedException	

Reference

[EnumValueCollection Class \[▸ 2391\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.27.2.13 EnumValueCollection.RemoveAt Method

Removes the IList.T. item at the specified index.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void RemoveAt(  
    int index  
)
```

Parameters

index	Type: System.Int32 The zero-based index of the item to remove.
-------	---

Implements

IList.T..RemoveAt(Int32)

Reference

[EnumValueCollection Class \[▸ 2391\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.27.2.14 EnumValueCollection.TryGetInfo Method

Tries the get information.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetInfo(  
    Object val,  
    out IEnumValue?? ei  
)
```

Parameters

val	Type: System.Object The value.
ei	Type: TwinCAT.TypeSystem.IEnumValue [▸ 2524] . The ei.

Return Value

Type: Boolean

true if the value can be parsed, false otherwise.



Reference

[EnumValueCollection Class \[▸ 2391\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.27.2.15 EnumValueCollection.TryParse Method

Overload List

	Name	Description
	TryParse(String, IConvertible.) [▶ 2406]	Parse the specified string to the enum value.
	TryParse(String, IEnumValue.) [▶ 2407]	Parse the specified string to the enum value.

Reference

[EnumValueCollection Class](#) [▶ 2391]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.27.2.15.1 EnumValueCollection.TryParse Method (String, IConvertible.)

Parse the specified string to the enum value.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryParse (
    string name,
    out IConvertible value
)
```

Parameters

name	Type: System.String The name.
value	Type: System.IConvertible. The value.

Return Value

Type: Boolean

true if the value can be parsed, false otherwise.

Implements

[IEnumValueCollection.TEnumValue, TValue..TryParse\(String, TValue.\)](#) [▶ 2535]

Reference

[EnumValueCollection Class](#) [▶ 2391]

[TryParse Overload](#) [▶ 2406]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.27.2.15.2 EnumValueCollection.TryParse Method (String, IEnumValue.)

Parse the specified string to the enum value.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryParse(  
    string name,  
    out IEnumValue?? value  
)
```

Parameters

name	Type: System.String The name.
value	Type: TwinCAT.TypeSystem.IEnumValue [▶ 2524]. The value.

Return Value

Type: Boolean

true if the value can be parsed, false otherwise.

Implements

[IEnumValueCollection.TEnumValue, TValue..TryParse\(String, TValue.\)](#) [[▶ 2535](#)]

Reference

[EnumValueCollection Class](#) [[▶ 2391](#)]

[TryParse Overload](#) [[▶ 2406](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.28 EnumValueCollection.T. Class

Collection of [EnumValues](#) [[▶ 2384](#)]

Inheritance Hierarchy

System.Object

 TwinCAT.TypeSystem.EnumValueCollection.T.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public class EnumValueCollection<T> : IList<EnumValue<T>>,  
    ICollection<EnumValue<T>>, IEnumerable<EnumValue<T>>, IEnumerable<T>,  
    IEnumValueCollection<EnumValue<T>, T>  
where T : struct, new(), IConvertible
```

Type Parameters






T	Base type of enum
---	-------------------

The EnumValueCollection.T. type exposes the following members.














Constructors













	Name	Description
	EnumValueCollection.T. [▶ 2427]	Initializes a new instance of the EnumValueCollection.T. class.

Properties


	Name	Description
	Count [▶ 2410]	Gets the number of elements contained in the ICollection.T..
	Empty [▶ 2412]	Return an Empty Collection.
	IsReadOnly [▶ 2410]	Gets a value indicating whether the ICollection.T. is read-only.
	Item.Int32. [▶ 2411]	Gets or sets the element at the specified index.
	Item.String. [▶ 2412]	Gets the enumeration value str from the string representation.

Methods

	Name	Description
	Add [▶ 2414]	Adds an item to the ICollection.T..
	AddValue [▶ 2425]	Adds the value.
	AsReadOnly [▶ 2414]	Gets as read only.
	Clear [▶ 2415]	Removes all items from the ICollection.T..
	Contains(EnumValue.T.) [▶ 2417]	Determines whether the ICollection.T. contains a specific value.
	Contains(T) [▶ 2416]	Determines whether [contains] [the specified value].
	Contains(String) [▶ 2416]	Determines whether [contains] [the specified name].
	CopyTo [▶ 2417]	Copies the entire list.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 2418]	Returns an enumerator that iterates through the collection.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetNames [▶ 2419]	Gets the names.

	Name	Description
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	GetValues [▶ 2419]	Gets the values.
	IndexOf [▶ 2420]	Determines the index of a specific item in the IList.T..
	Insert [▶ 2420]	Inserts an item to the IList.T. at the specified index.
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Parse [▶ 2421]	Parses the specified string to the Enum value.
	Remove [▶ 2422]	Removes the first occurrence of a specific object from the ICollection.T..
	RemoveAt [▶ 2422]	Removes the IList.T. item at the specified index.
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetValue [▶ 2425]	Tries to get the value
	TryParse(String, T.) [▶ 2424]	Parse the specified string to the enum value.
	TryParse(String, EnumValue.T..) [▶ 2424]	Parse the specified string to the enum value.

Operators

	Name	Description
	(EnumValueCollection.T. to EnumValueCollection) [▶ 2426]	Performs an explicit conversion from EnumValueCollection.T. to EnumValueCollection [▶ 2391].

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]





Also see about this


-  [EnumValueCollection.T..TryGetInfo Method](#) [[▶ 2423](#)]

6.11.28.1 EnumValueCollection.T. Properties

The [EnumValueCollection.T.](#) [[▶ 2407](#)] generic type exposes the following members.

Properties

	Name	Description
	Count [▶ 2410]	Gets the number of elements contained in the ICollection.T..
	Empty [▶ 2412]	Return an Empty Collection.
	IsReadOnly [▶ 2410]	Gets a value indicating whether the ICollection.T. is read-only.
	Item.Int32. [▶ 2411]	Gets or sets the element at the specified index.

	Name	Description
	Item.String. ▶ 2412	Gets the enumeration value str from the string representation.

Reference

[EnumValueCollection.T. Class](#) [▶ 2407](#)

[TwinCAT.TypeSystem Namespace](#) [▶ 2083](#)

6.11.28.1.1 EnumValueCollection.T..Count Property

Gets the number of elements contained in the ICollection.T..

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Count { get; }
```

Property Value

Type: Int32
The count.

Implements

ICollection.T..Count

Reference

[EnumValueCollection.T. Class](#) [▶ 2407](#)

[TwinCAT.TypeSystem Namespace](#) [▶ 2083](#)

6.11.28.1.2 EnumValueCollection.T..IsReadOnly Property

Gets a value indicating whether the ICollection.T. is read-only.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsReadOnly { get; }
```

Property Value

Type: Boolean
true if this instance is read only; otherwise, false.

Implements

ICollection.T..IsReadOnly



Reference

[EnumValueCollection.T. Class \[▸ 2407\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.28.1.3 EnumValueCollection.T..Item Property

Overload List

	Name	Description
	Item.Int32. [▸ 2411]	Gets or sets the element at the specified index.
	Item.String. [▸ 2412]	Gets the enumeration value str from the string representation.

Reference

[EnumValueCollection.T. Class \[▸ 2407\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.28.1.3.1 EnumValueCollection.T..Item Property (Int32)

Gets or sets the element at the specified index.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public EnumValue<T> this[
    int index
] { get; set; }
```

Parameters

index	Type: System.Int32 The index.
-------	----------------------------------

Return Value

Type: [EnumValue \[▸ 2384\].T \[▸ 2407\]](#).

EnumValue<T>.

Implements

IList.T..Item.Int32.

Exceptions

Exception	Condition
NotImplementedException	

Reference

[EnumValueCollection.T. Class \[▸ 2407\]](#)

[Item Overload \[▸ 2411\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.28.1.3.2 EnumValueCollection.T..Item Property (String)

Gets the enumeration value str from the string representation.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T this[
    string str
] { get; }
```

Parameters

str	Type: System.String The string.
-----	------------------------------------

Return Value

Type: [T \[▸ 2407\]](#)

T.

Implements

[IEnumerable.TEnumValue, TValue..Item.String. \[▸ 2530\]](#)

Exceptions

Exception	Condition
NotImplementedException	

Reference

[EnumValueCollection.T. Class \[▸ 2407\]](#)

[Item Overload \[▸ 2411\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.28.1.4 EnumValueCollection.T..Empty Property

Return an Empty Collection.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static EnumValueCollection<T> Empty { get; }
```

Return Value

Type: [EnumValueCollection](#) [[▶ 2407](#)].T [[▶ 2407](#)].
EnumValueCollection<T>.

Reference




















[EnumValueCollection.T. Class](#) [[▶ 2407](#)]





[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.28.2 EnumValueCollection.T. Methods

The [EnumValueCollection.T.](#) [[▶ 2407](#)] generic type exposes the following members.

Methods

	Name	Description
	Add [▶ 2414]	Adds an item to the ICollection.T..
	AddValue [▶ 2425]	Adds the value.
	AsReadOnly [▶ 2414]	Gets as read only.
	Clear [▶ 2415]	Removes all items from the ICollection.T..
	Contains(EnumValue.T.) [▶ 2417]	Determines whether the ICollection.T. contains a specific value.
	Contains(T) [▶ 2416]	Determines whether [contains] [the specified value].
	Contains(String) [▶ 2416]	Determines whether [contains] [the specified name].
	CopyTo [▶ 2417]	Copies the entire list.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 2418]	Returns an enumerator that iterates through the collection.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetNames [▶ 2419]	Gets the names.
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	GetValues [▶ 2419]	Gets the values.
	IndexOf [▶ 2420]	Determines the index of a specific item in the IList.T..
	Insert [▶ 2420]	Inserts an item to the IList.T. at the specified index.
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Parse [▶ 2421]	Parses the specified string to the Enum value.
	Remove [▶ 2422]	Removes the first occurrence of a specific object from the ICollection.T..
	RemoveAt [▶ 2422]	Removes the IList.T. item at the specified index.

	Name	Description
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetValue [▶ 2425]	Tries to get the value
	TryParse(String, T.) [▶ 2424]	Parse the specified string to the enum value.
	TryParse(String, EnumValue.T.) [▶ 2424]	Parse the specified string to the enum value.

Reference

[EnumValueCollection.T. Class \[\[▶ 2407\]\(#\)\]](#)

[TwinCAT.TypeSystem Namespace \[\[▶ 2083\]\(#\)\]](#)

Also see about this

 [EnumValueCollection.T..TryGetInfo Method \[\[▶ 2423\]\(#\)\]](#)

6.11.28.2.1 EnumValueCollection.T..Add Method

Adds an item to the ICollection.T..

Namespace: [TwinCAT.TypeSystem \[\[▶ 2083\]\(#\)\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Add(
    EnumValue<T> item
)
```

Parameters

item	Type: TwinCAT.TypeSystem.EnumValue [▶ 2384].T [▶ 2407] . The object to add to the ICollection.T..
------	--

Implements

ICollection.T..Add(T)

Reference

[EnumValueCollection.T. Class \[\[▶ 2407\]\(#\)\]](#)

[TwinCAT.TypeSystem Namespace \[\[▶ 2083\]\(#\)\]](#)

6.11.28.2.2 EnumValueCollection.T..AsReadOnly Method

Gets as read only.

Namespace: [TwinCAT.TypeSystem \[\[▶ 2083\]\(#\)\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyEnumValueCollection<T> AsReadOnly()
```

Field Value

Type: [ReadOnlyEnumValueCollection](#) [[▶ 2833](#)].T [[▶ 2407](#)].
As read only.

Return Value

Type: [ReadOnlyEnumValueCollection](#) [[▶ 2833](#)].T [[▶ 2407](#)].
ReadOnlyEnumValueCollection<T>.

Reference

[EnumValueCollection.T. Class](#) [[▶ 2407](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.28.2.3 EnumValueCollection.T..Clear Method

Removes all items from the ICollection.T..

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Clear()
```

Implements

ICollection.T..Clear.




Reference

[EnumValueCollection.T. Class](#) [[▶ 2407](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.28.2.4 EnumValueCollection.T..Contains Method

Overload List

	Name	Description
	Contains(EnumValue.T.) [▶ 2417]	Determines whether the ICollection.T. contains a specific value.
	Contains(T) [▶ 2416]	Determines whether [contains] [the specified value].
	Contains(String) [▶ 2416]	Determines whether [contains] [the specified name].

Reference[EnumValueCollection.T. Class \[► 2407\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.28.2.4.1 EnumValueCollection.T..Contains Method (String)**

Determines whether [contains] [the specified name].

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public bool Contains(
    string name
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: Boolean

true if [contains] [the specified name]; otherwise, false.

Implements[IEnumValueCollection.TEnumValue, TValue..Contains\(String\) \[► 2532\]](#)**Reference**[EnumValueCollection.T. Class \[► 2407\]](#)[Contains Overload \[► 2415\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.28.2.4.2 EnumValueCollection.T..Contains Method (T)**

Determines whether [contains] [the specified value].

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public bool Contains(
    T value
)
```

Parameters

value	Type: T [► 2407] The value.
-------	--

Return Value

Type: Boolean
true if [contains] [the specified value]; otherwise, false.

Reference

[EnumValueCollection.T. Class \[▸ 2407\]](#)

[Contains Overload \[▸ 2415\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.28.2.4.3 EnumValueCollection.T.Contains Method (EnumValue.T.)

Determines whether the ICollection.T. contains a specific value.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(  
    EnumValue<T> item  
)
```

Parameters

item	Type: TwinCAT.TypeSystem.EnumValue [▸ 2384].T [▸ 2407] . The object to locate in the ICollection.T..
------	---

Return Value

Type: Boolean
true if item is found in the ICollection.T.; otherwise, false.

Implements

ICollection.T..Contains(T)

Reference

[EnumValueCollection.T. Class \[▸ 2407\]](#)

[Contains Overload \[▸ 2415\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.28.2.5 EnumValueCollection.T.CopyTo Method

Copies the entire list.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void CopyTo(
    EnumValue<T>[] array,
    int arrayIndex
)
```

Parameters

array	Type: .TwinCAT.TypeSystem.EnumValue [▸ 2384].T [▸ 2407] .. The array.
arrayIndex	Type: System.Int32 Index of the array.

Implements

[ICollection.T..CopyTo\(.T., Int32\)](#)

Reference

[EnumValueCollection.T. Class \[▸ 2407\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.28.2.6 EnumValueCollection.T..Empty Method

Return an Empty Collection.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934 addedca3e72bc0ea15da1c14

Syntax

C#

```
public static EnumValueCollection<T> Empty()
```

Return Value

Type: [EnumValueCollection \[▸ 2407\].T \[▸ 2407\]](#).

EnumValueCollection<T>.

Reference

[EnumValueCollection.T. Class \[▸ 2407\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.28.2.7 EnumValueCollection.T..GetEnumerator Method

Returns an enumerator that iterates through the collection.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IEnumerator<EnumValue<T>> GetEnumerator()
```

Return Value

Type: [IEnumerator.EnumValue \[▸ 2384\].T \[▸ 2407\]](#)..
A [IEnumerator.T](#) that can be used to iterate through the collection.

Implements

[IEnumerable.T](#)..GetEnumerator.

Reference

[EnumValueCollection.T](#) Class [▸ 2407]

[TwinCAT.TypeSystem](#) Namespace [▸ 2083]

6.11.28.2.8 EnumValueCollection.T.GetNames Method

Gets the names.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string[] GetNames()
```

Return Value

Type: [.String](#).
[System.String\[\]](#).

Implements

[IEnumValueCollection.TEnumValue, TValue](#)..GetNames. [▸ 2533]

Reference

[EnumValueCollection.T](#) Class [▸ 2407]

[TwinCAT.TypeSystem](#) Namespace [▸ 2083]

6.11.28.2.9 EnumValueCollection.T.GetValues Method

Gets the values.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T[] GetValues()
```

Return Value

Type: [T](#) [[▶ 2407](#)].
[T\[\]](#).

Implements

[IEnumerable.TEnumValue, TValue..GetValues.](#) [[▶ 2533](#)]

Reference

[EnumValueCollection.T. Class](#) [[▶ 2407](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.28.2.10 EnumValueCollection.T..IndexOf Method

Determines the index of a specific item in the [IList.T..](#)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public int IndexOf(
    EnumValue<T> item
)
```

Parameters

item	Type: TwinCAT.TypeSystem.EnumValue [▶ 2384]. T [▶ 2407]. The object to locate in the IList.T..
------	---

Return Value

Type: [Int32](#)
The index of item if found in the list; otherwise, -1.

Implements

[IList.T..IndexOf\(T\)](#)

Reference

[EnumValueCollection.T. Class](#) [[▶ 2407](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.28.2.11 EnumValueCollection.T..Insert Method

Inserts an item to the [IList.T.](#) at the specified index.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Insert(
    int index,
    EnumValue<T> item
)
```

Parameters

index	Type: System.Int32 The zero-based index at which item should be inserted.
item	Type: TwinCAT.TypeSystem.EnumValue [▶ 2384]. T [▶ 2407]. The object to insert into the IList.T..

Implements

[IList.T..Insert\(Int32, T\)](#)

Reference

[EnumValueCollection.T. Class](#) [[▶ 2407](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.28.2.12 EnumValueCollection.T..Parse Method

Parses the specified string to the Enum value.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T Parse(
    string name
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: [T](#) [[▶ 2407](#)]
T.

Implements

[IEnumValueCollection.TEnumValue, TValue..Parse\(String\)](#) [[▶ 2533](#)]

Exceptions

Exception	Condition
ArgumentOutOfRangeException	name

Reference[EnumValueCollection.T. Class \[► 2407\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.28.2.13 EnumValueCollection.T..Remove Method**

Removes the first occurrence of a specific object from the ICollection.T..

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public bool Remove (
    EnumValue<T> item
)
```

Parameters

item	Type: TwinCAT.TypeSystem.EnumValue [► 2384].T [► 2407] . The object to remove from the ICollection.T..
------	---

Return Value

Type: Boolean

true if item was successfully removed from the ICollection.T.; otherwise, false. This method also returns false if item is not found in the original ICollection.T..

Implements

ICollection.T..Remove(T)

Exceptions

Exception	Condition
NotImplementedException	

Reference[EnumValueCollection.T. Class \[► 2407\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.28.2.14 EnumValueCollection.T..RemoveAt Method**

Removes the IList.T. item at the specified index.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public void RemoveAt (
    int index
)
```

Parameters

index	Type: System.Int32 The zero-based index of the item to remove.
-------	---

Implements

IList.T..RemoveAt(Int32)

Reference

[EnumValueCollection.T. Class \[▸ 2407\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.28.2.15 EnumValueCollection.T..TryGetInfo Method

Tries the get information.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public bool TryGetInfo(
    T val,
    out EnumValue<T>? ei
)
```

Parameters

val	Type: T [▸ 2407] The value.
ei	Type: TwinCAT.TypeSystem.EnumValue [▸ 2384].T [▸ 2407] .. The ei.

Return Value

Type: Boolean
true if XXXX, false otherwise.



Reference

[EnumValueCollection.T. Class \[▸ 2407\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.28.2.16 EnumValueCollection.T..TryParse Method

Overload List

	Name	Description
	TryParse(String, T.) [▸ 2424]	Parse the specified string to the enum value.
	TryParse(String, EnumValue.T..) [▸ 2424]	Parse the specified string to the enum value.

Reference

[EnumValueCollection.T. Class \[▸ 2407\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.28.2.16.1 EnumValueCollection.T..TryParse Method (String, EnumValue.T..)

Parse the specified string to the enum value.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryParse(  
    string name,  
    out EnumValue<T>? value  
)
```

Parameters

name	Type: System.String The name.
value	Type: TwinCAT.TypeSystem.EnumValue [▸ 2384].T [▸ 2407] .. The value.

Return Value

Type: Boolean
true if parsed, false otherwise.

Implements

[IEnumValueCollection.TEnumValue, TValue..TryParse\(String, TValue.\) \[▸ 2535\]](#)

Reference

[EnumValueCollection.T. Class \[▸ 2407\]](#)

[TryParse Overload \[▸ 2423\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.28.2.16.2 EnumValueCollection.T..TryParse Method (String, T.)

Parse the specified string to the enum value.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryParse(  
    string name,  
    out T value  
)
```


Parameters

name	Type: System.String The name.
value	Type: T [▶ 2407]. The value.

Return Value

Type: Boolean
true if parsed, false otherwise.

Implements

[IEnumValueCollection.TEnumValue, TValue..TryParse\(String, TValue.\)](#) [[▶ 2535](#)]

Reference

[EnumValueCollection.T. Class](#) [[▶ 2407](#)]

[TryParse Overload](#) [[▶ 2423](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.28.2.17 EnumValueCollection.T..AddValue Method

Adds the value.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public EnumValueCollection<T> AddValue(
    string name,
    T value
)
```

Parameters

name	Type: System.String The name.
value	Type: T [▶ 2407]. The value.

Return Value

Type: [EnumValueCollection](#) [[▶ 2407](#)].T [[▶ 2407](#)].
[EnumValueCollection<T>](#).

Reference

[EnumValueCollection.T. Class](#) [[▶ 2407](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.28.2.18 EnumValueCollection.T..TryGetValue Method

Tries to get the value

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetValue(
    T val,
    out EnumValue<T>? ei
)
```

Parameters

val	Type: T [▶ 2407] The value.
ei	Type: TwinCAT.TypeSystem.EnumValue [▶ 2384]. T [▶ 2407]. The ei.

Return Value

Type: Boolean

true if the value is found, false otherwise.

Reference



[EnumValueCollection.T. Class](#) [[▶ 2407](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.28.3 EnumValueCollection.T. Type Conversions

The [EnumValueCollection.T.](#) [[▶ 2407](#)] generic type exposes the following members.

Operators

	Name	Description
 	. (EnumValueCollection.T. to EnumValueCollection) [▶ 2426]	Performs an explicit conversion from EnumValueCollection.T. [▶ 2407] to EnumValueCollection [▶ 2391].

Reference

[EnumValueCollection.T. Class](#) [[▶ 2407](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.28.3.1 EnumValueCollection.T. . Conversion (EnumValueCollection.T. to EnumValueCollection)

Performs an explicit conversion from [EnumValueCollection.T.](#) [[▶ 2407](#)] to [EnumValueCollection](#) [[▶ 2391](#)].

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static explicit operator EnumValueCollection (  
    EnumValueCollection<T> coll  
)
```

Parameters

coll	Type: TwinCAT.TypeSystem.EnumValueCollection [▸ 2407].T [▸ 2407] . The coll.
------	---

Return Value

Type: [EnumValueCollection \[▸ 2391\]](#)
The result of the conversion.

Reference

[EnumValueCollection.T. Class \[▸ 2407\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.28.4 EnumValueCollection.T. Constructor

Initializes a new instance of the [EnumValueCollection.T. \[▸ 2407\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public EnumValueCollection()
```

Reference

[EnumValueCollection.T. Class \[▸ 2407\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.29 FieldCollection Class

Collection of [IField \[▸ 2535\]](#) objects.

Inheritance Hierarchy

System.Object

[TwinCAT.TypeSystem.Generic.InstanceCollection \[▸ 3104\].IField \[▸ 2535\]](#).

[TwinCAT.TypeSystem.FieldCollection](#)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229



Syntax

C#








```
public class FieldCollection : InstanceCollection<IField>,  
    IFieldCollection, IInstanceCollection<IField>, IList<IField>,  
    ICollection<IField>, IEnumerable<IField>, IEnumerable
```

The FieldCollection type exposes the following members.











Constructors
















	Name	Description
	FieldCollection. [▶ 2430]	Initializes a new instance of the FieldCollection class.
	FieldCollection(IEnumerable.IField.) [▶ 2430]	Initializes a new instance of the FieldCollection class (copy constructor)

Properties

	Name	Description
	Count [▶ 3107]	Gets the collection count. (Inherited from InstanceCollection.T. [▶ 3104].)
	InnerList [▶ 3108]	Gets the List of instances. (Inherited from InstanceCollection.T. [▶ 3104].)
	InnerPathDict [▶ 3108]	The Path dictionary (Inherited from InstanceCollection.T. [▶ 3104].)
	IsReadOnly [▶ 3109]	Gets a value indicating whether this instance is read only. (Inherited from InstanceCollection.T. [▶ 3104].)
	Item.Int32. [▶ 3109]	Gets or sets the Instance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	Item.String. [▶ 3110]	Gets the Instance [▶ 2549] with the specified instance path. (Inherited from InstanceCollection.T. [▶ 3104].)
	Mode [▶ 3111]	The mode this InstanceCollection.T. [▶ 3104] is working in. (Inherited from InstanceCollection.T. [▶ 3104].)

Methods

	Name	Description
	Add [▶ 3112]	Adds the specified item. (Inherited from InstanceCollection.T. [▶ 3104].)
	AddRange [▶ 3113]	Adds the specified items to this collection. (Inherited from InstanceCollection.T. [▶ 3104].)
	AsReadOnly [▶ 2432]	Returns a read only copy of this collection (shallow copy)
	Clear [▶ 3114]	Clears this instance. (Inherited from InstanceCollection.T. [▶ 3104].)
	Clone [▶ 2432]	Clones this FieldCollection.
	Contains(T) [▶ 3115]	Determines whether this collection contains the specified Instance [▶ 2549] (Inherited from InstanceCollection.T. [▶ 3104].)
	Contains(String) [▶ 3114]	Determines whether this collection contains an Instance [▶ 2549] with the specified InstanceName / InstancePath (Inherited from InstanceCollection.T. [▶ 3104].)
	ContainsName [▶ 3116]	Determines whether the specified instance name contains name. (Inherited from InstanceCollection.T. [▶ 3104].)
	CopyTo [▶ 3116]	Copies this InstanceCollection.T. [▶ 3104] to the specified array. (Inherited from InstanceCollection.T. [▶ 3104].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)



	Name	Description
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 3117]	Gets the enumerator. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetInstance [▶ 3117]	Gets the Instance [▶ 2549] by instance path. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetInstanceByName [▶ 3118]	Gets the name of the instance by. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 3119]	Determines the index of the specified Instance [▶ 2549]. (Inherited from InstanceCollection.T. [▶ 3104].)
	Insert [▶ 3119]	Inserts the specified Instance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 3120]	Removes the specified item. (Inherited from InstanceCollection.T. [▶ 3104].)
	RemoveAt [▶ 3121]	Removes the Instance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInstance [▶ 3121]	Tries to get the Instance [▶ 2549]. of the specified path. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetInstanceByName [▶ 3122]	Tries to get Instances by name. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetMember [▶ 2433]	Tries to get the specified member

Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.29.1 FieldCollection Constructor

Overload List

	Name	Description
	FieldCollection. [▶ 2430]	Initializes a new instance of the FieldCollection [▶ 2427] class.
	FieldCollection(IEnumerable.IField.) [▶ 2430]	Initializes a new instance of the FieldCollection [▶ 2427] class (copy constructor)

Reference

[FieldCollection Class](#) [▶ 2427]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.29.1.1 FieldCollection Constructor

Initializes a new instance of the [FieldCollection](#) [▶ 2427] class.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public FieldCollection()
```

Reference

[FieldCollection Class](#) [▶ 2427]

[FieldCollection Overload](#) [▶ 2429]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.29.1.2 FieldCollection Constructor (IEnumerable.IField.)

Initializes a new instance of the [FieldCollection](#) [▶ 2427] class (copy constructor)

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public FieldCollection(
    IEnumerable<IField> coll
)
```

Parameters

coll	Type: System.Collections.Generic.IEnumerable.IField [▶ 2535]. The coll.
------	--

Reference

[FieldCollection Class](#) [▶ 2427]



[FieldCollection Overload](#) [▶ 2429]






[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.29.2 FieldCollection Properties

The [FieldCollection](#) [▶ 2427] type exposes the following members.

Properties

	Name	Description
	Count [▶ 3107]	Gets the collection count. (Inherited from InstanceCollection.T. [▶ 3104].)
	InnerList [▶ 3108]	Gets the Llist of instances. (Inherited from InstanceCollection.T. [▶ 3104].)

	Name	Description
	InnerPathDict [▶ 3108]	The Path dictionary (Inherited from InstanceCollection.T. [▶ 3104].)
	IsReadOnly [▶ 3109]	Gets a value indicating whether this instance is read only. (Inherited from InstanceCollection.T. [▶ 3104].)
	Item.Int32. [▶ 3109]	Gets or sets the Instance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	Item.String. [▶ 3110]	Gets the Instance [▶ 2549] with the specified instance path. (Inherited from InstanceCollection.T. [▶ 3104].)
	Mode [▶ 3111]	The mode this InstanceCollection.T. [▶ 3104] is working in. (Inherited from InstanceCollection.T. [▶ 3104].)

Reference















[FieldCollection Class \[▶ 2427\]](#)












[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.29.3 FieldCollection Methods

The [FieldCollection \[▶ 2427\]](#) type exposes the following members.

Methods

	Name	Description
	Add [▶ 3112]	Adds the specified item. (Inherited from InstanceCollection.T. [▶ 3104].)
	AddRange [▶ 3113]	Adds the specified items to this collection. (Inherited from InstanceCollection.T. [▶ 3104].)
	AsReadOnly [▶ 2432]	Returns a read only copy of this collection (shallow copy)
	Clear [▶ 3114]	Clears this instance. (Inherited from InstanceCollection.T. [▶ 3104].)
	Clone [▶ 2432]	Clones this FieldCollection [▶ 2427] .
	Contains(T) [▶ 3115]	Determines whether this collection contains the specified Instance [▶ 2549] (Inherited from InstanceCollection.T. [▶ 3104].)
	Contains(String) [▶ 3114]	Determines whether this collection contains an Instance [▶ 2549] with the specified InstanceName / InstancePath (Inherited from InstanceCollection.T. [▶ 3104].)
	ContainsName [▶ 3116]	Determines whether the specified instance name contains name. (Inherited from InstanceCollection.T. [▶ 3104].)
	CopyTo [▶ 3116]	Copies this InstanceCollection.T. [▶ 3104] to the specified array. (Inherited from InstanceCollection.T. [▶ 3104].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 3117]	Gets the enumerator. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetInstance [▶ 3117]	Gets the Instance [▶ 2549] by instance path. (Inherited from InstanceCollection.T. [▶ 3104].)

	Name	Description
	GetInstanceByName [▶ 3118]	Gets the name of the instance by. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 3119]	Determines the index of the specified IInstance [▶ 2549]. (Inherited from InstanceCollection.T. [▶ 3104].)
	Insert [▶ 3119]	Inserts the specified IInstance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 3120]	Removes the specified item. (Inherited from InstanceCollection.T. [▶ 3104].)
	RemoveAt [▶ 3121]	Removes the IInstance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInstance [▶ 3121]	Tries to get the IInstance [▶ 2549]. of the specified path. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetInstanceByName [▶ 3122]	Tries to get Instances by name. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetMember [▶ 2433]	Tries to get the specified member

Reference

[FieldCollection Class](#) [[▶ 2427](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.29.3.1 FieldCollection.AsReadOnly Method

Returns a read only copy of this collection (shallow copy)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyFieldCollection AsReadOnly()
```

Return Value

Type: [ReadOnlyFieldCollection](#) [[▶ 2841](#)]

The readonly copy.

Reference

[FieldCollection Class](#) [[▶ 2427](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.29.3.2 FieldCollection.Clone Method

Clones this [FieldCollection](#) [[▶ 2427](#)].

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public FieldCollection Clone()
```

Return Value

Type: [FieldCollection](#) [▸ 2427]

A cloned [FieldCollection](#) [▸ 2427].

Reference

[FieldCollection Class](#) [▸ 2427]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.29.3.3 FieldCollection.TryGetMember Method

Tries to get the specified member

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetMember(  
    string fieldName,  
    out IField?? symbol  
)
```

Parameters

fieldName	Type: System.String Name of the member.
symbol	Type: TwinCAT.TypeSystem.IField [▸ 2535]. The symbol.

Return Value

Type: Boolean

true if found, false otherwise.

Implements

[IFieldCollection.TryGetMember\(String, IField.\)](#) [▸ 2541]

Reference

[FieldCollection Class](#) [▸ 2427]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.30 IAliasInstance Interface

Interface representing an instance of an [IAliasType](#) [▸ 2436].

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229








Syntax





C#

```
public interface IAliasInstance : ISymbol,
    IAttributedInstance, IInstance, IBitSize
```

The IAliasInstance type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)

	Name	Description
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472] .)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691] .)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549] . (Inherited from Instance [▶ 2549] .)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from AttributedInstance [▶ 2469] .)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021] .)

Reference












[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)













[TwinCAT.TypeSystem.ISymbol \[▶ 2691\]](#)

6.11.30.1 IAliasInstance Properties

The [IAliasInstance \[▶ 2433\]](#) type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from AttributedInstance [▶ 2469] .)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472] .)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472] .)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691] .)
	Comment [▶ 2550]	Gets the comment of the Instance [▶ 2549] (Inherited from Instance [▶ 2549] .)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549] . (Inherited from Instance [▶ 2549] .)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from Instance [▶ 2549] .)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from Instance [▶ 2549] .)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472] .)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($BitSize \% 8 == 0$) (Inherited from IBitSize [▶ 2472] .)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691] .)

	Name	Description
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Reference

[IAliasInstance Interface](#) [[▶ 2433](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.30.2 IAliasInstance Methods

The [IAliasInstance](#) [[▶ 2433](#)] type exposes the following members.

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[IAliasInstance Interface](#) [[▶ 2433](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.31 IAliasType Interface

Interface representing an Alias Type

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229















Syntax

C#

```
public interface IAliasType : IDataTypeInfo,
    IBitSize
```

The IAliasType type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataTypeInfo [▶ 2475] (Inherited from IDataTypeInfo [▶ 2475].)
	BaseType [▶ 2439]	Gets the Base Type
	BaseTypeName [▶ 2440]	Gets the BaseType name
	BitSize [▶ 2473]	Gets the size of the IDataTypeInfo [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataTypeInfo [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataTypeInfo [▶ 2475].)
	FullName [▶ 2479]	Gets the full name of the IDataTypeInfo [▶ 2475] (Namespace + Name) (Inherited from IDataTypeInfo [▶ 2475].)
	Id [▶ 2480]	Gets the ID of the DataType (Inherited from IDataTypeInfo [▶ 2475].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from IBitSize [▶ 2472].)
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataTypeInfo [▶ 2475].)
	Namespace [▶ 2482]	Gets the namespace string within the IDataTypeInfo [▶ 2475] exists. (Inherited from IDataTypeInfo [▶ 2475].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Extension Methods

	Name	Description
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer [▶ 3027]	Overloaded.

	Name	Description
		Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive. [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve. [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]















Also see about this

- [IDataType.IsContainer](#) Property [[▶ 2480](#)]
- [IDataType.IsPointer](#) Property [[▶ 2481](#)]
- [IDataType.IsPrimitive](#) Property [[▶ 2481](#)]
- [IDataType.IsReference](#) Property [[▶ 2482](#)]

6.11.31.1 IAliasType Properties

The [IAliasType](#) [[▶ 2436](#)] type exposes the following members.

Properties





	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataType [▶ 2475] (Inherited from IDataType [▶ 2475].)
	BaseType [▶ 2439]	Gets the Base Type
	BaseTypeName [▶ 2440]	Gets the BaseType name
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)
	FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from IDataType [▶ 2475].)
	Id [▶ 2480]	Gets the ID of the Data Type (Inherited from IDataType [▶ 2475].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full Data Type but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475].)
	Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Reference

[IAliasType Interface](#) [[▶ 2436](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

Also see about this

-  [IDataType.IsContainer Property](#) [[▶ 2480](#)]
-  [IDataType.IsPointer Property](#) [[▶ 2481](#)]
-  [IDataType.IsPrimitive Property](#) [[▶ 2481](#)]
-  [IDataType.IsReference Property](#) [[▶ 2482](#)]

6.11.31.1.1 IAliasType.BaseType Property

Gets the Base Type

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDataType? BaseType { get; }
```

Property Value

Type: [IDataType](#) [[▸ 2475](#)]

The alias base type or **null** if not resolved.

Reference

[IAliasType Interface](#) [[▸ 2436](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.31.1.2 IAliasType.BaseTypeName Property

Gets the BaseType name

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string BaseTypeName { get; }
```

Property Value

Type: String

Reference

[IAliasType Interface](#) [[▸ 2436](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.31.2 IAliasType Methods

The [IAliasType](#) [[▸ 2436](#)] type exposes the following members.

Extension Methods

	Name	Description
	IsArrayOfPrimitives [▸ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▸ 3021].)
	IsArrayOfPrimitives(Boolean) [▸ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▸ 3021].)
	IsContainer [▸ 3027]	Overloaded. Gets a value indicating whether this IDataType [▸ 2475] is a container type (Defined by DataTypeExtension [▸ 3021].)
	IsContainer(Boolean) [▸ 3028]	Overloaded. Gets a value indicating whether this IDataType [▸ 2475] is a container type (Defined by DataTypeExtension [▸ 3021].)

	Name	Description
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive. [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve. [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[IAliasType Interface](#) [[▶ 2436](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.32 IAnyTypeMarshaler Interface

Interface [IAnyTypeMarshaler](#) Implements the [IGenericTypeMarshaler](#) [[▶ 2542](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]


Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public interface IAnyTypeMarshaler : IGenericTypeMarshaler,
    ITypeMarshaler
```

Methods

	Name	Description
	CanMarshal(Type) [▶ 2732]	Determines whether ADS can marshal the specified managed data type. (Inherited from ITypeMarshaler [▶ 2731].)

	Name	Description
	CanMarshal(Type, .Int32, Encoding) [▶ 2446]	Determines whether ADS can marshal the specified managed data type.
	CanMarshalValue(Object) [▶ 2736]	Determines whether ADS can marshal the specified value (Inherited from ITypeMarshaler [▶ 2731].)
	CanMarshalValue(Object, .Int32, Encoding) [▶ 2451]	Determines whether ADS can marshal the specified managed data type.
	Marshal(Object, Encoding, Span.Byte.) [▶ 2734]	Marshals the specified value to the specified destination memory / span. (Inherited from ITypeMarshaler [▶ 2731].)
	Marshal(Object, .Int32, Encoding, Span.Byte.) [▶ 2447]	Marshals the specified value to the specified destination.
	MarshalValueSize(Object, Encoding) [▶ 2736]	Gets the byte size of the value when marshalled. (Inherited from ITypeMarshaler [▶ 2731].)
	MarshalValueSize(Object, .Int32, Encoding) [▶ 2452]	Gets the byte size of the value when marshalled.
	Unmarshal(Type, ReadOnlySpan.Byte, Encoding, Object.) [▶ 2735]	Unmarshals the specified managed type from memory / span (Inherited from ITypeMarshaler [▶ 2731].)
	Unmarshal(Type, .Int32, ReadOnlySpan.Byte, Encoding, Object.) [▶ 2450]	Unmarshals the source data to an managed value of the specified type.
	Unmarshal.T.(ReadOnlySpan.Byte, Encoding, T.) [▶ 2544]	Unmarshals the source data to an managed value of the specified type. (Inherited from IGenericTypeMarshaler [▶ 2542].)

Remarks







The [IAnyTypeMarshaler](#) supports value marshalling / Unmarshalling with the ADS ANY_TYPE concept, what means that the specified managed type is supported by an arguments metadata array (args parameter).

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]











[TwinCAT.TypeSystem.IGenericTypeMarshaler](#) [[▶ 2542](#)]


Also see about this

-  [ITypeMarshaler.CanMarshal Method \(Object\) \[▶ 2733\]](#)
-  [ITypeMarshaler.CanMarshal Method \(Type\) \[▶ 2733\]](#)
-  [IAnyTypeMarshaler.CanMarshal Method \(Object, .Int32.\) \[▶ 2444\]](#)
-  [IAnyTypeMarshaler.CanMarshal Method \(Type, .Int32.\) \[▶ 2445\]](#)
-  [ITypeMarshaler.MarshalSize Method \[▶ 2735\]](#)
-  [IAnyTypeMarshaler.MarshalSize Method \(Object, .Int32., Encoding\) \[▶ 2448\]](#)

6.11.32.1 IAnyTypeMarshaler Methods

Methods

	Name	Description
	CanMarshal(Type) [▶ 2732]	Determines whether ADS can marshal the specified managed data type. (Inherited from ITypeMarshaler [▶ 2731].)
	CanMarshal(Type, .Int32., Encoding) [▶ 2446]	Determines whether ADS can marshal the specified managed data type.
	CanMarshalValue(Object) [▶ 2736]	Determines whether ADS can marshal the specified value (Inherited from ITypeMarshaler [▶ 2731].)
	CanMarshalValue(Object, .Int32., Encoding) [▶ 2451]	Determines whether ADS can marshal the specified managed data type.
	Marshal(Object, Encoding, Span.Byte.) [▶ 2734]	Marshals the specified value to the specified destination memory / span. (Inherited from ITypeMarshaler [▶ 2731].)
	Marshal(Object, .Int32., Encoding, Span.Byte.) [▶ 2447]	Marshals the specified value to the specified destination.
	MarshalValueSize(Object, Encoding) [▶ 2736]	Gets the byte size of the value when marshalled. (Inherited from ITypeMarshaler [▶ 2731].)
	MarshalValueSize(Object, .Int32., Encoding) [▶ 2452]	Gets the byte size of the value when marshalled.
	Unmarshal(Type, ReadOnlySpan.Byte, Encoding, Object.) [▶ 2735]	Unmarshals the specified managed type from memory / span (Inherited from ITypeMarshaler [▶ 2731].)
	Unmarshal(Type, .Int32., ReadOnlySpan.Byte, Encoding, Object.) [▶ 2450]	Unmarshals the source data to an managed value of the specified type.







	Name	Description
	Unmarshal.T. (ReadOnlySpan.Byte , Encoding, T.) [▶ 2544]	Unmarshals the source data to an managed value of the specified type. (Inherited from IGenericTypeMarshaler [▶ 2542].)

Reference

[IAnyTypeMarshaler Interface \[▶ 2441\]](#)



[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

Also see about this

-  [ITypeMarshaler.CanMarshal Method \(Object\) \[▶ 2733\]](#)
-  [ITypeMarshaler.CanMarshal Method \(Type\) \[▶ 2733\]](#)
-  [IAnyTypeMarshaler.CanMarshal Method \(Object, .Int32.\) \[▶ 2444\]](#)
-  [IAnyTypeMarshaler.CanMarshal Method \(Type, .Int32.\) \[▶ 2445\]](#)
-  [ITypeMarshaler.MarshalSize Method \[▶ 2735\]](#)
-  [IAnyTypeMarshaler.MarshalSize Method \(Object, .Int32., Encoding\) \[▶ 2448\]](#)

6.11.32.1.1 IAnyTypeMarshaler.CanMarshal Method

Overload List





	Name	Description
	CanMarshal(Type) [▶ 2732]	Determines whether ADS can marshal the specified managed data type. (Inherited from ITypeMarshaler [▶ 2731].)
	CanMarshal(Type, .Int32., Encoding) [▶ 2446]	Determines whether ADS can marshal the specified managed data type.

Reference

[IAnyTypeMarshaler Interface \[▶ 2441\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

Also see about this

-  [ITypeMarshaler.CanMarshal Method \(Object\) \[▶ 2733\]](#)
-  [ITypeMarshaler.CanMarshal Method \(Type\) \[▶ 2733\]](#)
-  [IAnyTypeMarshaler.CanMarshal Method \(Object, .Int32.\) \[▶ 2444\]](#)
-  [IAnyTypeMarshaler.CanMarshal Method \(Type, .Int32.\) \[▶ 2445\]](#)

6.11.32.1.1.1 IAnyTypeMarshaler.CanMarshal Method (Object, .Int32.)

Determines whether ADS can marshal the specified managed data type.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
bool CanMarshal(  
    Object value,  
    int[] args  
)
```

Parameters

value	Type: System.Object The managed value.
args	Type: .System.Int32 . The arguments.

Return Value

Type: [Boolean](#)
true if this instance can marshal the specified managed type; otherwise, false.

Reference

[IAnyTypeMarshaler Interface](#) [[▶ 2441](#)]

[CanMarshal Overload](#) [[▶ 2444](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.32.1.1.2 IAnyTypeMarshaler.CanMarshal Method (Type, .Int32.)

Determines whether ADS can marshal the specified managed data type.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
bool CanMarshal(  
    Type anyType,  
    int[] args  
)
```

Parameters

anyType	Type: System.Type The Managed data type.
args	Type: .System.Int32 . The arguments.

Return Value

Type: [Boolean](#)
true if this instance can marshal the specified managed type; otherwise, false.

Reference

[IAnyTypeMarshaler Interface](#) [[▶ 2441](#)]

[CanMarshal Overload](#) [[▶ 2444](#)]

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.32.1.1.3 IAnyTypeMarshaler.CanMarshal Method (Type, .Int32., Encoding)

Determines whether ADS can marshal the specified managed data type.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool CanMarshal(
    Type anyType,
    int[] args,
    Encoding encoding
)
```

Parameters

anyType	Type: System.Type The Managed data type.
args	Type: .System.Int32. The arguments.
encoding	Type: System.Text.Encoding The encoding.

Return Value

Type: Boolean

true if this instance can marshal the specified managed type; otherwise, false.

Reference



[IAnyTypeMarshaler Interface \[► 2441\]](#)

[CanMarshal Overload \[► 2444\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.32.1.2 IAnyTypeMarshaler.Marshal Method

Overload List

	Name	Description
	Marshal(Object, Encoding, Span.Byte.) [► 2734]	Marshals the specified value to the specified destination memory / span. (Inherited from ITypeMarshaler [► 2731].)
	Marshal(Object, .Int32., Encoding, Span.Byte.) [► 2447]	Marshals the specified value to the specified destination.

Reference

[IAnyTypeMarshaler Interface \[► 2441\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.32.1.2.1 IAnyTypeMarshaler.Marshal Method (Object, .Int32., Encoding, Span`1)

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
int Marshal(  
    Object anyValue,  
    int[] args,  
    Encoding encoding,  
    Span destination  
)
```

Parameters

anyValue	Type: System.Object
args	Type: .System.Int32.
encoding	Type: System.Text.Encoding
destination	Type: Span

Return Value

Type: [Int32](#)

Reference

[IAnyTypeMarshaler Interface](#) [► 2441]

[Marshal Overload](#) [► 2446]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.32.1.2.2 IAnyTypeMarshaler.Marshal Method (Object, .Int32., Encoding, Span.Byte.)

Marshals the specified value to the specified destination.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Marshal(  
    Object anyValue,  
    int[] args,  
    Encoding encoding,  
    Span<byte> destination  
)
```

Parameters



anyValue	Type: System.Object The value to marshal.
args	Type: .System.Int32. The arguments.
encoding	Type: System.Text.Encoding The encoding.
destination	Type: System.Span.Byte. The destination span.

Return Value

Type: Int32

TheNumber of marshalled bytes.

Reference[IAnyTypeMarshaler Interface \[► 2441\]](#)[Marshal Overload \[► 2446\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.32.1.3 IAnyTypeMarshaler.MarshalSize Method****Overload List**

	Name	Description
	MarshalSize(Object, Encoding) [► 2735]	Gets the byte size of the value when marshalled. (Inherited from ITypeMarshaler [► 2731].)
	MarshalSize(Object, .Int32., Encoding) [► 2448]	Gets the byte size of the value when marshalled.

Reference[IAnyTypeMarshaler Interface \[► 2441\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.32.1.3.1 IAnyTypeMarshaler.MarshalSize Method (Object, .Int32., Encoding)**

Gets the byte size of the value when marshalled.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3**Syntax****C#**

```
int MarshalSize(
    Object anyValue,
    int[] args,
    Encoding encoding
)
```


Parameters

anyValue Type: System.Object
Any value.

args Type: .System.Int32.
The arguments.

encoding Type: System.Text.Encoding
The encoding.

Return Value




Type: Int32
The marshal size of the value.

Reference

- [IAnyTypeMarshaler Interface \[▶ 2441\]](#)
- [MarshalSize Overload \[▶ 2448\]](#)
- [TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.32.1.4 IAnyTypeMarshaler.Unmarshal Method

Overload List

	Name	Description
	Unmarshal(T, ReadOnlySpan<Byte>, Encoding, T.) [▶ 2544]	Unmarshals the source data to an managed value of the specified type. (Inherited from IGenericTypeMarshaler [▶ 2542].)
	Unmarshal(Type, ReadOnlySpan<Byte>, Encoding, Object.) [▶ 2735]	Unmarshals the specified managed type from memory / span (Inherited from ITypeMarshaler [▶ 2731].)
	Unmarshal(Type, Int32, ReadOnlySpan<Byte>, Encoding, Object.) [▶ 2450]	Unmarshals the source data to an managed value of the specified type.

Reference

- [IAnyTypeMarshaler Interface \[▶ 2441\]](#)
- [TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.32.1.4.1 IAnyTypeMarshaler.Unmarshal Method (Type, Int32, ReadOnlySpan<Byte>, Void, Byte)

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)
Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
int Unmarshal(
    Type anyType,
    int[] args,
    ReadOnlySpan source,
    void encoding,
    byte value
)
```

Parameters

anyType	Type: System.Type
args	Type: .System.Int32.
source	Type: ReadOnlySpan
encoding	Type: System.Void
value	Type: System.Byte

Return Value

Type: [Int32](#)

Reference

[IAnyTypeMarshaler Interface](#) [[▶ 2441](#)]

[Unmarshal Overload](#) [[▶ 2449](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.32.1.4.2 IAnyTypeMarshaler.Unmarshal Method (Type, .Int32., ReadOnlySpan.Byte., Encoding, Object.)

Unmarshals the source data to an managed value of the specified type.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Unmarshal(
    Type anyType,
    int[] args,
    ReadOnlySpan<byte> source,
    Encoding encoding,
    out Object value
)
```

Parameters

anyType	Type: System.Type The managed type.
args	Type: .System.Int32. The arguments.

source	Type: System.ReadOnlySpan.Byte. The source data.
encoding	Type: System.Text.Encoding The encoding.
value	Type: System.Object. The value.

Return Value

Type: Int32
Number of unmarshaled bytes.

Reference

- [IAnyTypeMarshaler Interface \[▶ 2441\]](#)
- [Unmarshal Overload \[▶ 2449\]](#)
- [TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.32.1.5 IAnyTypeMarshaler.CanMarshalValue Method

Overload List

	Name	Description
	CanMarshalValue(Object) [▶ 2736]	Determines whether ADS can marshal the specified value (Inherited from ITypeMarshaler [▶ 2731] .)
	CanMarshalValue(Object, Int32, Encoding) [▶ 2451]	Determines whether ADS can marshal the specified managed data type.

Reference

- [IAnyTypeMarshaler Interface \[▶ 2441\]](#)
- [TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.32.1.5.1 IAnyTypeMarshaler.CanMarshalValue Method (Object, Int32, Encoding)

Determines whether ADS can marshal the specified managed data type.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool CanMarshalValue(
    Object value,
    int[] args,
    Encoding encoding
)
```

Parameters

value	Type: System.Object The managed value.
args	Type: .System.Int32. The arguments.
encoding	Type: System.Text.Encoding The encoding.

Return Value

Type: Boolean

true if this instance can marshal the specified managed type; otherwise, false.

Reference[IAnyTypeMarshaler Interface \[► 2441\]](#)[CanMarshalValue Overload \[► 2451\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.32.1.6 IAnyTypeMarshaler.MarshalValueSize Method****Overload List**

	Name	Description
	MarshalValueSize(Object, Encoding) [► 2736]	Gets the byte size of the value when marshalled. (Inherited from ITypeMarshaler [► 2731] .)
	MarshalValueSize(Object, .Int32., Encoding) [► 2452]	Gets the byte size of the value when marshalled.

Reference[IAnyTypeMarshaler Interface \[► 2441\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.32.1.6.1 IAnyTypeMarshaler.MarshalValueSize Method (Object, .Int32., Encoding)**

Gets the byte size of the value when marshalled.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
int MarshalValueSize(
    Object anyValue,
    int[] args,
    Encoding encoding
)
```

Parameters

anyValue	Type: System.Object Any value.
args	Type: .System.Int32. The arguments.
encoding	Type: System.Text.Encoding The encoding.

Return Value

Type: Int32
The marshal size of the value.

Reference

- [IAnyTypeMarshaler Interface \[▶ 2441\]](#)
- [MarshalValueSize Overload \[▶ 2452\]](#)
- [TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.33 IArrayInstance Interface

Interface representing an array instance

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)
Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229










Syntax

C#

```
public interface IArrayInstance : ISymbol,
    IAttributedInstance, IInstance, IBitSize
```

The IArrayInstance type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469] .)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472] .)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472] .)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691] .)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549] .)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549] . (Inherited from IInstance [▶ 2549] .)
	Dimensions [▶ 2456]	Gets the dimensions as read only collection.
	Elements [▶ 2456]	Gets the contained Array Elements as read only collection.
	ElementType [▶ 2457]	Gets the type of the contained elements.

	Name	Description
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is a container type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Item [▶ 2457]	Gets the ISymbol [▶ 2691] with the specified indices.
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference






[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.33.1 IArrayInstance Properties

The [IArrayInstance](#) [▶ 2453] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	Dimensions [▶ 2456]	Gets the dimensions as read only collection.
	Elements [▶ 2456]	Gets the contained Array Elements as read only collection.
	ElementType [▶ 2457]	Gets the type of the contained elements.
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Item [▶ 2457]	Gets the ISymbol [▶ 2691] with the specified indices.

	Name	Description
	Parent [▸ 2696]	Gets the parent Symbol (Inherited from ISymbol [▸ 2691].)
	Size [▸ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▸ 2474] (Inherited from IBitSize [▸ 2472].)
	SubSymbols [▸ 2696]	Gets the SubSymbols of the ISymbol [▸ 2691] (Inherited from ISymbol [▸ 2691].)
	TypeName [▸ 2553]	Gets the name of the DataType [▸ 2475] that is used for this Instance [▸ 2549]. (Inherited from Instance [▸ 2549].)
	ValueEncoding [▸ 2472]	Gets the value encoding. (Inherited from AttributedInstance [▸ 2469].)

Reference

[IArrayInstance Interface](#) [[▸ 2453](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.33.1.1 IArrayInstance.Dimensions Property

Gets the dimensions as read only collection.

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDimensionCollection Dimensions { get; }
```

Property Value

Type: [IDimensionCollection](#) [[▸ 2492](#)]

The dimensions.

Reference

[IArrayInstance Interface](#) [[▸ 2453](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.33.1.2 IArrayInstance.Elements Property

Gets the contained Array Elements as read only collection.

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISymbolCollection<ISymbol> Elements { get; }
```

Property Value

Type: [ISymbolCollection](#) [[▸ 2700](#)].[ISymbol](#) [[▸ 2691](#)].

The elements.

Reference

[IArrayInstance Interface](#) [► 2453]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.33.1.3 IArrayInstance.ElementType Property

Gets the type of the contained elements.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDataType? ElementType { get; }
```

Property Value

Type: [IDataType](#) [► 2475]

The type of the element.

Reference

[IArrayInstance Interface](#) [► 2453]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.33.1.4 IArrayInstance.Item Property

Gets the [ISymbol](#) [► 2691] with the specified indices.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISymbol this[  
    int[] indices  
] { get; }
```

Parameters

indices	Type: <code>.System.Int32</code> . The indices.
---------	--

Property Value

Type: [ISymbol](#) [► 2691]

The [ISymbol](#) [► 2691].

Return Value

Type: [ISymbol](#) [► 2691]



Reference[IArrayInstance Interface](#) [► 2453][TwinCAT.TypeSystem Namespace](#) [► 2083]**6.11.33.2 IArrayInstance Methods**

The [IArrayInstance](#) [► 2453] type exposes the following members.

Extension Methods

	Name	Description
	ReferencesExternalData [► 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [► 3021].)

Reference[IArrayInstance Interface](#) [► 2453][TwinCAT.TypeSystem Namespace](#) [► 2083]**6.11.33.2.1 IArrayInstance.TryGetElement Method****Overload List**

	Name	Description
	TryGetElement(IList<Int32>, ISymbol) [► 2458]	Tries to get the array element with the specified indices (jagged array support).
	TryGetElement(Int32, ISymbol) [► 2459]	Tries to get the array element with specified indices (only first level on jagged arrays)

Reference[IArrayInstance Interface](#) [► 2453][TwinCAT.TypeSystem Namespace](#) [► 2083]**6.11.33.2.1.1 IArrayInstance.TryGetElement Method (IList<Int32>, ISymbol)**

Tries to get the array element with the specified indices (jagged array support).

Namespace: [TwinCAT.TypeSystem](#) [► 2083]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3**Syntax****C#**

```
bool TryGetElement(
    IList<int[]> jaggedIndices,
    out ISymbol?? symbol
)
```

Parameters

jaggedIndices	Type: System.Collections.Generic.IList<Int32> . The jagged indices list.
symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691]. The symbol.

Return Value

Type: [Boolean](#)
true if found, false if the jagged indices specifiers is out-of-range.

Reference

[IArrayInstance Interface](#) [[▶ 2453](#)]

[TryGetElement Overload](#) [[▶ 2458](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.33.2.1.2 IArrayInstance.TryGetElement Method (.Int32., ISymbol.)

Tries to get the array element with specified indices (only first level on jagged arrays)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
bool TryGetElement(  
    int[] indices,  
    out ISymbol?? symbol  
)
```

Parameters

indices	Type: System.Int32 . The indices.
symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691]. The found Array element symbol (out-parameter).

Return Value

Type: [Boolean](#)
true if found, false if the indices specifiers is out-of-range.

Reference

[IArrayInstance Interface](#) [[▶ 2453](#)]

[TryGetElement Overload](#) [[▶ 2458](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.34 IArrayType Interface

Interface representing an array [DataType](#) [[▶ 2475](#)].

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public interface IArrayType : IDatatype,
    IBitSize
```

The IArrayType type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDatatype [▶ 2475] (Inherited from IDatatype [▶ 2475].)
	BitSize [▶ 2473]	Gets the size of the IDatatype [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDatatype [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDatatype [▶ 2475].)
	Dimensions [▶ 2463]	Gets the dimensions as read only collection.
	ElementType [▶ 2463]	Gets the type of the contained elements.
	ElementTypeName [▶ 2463]	Gets the name of the element datatype.
	FullName [▶ 2479]	Gets the full name of the IDatatype [▶ 2475] (Namespace + Name) (Inherited from IDatatype [▶ 2475].)
	Id [▶ 2480]	Gets the ID of the Data Type (Inherited from IDatatype [▶ 2475].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full Data Type but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsJagged [▶ 2464]	Gets a value indicating whether this instance is jagged.
	JaggedLevel [▶ 2464]	Gets the jagged level (Non-Jagged Arrays have level 1)
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDatatype [▶ 2475].)
	Namespace [▶ 2482]	Gets the namespace string within the IDatatype [▶ 2475] exists. (Inherited from IDatatype [▶ 2475].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Extension Methods

	Name	Description
	IsArrayOfPrimitives. [▶ 3025]	Overloaded.

Name	Description
	Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsContainer. [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive. [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
Resolve. [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

Also see about this

- ▶ [IDataType.IsContainer Property](#) [▶ 2480]
- ▶ [IDataType.IsPointer Property](#) [▶ 2481]
- ▶ [IDataType.IsPrimitive Property](#) [▶ 2481]
- ▶ [IDataType.IsReference Property](#) [▶ 2482]

6.11.34.1 IArrayType Properties

The [IArrayType \[▸ 2459\]](#) type exposes the following members.

Properties





	Name	Description
	Attributes [▸ 2478]	Gets the attributes of the IDataType [▸ 2475] (Inherited from IDataType [▸ 2475] .)
	BitSize [▸ 2473]	Gets the size of the IDataType [▸ 2475] in bits. (Inherited from IBitSize [▸ 2472] .)
	ByteSize [▸ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▸ 2472] .)
	Category [▸ 2478]	Gets the Data Type category (Inherited from IDataType [▸ 2475] .)
	Comment [▸ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▸ 2475] .)
	Dimensions [▸ 2463]	Gets the dimensions as read only collection.
	ElementType [▸ 2463]	Gets the type of the contained elements.
	ElementTypeName [▸ 2463]	Gets the name of the element datatype.
	FullName [▸ 2479]	Gets the full name of the IDataType [▸ 2475] (Namespace + Name) (Inherited from IDataType [▸ 2475] .)
	Id [▸ 2480]	Gets the ID of the Data Type (Inherited from IDataType [▸ 2475] .)
	IsBitType [▸ 2474]	Gets a value indicating whether this instance is not basing on a full Data Type but instead of some sort of bit mapping (Inherited from IBitSize [▸ 2472] .)
	IsByteAligned [▸ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▸ 2472] .)
	IsJagged [▸ 2464]	Gets a value indicating whether this instance is jagged.
	JaggedLevel [▸ 2464]	Gets the jagged level (Non-Jagged Arrays have level 1)
	Name [▸ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▸ 2475] .)
	Namespace [▸ 2482]	Gets the namespace string within the IDataType [▸ 2475] exists. (Inherited from IDataType [▸ 2475] .)
	Size [▸ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▸ 2474] (Inherited from IBitSize [▸ 2472] .)

Reference

[IArrayType Interface \[▸ 2459\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

Also see about this

-  [IDataType.IsContainer Property \[▸ 2480\]](#)
-  [IDataType.IsPointer Property \[▸ 2481\]](#)
-  [IDataType.IsPrimitive Property \[▸ 2481\]](#)
-  [IDataType.IsReference Property \[▸ 2482\]](#)

6.11.34.1.1 IArrayType.Dimensions Property

Gets the dimensions as read only collection.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDimensionCollection Dimensions { get; }
```

Property Value

Type: [IDimensionCollection](#) [[▶ 2492](#)]

The dimensions.

Reference

[IArrayType Interface](#) [[▶ 2459](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.34.1.2 IArrayType.ElementType Property

Gets the type of the contained elements.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDataType? ElementType { get; }
```

Property Value

Type: [IDataType](#) [[▶ 2475](#)]

The type of the element or **null** if not resolved.

Reference

[IArrayType Interface](#) [[▶ 2459](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.34.1.3 IArrayType.ElementTypeName Property

Gets the name of the element datatype.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string ElementTypeName { get; }
```

Property Value

Type: String
The name of the element datatype.

Reference

[IArrayType Interface](#) [► 2459]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.34.1.4 IArrayType.IsJagged Property

Gets a value indicating whether this instance is jagged.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsJagged { get; }
```

Property Value

Type: Boolean
true if this instance is jagged; otherwise, false.

Reference

[IArrayType Interface](#) [► 2459]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.34.1.5 IArrayType.JaggedLevel Property

Gets the jagged level (Non-Jagged Arrays have level 1)

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int JaggedLevel { get; }
```

Property Value

Type: Int32
The jagged level.

Reference

[IArrayType Interface](#) [► 2459]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.34.2 IArrayType Methods

The [IArrayType](#) [► 2459] type exposes the following members.

Extension Methods

Name	Description
IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[IArrayType Interface](#) [[▶ 2459](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.35 IArrayValue Interface

Interface [IArrayValue](#)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229








Syntax

C#










```
public interface IArrayValue : IValue
```

The IArrayValue type exposes the following members.

Properties

	Name	Description
	Age [▸ 2746]	Gets the age of the value (last successful read of the value) (Inherited from IValue [▸ 2745].)
	CachedRaw [▸ 2746]	Gets the cached Raw internal Data. (Inherited from IValue [▸ 2745].)
	DataType [▸ 2747]	Gets the data type bound to this IValue [▸ 2745] (Inherited from IValue [▸ 2745].)
	IsPrimitive [▸ 2747]	Gets a value indicating whether this IValue [▸ 2745] is a primitive value. (Inherited from IValue [▸ 2745].)
	Symbol [▸ 2748]	Gets the symbol bound to this IValue [▸ 2745] . (Inherited from IValue [▸ 2745].)
	TimeStamp [▸ 2748]	Gets the Time stamp of the last successful read of the Value (local user time, UTC) (Inherited from IValue [▸ 2745].)
	UpdateMode [▸ 2748]	Gets the update mode (not implemented yet) (Inherited from IValue [▸ 2745].)

Methods

	Name	Description
	Read [▸ 2749]	Reads the value (via ADS) (Inherited from IValue [▸ 2745].)
	ReadAsync [▸ 2750]	Reads the value (via ADS) (Inherited from IValue [▸ 2745].)
	ResolveValue [▸ 2750]	Resolves the Value object to its primitive value. (Inherited from IValue [▸ 2745].)
	TryGetArrayElement Values [▸ 2468]	Returns Array Element values.
	TryGetIndexValue [▸ 2468]	Tries to get the specified Array Element
	TryResolveValue [▸ 2751]	Tries to resolves the Value object to its primitive value. (Inherited from IValue [▸ 2745].)
	TrySetIndexValue [▸ 2469]	Tries to set the indexed value on Arrays
	Write [▸ 2751]	Writes the value (via ADS) (Inherited from IValue [▸ 2745].)
	WriteAsync [▸ 2751]	Writes the value (via ADS) (Inherited from IValue [▸ 2745].)

Reference








[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

[TwinCAT.TypeSystem.IValue \[▸ 2745\]](#)

6.11.35.1 IArrayValue Properties

The [IArrayValue \[▸ 2465\]](#) type exposes the following members.

Properties

	Name	Description
	Age [▸ 2746]	Gets the age of the value (last successful read of the value) (Inherited from IValue [▸ 2745].)
	CachedRaw [▸ 2746]	Gets the cached Raw internal Data. (Inherited from IValue [▸ 2745].)
	DataType [▸ 2747]	Gets the data type bound to this IValue [▸ 2745] (Inherited from IValue [▸ 2745].)
	IsPrimitive [▸ 2747]	Gets a value indicating whether this IValue [▸ 2745] is a primitive value. (Inherited from IValue [▸ 2745].)
	Symbol [▸ 2748]	Gets the symbol bound to this IValue [▸ 2745] . (Inherited from IValue [▸ 2745].)
	TimeStamp [▸ 2748]	Gets the Time stamp of the last successful read of the Value (local user time, UTC) (Inherited from IValue [▸ 2745].)
	UpdateMode [▸ 2748]	Gets the update mode (not implemented yet) (Inherited from IValue [▸ 2745].)

Reference










[IArrayValue Interface \[▸ 2465\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.35.2 IArrayValue Methods

The [IArrayValue \[▸ 2465\]](#) type exposes the following members.

Methods

	Name	Description
	Read [▸ 2749]	Reads the value (via ADS) (Inherited from IValue [▸ 2745].)
	ReadAsync [▸ 2750]	Reads the value (via ADS) (Inherited from IValue [▸ 2745].)
	ResolveValue [▸ 2750]	Resolves the Value object to its primitive value. (Inherited from IValue [▸ 2745].)
	TryGetArrayElement Values [▸ 2468]	Returns Array Element values.
	TryGetIndexValue [▸ 2468]	Tries to get the specified Array Element
	TryResolveValue [▸ 2751]	Tries to resolves the Value object to its primitive value. (Inherited from IValue [▸ 2745].)
	TrySetIndexValue [▸ 2469]	Tries to set the indexed value on Arrays
	Write [▸ 2751]	Writes the value (via ADS) (Inherited from IValue [▸ 2745].)
	WriteAsync [▸ 2751]	Writes the value (via ADS) (Inherited from IValue [▸ 2745].)

Reference[IArrayValue Interface \[▸ 2465\]](#)[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)**6.11.35.2.1 IArrayValue.TryGetArrayElementValues Method**

Returns Array Element values.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
bool TryGetArrayElementValues(
    out IEnumerable<Object>? elementValues
)
```

Parameters

elementValues	Type: System.Collections.Generic.IEnumerable.Object.. The element values.
---------------	--

Return Value

Type: Boolean

true if the element values are returned, false otherwise.

Reference[IArrayValue Interface \[▸ 2465\]](#)[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)**6.11.35.2.2 IArrayValue.TryGetIndexValue Method**

Tries to get the specified Array Element

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
bool TryGetIndexValue(
    int[] indices,
    out Object?? value
)
```

Parameters

indices	Type: .System.Int32. The indices.
value	Type: System.Object. The value.

Return Value

Type: Boolean
true if the indexed value is returned, false otherwise.

Reference

[IArrayValue Interface](#) [► 2465]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.35.2.3 IArrayValue.TrySetIndexValue Method

Tries to set the indexed value on Arrays

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
bool TrySetIndexValue (
    Object[] indexes,
    Object value
)
```

Parameters

indexes	Type: .System.Object. The indexes.
value	Type: System.Object The value.

Return Value

Type: Boolean
true if succeeded, false otherwise.

Reference

[IArrayValue Interface](#) [► 2465]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.36 IAttributedInstance Interface

Interface IAttributedInstance

Namespace: [TwinCAT.TypeSystem](#) [► 2083]
















Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public interface IAttributedInstance : IInstance,
    IBitSize
```

The IAttributedInstance type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes.
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full IDataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	TypeName [▶ 2553]	Gets the name of the IDataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding.

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference
















[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

[TwinCAT.TypeSystem.IInstance](#) [[▶ 2549](#)]

6.11.36.1 IAttributedInstance Properties

The [IAttributedInstance](#) [[▶ 2469](#)] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes.
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full IDataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($BitSize \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	TypeName [▶ 2553]	Gets the name of the IDataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding.

Reference

[IAttributedInstance Interface](#) [[▶ 2469](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.36.1.1 IAttributedInstance.Attributes Property

Gets the Type Attributes.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ITypeAttributeCollection Attributes { get; }
```

Property Value

Type: [ITypeAttributeCollection](#) [[▶ 2726](#)]

The attributes.

Reference

[IAttributedInstance Interface](#) [[▶ 2469](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.36.1.2 IAttributedInstance.ValueEncoding Property

Gets the value encoding.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
Encoding ValueEncoding { get; }
```

Property Value

Type: Encoding

The value encoding.

Reference

[IAttributedInstance Interface](#) [[▶ 2469](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.36.2 IAttributedInstance Methods

The [IAttributedInstance](#) [[▶ 2469](#)] type exposes the following members.

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[IAttributedInstance Interface](#) [[▶ 2469](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.37 IBitSize Interface

Interface IBitSize

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229






Syntax

C#

```
public interface IBitSize
```

The IBitSize type exposes the following members.

Properties

	Name	Description
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits.
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474]

Remarks

Specifies the Bitness of the the object and the bit resp. byte sizes.






Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.37.1 IBitSize Properties

The [IBitSize \[▶ 2472\]](#) type exposes the following members.

Properties

	Name	Description
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits.
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474]

Reference

[IBitSize Interface \[▶ 2472\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.37.1.1 IBitSize.BitSize Property

Gets the size of the [IDataType \[▶ 2475\]](#) in bits.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int BitSize { get; }
```

Property Value

Type: Int32

The size of [IDataType \[▸ 2475\]](#) in bits.

Reference

[IBitSize Interface \[▸ 2472\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.37.1.2 IBitSize.ByteSize Property

Gets the (aligned) size of of the Type/Instance in Bytes

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int ByteSize { get; }
```

Property Value

Type: Int32

The size of the byte.

Reference

[IBitSize Interface \[▸ 2472\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.37.1.3 IBitSize.IsBitType Property

Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsBitType { get; }
```

Property Value

Type: Boolean

true if this instance is bit mapping; otherwise, false.

Reference

[IBitSize Interface](#) [[▶ 2472](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.37.1.4 IBitSize.IsByteAligned Property

Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsByteAligned { get; }
```

Property Value

Type: Boolean

true if this instance is byte aligned; otherwise, false.

Reference

[IBitSize Interface](#) [[▶ 2472](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.37.1.5 IBitSize.Size Property

Gets the size of the object in bytes or Bits dependant on [IsBitType](#) [[▶ 2474](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Size { get; }
```

Property Value

Type: Int32

The size of the bit.

Reference

[IBitSize Interface](#) [[▶ 2472](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.38 IDataType Interface

Base interface for objects representing data types

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229













Syntax

C#

```
public interface IDataTypeInfo : IBitSize
```

The IDataTypeInfo type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataTypeInfo
	BitSize [▶ 2473]	Gets the size of the IDataTypeInfo in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category
	Comment [▶ 2479]	Gets the comment behind the variable declaration.
	FullName [▶ 2479]	Gets the full name of the IDataTypeInfo (Namespace + Name)
	Id [▶ 2480]	Gets the ID of the DataType
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	Name [▶ 2482]	Gets the name of the Data Type (without namespace)
	Namespace [▶ 2482]	Gets the namespace string within the IDataTypeInfo exists.
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Extension Methods





	Name	Description
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataTypeInfo is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataTypeInfo is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataTypeInfo is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive [▶ 3033]	Overloaded.

	Name	Description
		Gets a value indicating whether this IDataTyep is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataTyep is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataTyep is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]






Also see about this








-  [IDataType.IsContainer Property](#) [▶ 2480]
-  [IDataType.IsPointer Property](#) [▶ 2481]
-  [IDataType.IsPrimitive Property](#) [▶ 2481]
-  [IDataType.IsReference Property](#) [▶ 2482]

6.11.38.1 IDataTyep Properties

The [IDataTyep](#) [▶ 2475] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataTyep [▶ 2475]
	BitSize [▶ 2473]	Gets the size of the IDataTyep [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category
	Comment [▶ 2479]	Gets the comment behind the variable declaration.





	Name	Description
	FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name)
	Id [▶ 2480]	Gets the ID of the DataType
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from IBitSize [▶ 2472].)
	Name [▶ 2482]	Gets the name of the Data Type (without namespace)
	Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists.
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Reference

[IDataType Interface](#) [[▶ 2475](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

Also see about this

-  [IDataType.IsContainer](#) Property [[▶ 2480](#)]
-  [IDataType.IsPointer](#) Property [[▶ 2481](#)]
-  [IDataType.IsPrimitive](#) Property [[▶ 2481](#)]
-  [IDataType.IsReference](#) Property [[▶ 2482](#)]

6.11.38.1.1 IDatatype.Attributes Property

Gets the attributes of the [IDataType](#) [[▶ 2475](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ITypeAttributeCollection Attributes { get; }
```

Property Value

Type: [ITypeAttributeCollection](#) [[▶ 2726](#)]

The attributes.

Reference

[IDataType Interface](#) [[▶ 2475](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.38.1.2 IDatatype.Category Property

Gets the Data Type category

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
DataTypeCategory Category { get; }
```

Property Value

Type: [DataTypeCategory](#) [► 2111]

The category.

Reference

[IDataType Interface](#) [► 2475]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.38.1.3 **IDataType.Comment Property**

Gets the comment behind the variable declaration.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string Comment { get; }
```

Property Value

Type: String

Comment behind the variable declaration.

Reference

[IDataType Interface](#) [► 2475]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.38.1.4 **IDataType.FullName Property**

Gets the full name of the [IDataType](#) [► 2475] (Namespace + Name)

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string FullName { get; }
```

Property Value

Type: String

The full name.

Reference

[IDataType Interface](#) [► 2475]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.38.1.5 IDataTypeId Property

Gets the ID of the DataType

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Id { get; }
```

Property Value

Type: [Int32](#)
The id.

Reference

[IDataType Interface](#) [► 2475]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.38.1.6 IDataTypeIdContainer Property

Gets a value indicating whether this [IDataType](#) [► 2475] is a container type

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
bool IsContainer { get; }
```

Property Value

Type: [Boolean](#)
true if this instance is container type; otherwise, false.

Remarks

Container Types are all types that contain SubElements like

- [Array](#) [► 2111]
- [Pointer](#) [► 2111]
- [Union](#) [► 2111]
- [Struct](#) [► 2111]
- [Function](#) [► 2111]
- [FunctionBlock](#) [► 2111]
- [Program](#) [► 2111]

and the [Alias](#) [▸ 2111] and [Reference](#) [▸ 2111] types, if they have a container type as base type.

Reference

[IDataType Interface](#) [▸ 2475]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

[IDataType.Category](#) [▸ 2478]

6.11.38.1.7 IDatatype.IsPointer Property

Gets a value indicating whether this [IDataType](#) [▸ 2475] is a pointer type

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
bool IsPointer { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is pointer type; otherwise, false.

Remarks

Pointer types can be dereferenced with the '^' operator.

Reference

[IDataType Interface](#) [▸ 2475]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

[IDataType.Category](#) [▸ 2478]

6.11.38.1.8 IDatatype.IsPrimitive Property

Gets a value indicating whether this [IDataType](#) [▸ 2475] is primitive

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
bool IsPrimitive { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is primitive; otherwise, false.

Reference

[IDataType Interface](#) [▸ 2475]

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.38.1.9 IDataType.IsReference Property

Gets a value indicating whether this [IDataType \[▸ 2475\]](#) is a reference type

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
bool IsReference { get; }
```

Property Value

Type: [Boolean](#)

true if this instance is container type; otherwise, false.

Remarks

Reference types can be dereferenced.

Reference

[IDataType Interface \[▸ 2475\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

[IDataType.Category \[▸ 2478\]](#)

6.11.38.1.10 IDataType.Name Property

Gets the name of the Data Type (without namespace)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string Name { get; }
```

Property Value

Type: [String](#)

The name.

Reference

[IDataType Interface \[▸ 2475\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.38.1.11 IDataType.Namespace Property

Gets the namespace string within the [IDataType \[▸ 2475\]](#) exists.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string Namespace { get; }
```

Property Value

Type: String
The namespace.

Reference

[IDataType Interface \[▸ 2475\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.38.2 IDataType Methods

The [IDataType \[▸ 2475\]](#) type exposes the following members.

Extension Methods

Name	Description
IsArrayOfPrimitives. [▸ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▸ 3021].)
IsArrayOfPrimitives(Boolean) [▸ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▸ 3021].)
IsContainer. [▸ 3027]	Overloaded. Gets a value indicating whether this IDataType [▸ 2475] is a container type (Defined by DataTypeExtension [▸ 3021].)
IsContainer(Boolean) [▸ 3028]	Overloaded. Gets a value indicating whether this IDataType [▸ 2475] is a container type (Defined by DataTypeExtension [▸ 3021].)
IsPointer [▸ 3031]	Gets a value indicating whether this IDataType [▸ 2475] is a pointer type (Defined by DataTypeExtension [▸ 3021].)
IsPrimitive. [▸ 3033]	Overloaded. Gets a value indicating whether this IDataType [▸ 2475] is primitive (Defined by DataTypeExtension [▸ 3021].)
IsPrimitive(Boolean) [▸ 3034]	Overloaded. Gets a value indicating whether this IDataType [▸ 2475] is primitive (Defined by DataTypeExtension [▸ 3021].)
IsReference [▸ 3036]	Gets a value indicating whether this IDataType [▸ 2475] is a reference type (Defined by DataTypeExtension [▸ 3021].)
ReferencesExternalData [▸ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▸ 3021].)
ResolvableAsPrimitive. [▸ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▸ 3021].)

	Name	Description
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021] .)
	Resolve. [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021] .)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021] .)

Reference

[IDataType Interface \[▶ 2475\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.39 IDataTypeCollection Interface

Interface IDataTypeCollection Implements the ICollection.T.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229





Syntax

C#




```
public interface IDataTypeCollection : IDataTypeCollection<IDataType>,
    IList<IDataType>, ICollection<IDataType>, IEnumerable<IDataType>,
    IEnumerable
```









The IDataTypeCollection type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection. IDataType [▶ 2475] ..)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection. IDataType [▶ 2475] ..)
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList. IDataType [▶ 2475] ..)
	Item.String. [▶ 2488]	Gets the IDataType [▶ 2475] with the specified name. (Inherited from IDataTypeCollection.T. [▶ 2486] .)

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection. IDataType [▶ 2475] ..)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection. IDataType [▶ 2475] ..)
	Contains	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection. IDataType [▶ 2475] ..)

	Name	Description
	ContainsType [▶ 2489]	Determines whether the container contains the specified IDataType [▶ 2475]. (Inherited from IDataTypeCollection.T. [▶ 2486].)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.IDataType [▶ 2475].)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.IDataType [▶ 2475].)
	IndexOf	Determines the index of a specific item in the IList.T. (Inherited from IList.IDataType [▶ 2475].)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.IDataType [▶ 2475].)
	Remove	Removes the first occurrence of a specific object from the ICollection.T. (Inherited from ICollection.IDataType [▶ 2475].)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.IDataType [▶ 2475].)
	TryGetType [▶ 2490]	Tries to get the specified IDataType [▶ 2475] from the IDataTypeCollection.T. [▶ 2486]. (Inherited from IDataTypeCollection.T. [▶ 2486].)

Reference





[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

[System.Collections.Generic.ICollection.T.](#)

6.11.39.1 IDataValueCollection Properties

The [IDataValueCollection](#) [▶ 2484] type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T. (Inherited from ICollection.IDataType [▶ 2475].)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.IDataType [▶ 2475].)
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.IDataType [▶ 2475].)
	Item.String. [▶ 2488]	Gets the IDataType [▶ 2475] with the specified name. (Inherited from IDataValueCollection.T. [▶ 2486].)

Reference












[IDataValueCollection Interface](#) [▶ 2484]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.39.2 IDataValueCollection Methods

The [IDataValueCollection](#) [▶ 2484] type exposes the following members.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.IDataType [▶ 2475]..)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.IDataType [▶ 2475]..)
	Contains	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.IDataType [▶ 2475]..)
	ContainsType [▶ 2489]	Determines whether the container contains the specified IDataType [▶ 2475]. (Inherited from IDataTypeCollection.T. [▶ 2486]..)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.IDataType [▶ 2475]..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.IDataType [▶ 2475]..)
	IndexOf	Determines the index of a specific item in the IList.T.. (Inherited from IList.IDataType [▶ 2475]..)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.IDataType [▶ 2475]..)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.IDataType [▶ 2475]..)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.IDataType [▶ 2475]..)
	TryGetType [▶ 2490]	Tries to get the specified IDataType [▶ 2475] from the IDataTypeCollection.T. [▶ 2486]. (Inherited from IDataTypeCollection.T. [▶ 2486]..)

Reference

[IDataTypeCollection Interface](#) [[▶ 2484](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.40 IDatatypeCollection.T. Interface

Data Type container interface

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#





```
public interface IDatatypeCollection<T> : IList<T>,
    ICollection<T>, IEnumerable<T>, IEnumerable
where T : class, IDataType
```

Type Parameters












T	Data Type type.
---	-----------------

The IDatatypeCollection.T. type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.T..)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.T..)
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.T..)
	<u>Item.String.</u> ▶ 2488	Gets the IDataType ▶ 2475 with the specified name.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.T..)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.T..)
	Contains	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.T..)
	<u>ContainsType</u> ▶ 2489	Determines whether the container contains the specified IDataType ▶ 2475 .
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.T..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.T..)
	IndexOf	Determines the index of a specific item in the IList.T.. (Inherited from IList.T..)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.T..)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.T..)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.T..)
	<u>TryGetType</u> ▶ 2490	Tries to get the specified IDataType ▶ 2475 from the IDataTypeCollection.T.





Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083](#)

6.11.40.1 IDataValueCollection.T. Properties

The [IDataValueCollection.T.](#) [▶ 2486](#) generic type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.T. ▶ 2486 ..)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.T. ▶ 2486 ..)
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.T. ▶ 2486 ..)
	<u>Item.String.</u> ▶ 2488	Gets the IDataType ▶ 2475 with the specified name.

Reference[IDataTypeCollection.T. Interface](#) [[▶ 2486](#)][TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]**6.11.40.1.1 IDataTypeCollection.T..Item Property****Overload List**

	Name	Description
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.T [▶ 2486]..)
	Item.String. [▶ 2488]	Gets the IDataType [▶ 2475] with the specified name.

Reference[IDataTypeCollection.T. Interface](#) [[▶ 2486](#)][TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]**6.11.40.1.1.1 IDataTypeCollection.T..Item Property (String)**Gets the [IDataType](#) [[▶ 2475](#)] with the specified name.**Namespace:** [TwinCAT.TypeSystem](#) [[▶ 2083](#)]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
T this[
    string name
] { get; }
```

Parameters












name	Type: System.String The name.
------	----------------------------------

Return ValueType: [T](#) [[▶ 2486](#)]

T.

Reference[IDataTypeCollection.T. Interface](#) [[▶ 2486](#)][Item Overload](#) [[▶ 2488](#)][TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]**6.11.40.2 IDataTypeCollection.T. Methods**The [IDataTypeCollection.T.](#) [[▶ 2486](#)] generic type exposes the following members.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.T [▶ 2486]..)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.T [▶ 2486]..)
	Contains	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.T [▶ 2486]..)
	ContainsType [▶ 2489]	Determines whether the container contains the specified IDataType [▶ 2475] .
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.T [▶ 2486]..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.T [▶ 2486]..)
	IndexOf	Determines the index of a specific item in the IList.T.. (Inherited from IList.T [▶ 2486]..)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.T [▶ 2486]..)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.T [▶ 2486]..)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.T [▶ 2486]..)
	TryGetType [▶ 2490]	Tries to get the specified IDataType [▶ 2475] from the IDataTypeCollection.T. [▶ 2486] .

Reference

[IDataTypeCollection.T. Interface \[▶ 2486\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.40.2.1 IDataValueCollection.T..ContainsType Method

Determines whether the container contains the specified [IDataType \[▶ 2475\]](#).

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool ContainsType(
    string name
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: Boolean
true if contained; otherwise, false.

Reference[IDataTypeCollection.T. Interface](#) [► 2486][TwinCAT.TypeSystem Namespace](#) [► 2083]**6.11.40.2.2 IDataTypeCollection.T..TryGetType Method**Tries to get the specified [IDataType](#) [► 2475] from the [IDataTypeCollection.T.](#) [► 2486].**Namespace:** [TwinCAT.TypeSystem](#) [► 2083]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
bool TryGetType (
    string name,
    out T? type
)
```

Parameters



name	Type: System.String The name.
type	Type: T [► 2486]. The type (Out parameter)

Return ValueType: Boolean
true if found**Reference**[IDataTypeCollection.T. Interface](#) [► 2486][TwinCAT.TypeSystem Namespace](#) [► 2083]**6.11.41 IDimension Interface**Interface representing a single Dimension of an [ArrayType](#) [► 2459].**Namespace:** [TwinCAT.TypeSystem](#) [► 2083]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public interface IDimension
```

The IDimension type exposes the following members.

Properties

	Name	Description
	ElementCount [► 2491]	Gets the number of elements within that IDimension.
	LowerBound [► 2491]	Gets the lower bound of elements within that IDimension.



Reference

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.41.1 IDimension Properties

The [IDimension \[► 2490\]](#) type exposes the following members.

Properties

	Name	Description
	ElementCount [► 2491]	Gets the number of elements within that IDimension [► 2490] .
	LowerBound [► 2491]	Gets the lower bound of elements within that IDimension [► 2490] .

Reference

[IDimension Interface \[► 2490\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.41.1.1 IDimension.ElementCount Property

Gets the number of elements within that [IDimension \[► 2490\]](#).

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int ElementCount { get; }
```

Property Value

Type: Int32

The element count.

Reference

[IDimension Interface \[► 2490\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.41.1.2 IDimension.LowerBound Property

Gets the lower bound of elements within that [IDimension \[► 2490\]](#).

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int LowerBound { get; }
```

Property Value

Type: Int32
The lower bound.

Reference

[IDimension Interface \[► 2490\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.42 IDimensionCollection Interface

Interface IDimensionCollection

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)








Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**





```
public interface IDimensionCollection : IList<IDimension>,
    ICollection<IDimension>, IEnumerable<IDimension>, IEnumerable
```







The IDimensionCollection type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.IDimension [► 2490].)
	ElementCount [► 2494]	Gets the Number of elements in all Dimensions
	IsNonZeroBased [► 2495]	Gets a value indicating whether this instance is non zero index based.
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.IDimension [► 2490].)
	Item	Gets or sets the element at the specified index. (Inherited from IList.IDimension [► 2490].)
	LowerBounds [► 2494]	Gets the lower bounds.
	UpperBounds [► 2494]	Gets the lower bounds.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.IDimension [► 2490].)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.IDimension [► 2490].)
	Contains	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.IDimension [► 2490].)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.IDimension [► 2490].)

	Name	Description
	GetDimensionLengths [▶ 2496]	Gets an array the specifies the Lengths of each Array Dimension
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.IDimension [▶ 2490]..)
	IndexOf	Determines the index of a specific item in the IList.T.. (Inherited from IList.IDimension [▶ 2490]..)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.IDimension [▶ 2490]..)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.IDimension [▶ 2490]..)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.IDimension [▶ 2490]..)

Extension Methods

	Name	Description
	AddDimension [▶ 2982]	Adds a Dimension (FluentInterface) (Defined by IDimensionCollectionExtension [▶ 2981]..)

Reference








[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

[TwinCAT.TypeSystem.IDimension](#) [[▶ 2490](#)]

6.11.42.1 IDimensionCollection Properties

The [IDimensionCollection](#) [[▶ 2492](#)] type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.IDimension [▶ 2490]..)
	ElementCount [▶ 2494]	Gets the Number of elements in all Dimensions
	IsNonZeroBased [▶ 2495]	Gets a value indicating whether this instance is non zero index based.
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.IDimension [▶ 2490]..)
	Item	Gets or sets the element at the specified index. (Inherited from IList.IDimension [▶ 2490]..)
	LowerBounds [▶ 2494]	Gets the lower bounds.
	UpperBounds [▶ 2494]	Gets the lower bounds.

Reference

[IDimensionCollection Interface](#) [[▶ 2492](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.42.1.1 IDimensionCollection.ElementCount Property

Gets the Number of elements in all Dimensions

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int ElementCount { get; }
```

Property Value

Type: Int32

Reference

[IDimensionCollection Interface](#) [[▶ 2492](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.42.1.2 IDimensionCollection.LowerBounds Property

Gets the lower bounds.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int[] LowerBounds { get; }
```

Property Value

Type: .Int32.

The lower bounds.

Reference

[IDimensionCollection Interface](#) [[▶ 2492](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.42.1.3 IDimensionCollection.UpperBounds Property

Gets the lower bounds.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int[] UpperBounds { get; }
```

Property Value

Type: .Int32.
The lower bounds.

Reference

[IDimensionCollection Interface](#) |> [2492](#)

[TwinCAT.TypeSystem Namespace](#) |> [2083](#)

6.11.42.1.4 IDimensionCollection.IsNonZeroBased Property

Gets a value indicating whether this instance is non zero index based.

Namespace: [TwinCAT.TypeSystem](#) |> [2083](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsNonZeroBased { get; }
```

Property Value

Type: Boolean
true if this instance is non zero indexed based; otherwise, false.

Reference








[IDimensionCollection Interface](#) |> [2492](#)




[TwinCAT.TypeSystem Namespace](#) |> [2083](#)

6.11.42.2 IDimensionCollection Methods

The [IDimensionCollection](#) |> [2492](#) type exposes the following members.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.IDimension > 2490)..)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.IDimension > 2490)..)
	Contains	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.IDimension > 2490)..)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.IDimension > 2490)..)
	GetDimensionLengths > 2496	Gets an array the specifies the Lengths of each Array Dimension
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.IDimension > 2490)..)
	IndexOf	Determines the index of a specific item in the IList.T.. (Inherited from IList.IDimension > 2490)..)

	Name	Description
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.IDimension ▶ 2490 ..)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.IDimension ▶ 2490 ..)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.IDimension ▶ 2490 ..)

Extension Methods

	Name	Description
	AddDimension ▶ 2982	Adds a Dimension (FluentInterface) (Defined by IDimensionCollectionExtension ▶ 2981 .)

Reference

[IDimensionCollection Interface](#) [▶ 2492](#)

[TwinCAT.TypeSystem Namespace](#) [▶ 2083](#)

6.11.42.2.1 IDimensionCollection.GetDimensionLengths Method

Gets an array the specifies the Lengths of each Array Dimension

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int[] GetDimensionLengths ()
```

Return Value

Type: .Int32.
System.Int32[].

Reference

[IDimensionCollection Interface](#) [▶ 2492](#)

[TwinCAT.TypeSystem Namespace](#) [▶ 2083](#)

6.11.43 IDynamicSymbol Interface

Interface IDynamicSymbol

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229









Syntax


C#

```
public interface IDynamicSymbol : ISymbol,
    IAttributedInstance, IInstance, IBitSize
```

The IDynamicSymbol type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full IDataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	NormalizedName [▶ 2499]	Gets the normalized instance name (fixed name for dynamic property access)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)

	Name	Description
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
















Reference










[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.43.1 IDynamicSymbol Properties

The [IDynamicSymbol](#) [▶ 2496] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)

	Name	Description
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	NormalizedName [▶ 2499]	Gets the normalized instance name (fixed name for dynamic property access)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Reference

[IDynamicSymbol Interface](#) [[▶ 2496](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.43.1.1 IDynamicSymbol.NormalizedName Property

Gets the normalized instance name (fixed name for dynamic property access)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string NormalizedName { get; }
```

Property Value

Type: String

The normalized instance name (can be the same like [InstanceName](#) [[▶ 2551](#)])

Reference

[IDynamicSymbol Interface](#) [[▶ 2496](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

[IInstance.InstanceName](#) [[▶ 2551](#)]

[ISymbolFactory.InvalidCharacters](#) [[▶ 2705](#)]

6.11.43.2 IDynamicSymbol Methods

The [IDynamicSymbol](#) [[▶ 2496](#)] type exposes the following members.

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[IDynamicSymbol Interface](#) [[▶ 2496](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.44 IDynamicSymbolLoader Interface

Dynamic symbol loader interface

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229








Syntax

C#



```
public interface IDynamicSymbolLoader : ISymbolLoader,
    ISymbolProvider, ISymbolServer
```





The IDynamicSymbolLoader type exposes the following members.

Properties

	Name	Description
	BuildInTypes [▶ 2716]	Gets the build in types. (Inherited from ISymbolLoader [▶ 2714].)
	DataTypes [▶ 2720]	Gets the data types (Inherited from ISymbolServer [▶ 2719].)
	DefaultValueEncoding [▶ 2721]	Gets the default value encoding. (Inherited from ISymbolServer [▶ 2719].)
	RootNamespaceName [▶ 2718]	Gets the name of the root namespace (Inherited from ISymbolProvider [▶ 2717].)
	Settings [▶ 2716]	Gets or sets the access Method (Inherited from ISymbolLoader [▶ 2714].)
	Symbols [▶ 2721]	Gets the symbols. (Inherited from ISymbolServer [▶ 2719].)
	SymbolsDynamic [▶ 2501]	Gets the symbols (late bound as dynamic objects)

Methods

	Name	Description
	GetDataTypesAsync [▶ 2722]	Gets the data types asynchronously. (Inherited from ISymbolServer [▶ 2719].)
	GetDynamicSymbolsAsync [▶ 2502]	Gets the dynamic symbols asynchronously

	Name	Description
	GetSymbolsAsync [▶ 2723]	Gets the symbols asynchronously (Inherited from ISymbolServer [▶ 2719].)
	ResetCachedSymbolicData [▶ 2724]	Resets the cached symbolic data. (Inherited from ISymbolServer [▶ 2719].)
	TryGetDataTypes [▶ 2723]	Tries to get the symbols from the device target. (Inherited from ISymbolServer [▶ 2719].)
	TryGetSymbols [▶ 2724]	Tries to geth the symbols from the device target. (Inherited from ISymbolServer [▶ 2719].)

Remarks

Dynamic symbols are late bound symbols who are created and will expand during runtime. These symbols can represent complex user defined type instances like PLC Structures and Arrays created during PLC access and assure type safe to their fields and elements.








Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.44.1 IDynamicSymbolLoader Properties

The [IDynamicSymbolLoader](#) [▶ 2500] type exposes the following members.

Properties

	Name	Description
	BuildInTypes [▶ 2716]	Gets the build in types. (Inherited from ISymbolLoader [▶ 2714].)
	DataTypes [▶ 2720]	Gets the data types (Inherited from ISymbolServer [▶ 2719].)
	DefaultValueEncoding [▶ 2721]	Gets the default value encoding. (Inherited from ISymbolServer [▶ 2719].)
	RootNamespaceName [▶ 2718]	Gets the name of the root namespace (Inherited from ISymbolProvider [▶ 2717].)
	Settings [▶ 2716]	Gets or sets the access Method (Inherited from ISymbolLoader [▶ 2714].)
	Symbols [▶ 2721]	Gets the symbols. (Inherited from ISymbolServer [▶ 2719].)
	SymbolsDynamic [▶ 2501]	Gets the symbols (late bound as dynamic objects)

Reference

[IDynamicSymbolLoader Interface](#) [▶ 2500]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.44.1.1 IDynamicSymbolLoader.SymbolsDynamic Property

Gets the symbols (late bound as dynamic objects)

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDynamicSymbolsCollection SymbolsDynamic { get; }
```

Property Value

Type: [IDynamicSymbolsCollection](#) [[▶ 2503](#)]

The dynamic symbols.

Reference







[IDynamicSymbolLoader Interface](#) [[▶ 2500](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.44.2 IDynamicSymbolLoader Methods

The [IDynamicSymbolLoader](#) [[▶ 2500](#)] type exposes the following members.

Methods

	Name	Description
	GetDataTypesAsync [▶ 2722]	Gets the data types asynchronously. (Inherited from ISymbolServer [▶ 2719].)
	GetDynamicSymbolsAsync [▶ 2502]	Gets the dynamic symbols asynchronously
	GetSymbolsAsync [▶ 2723]	Gets the symbols asynchronously (Inherited from ISymbolServer [▶ 2719].)
	ResetCachedSymbolicData [▶ 2724]	Resets the cached symbolic data. (Inherited from ISymbolServer [▶ 2719].)
	TryGetDataTypes [▶ 2723]	Tries to get the symbols from the device target. (Inherited from ISymbolServer [▶ 2719].)
	TryGetSymbols [▶ 2724]	Tries to geth the symbols from the device target. (Inherited from ISymbolServer [▶ 2719].)

Reference

[IDynamicSymbolLoader Interface](#) [[▶ 2500](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.44.2.1 IDynamicSymbolLoader.GetDynamicSymbolsAsync Method

Gets the dynamic symbols asynchronously

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultDynamicSymbols> GetDynamicSymbolsAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: Task.ResultDynamicSymbols [▶ 2876].
Task<ResultDynamicSymbols>.

Reference

[IDynamicSymbolLoader Interface \[▶ 2500\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.45 IDynamicSymbolsCollection Interface

Interface IDynamicSymbolsContainer Implements the IDynamicMetaObjectProvider

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax

C#

```
public interface IDynamicSymbolsCollection : IDynamicMetaObjectProvider,
    IEnumerable<ISymbol>, IEnumerable
```

The IDynamicSymbolsCollection type exposes the following members.

Methods

	Name	Description
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.ISymbol [▶ 2691].)
	GetMetaObject	Returns the DynamicMetaObject responsible for binding operations performed on this object. (Inherited from IDynamicMetaObjectProvider.)

Reference



[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

System.Dynamic.IDynamicMetaObjectProvider

6.11.45.1 IDynamicSymbolsCollection Methods

The IDynamicSymbolsCollection [▶ 2503] type exposes the following members.

Methods

	Name	Description
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.ISymbol [▶ 2691].)
	GetMetaObject	Returns the DynamicMetaObject responsible for binding operations performed on this object. (Inherited from IDynamicMetaObjectProvider.)

Reference

[IDynamicSymbolsCollection Interface](#) [▶ 2503]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.46 IDynamicValue Interface

Interface IDynamicValue Implements the IDynamicMetaObjectProvider Implements the [IValue](#) [▶ 2745]
Implements the [IStructValue](#) [▶ 2678] Implements the [IArrayValue](#) [▶ 2465]

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229










Syntax

C#





```
public interface IDynamicValue : IDynamicMetaObjectProvider,
    IValue, IStructValue, IArrayValue
```









The IDynamicValue type exposes the following members.

Properties

	Name	Description
	Age [▶ 2746]	Gets the age of the value (last successful read of the value) (Inherited from IValue [▶ 2745].)
	CachedRaw [▶ 2746]	Gets the cached Raw internal Data. (Inherited from IValue [▶ 2745].)
	DataType [▶ 2747]	Gets the data type bound to this IValue [▶ 2745] (Inherited from IValue [▶ 2745].)
	IsPrimitive [▶ 2747]	Gets a value indicating whether this IValue [▶ 2745] is a primitive value. (Inherited from IValue [▶ 2745].)
	ParentValue [▶ 2506]	Gets the parent value.
	Symbol [▶ 2748]	Gets the symbol bound to this IValue [▶ 2745]. (Inherited from IValue [▶ 2745].)
	TimeStamp [▶ 2748]	Gets the Time stamp of the last successful read of the Value (local user time, UTC) (Inherited from IValue [▶ 2745].)
	UpdateMode [▶ 2748]	Gets the update mode (not implemented yet) (Inherited from IValue [▶ 2745].)
	ValueFactory [▶ 2506]	Gets the value factory.

Methods

	Name	Description
	GetMetaObject	Returns the DynamicMetaObject responsible for binding operations performed on this object. (Inherited from IDynamicMetaObjectProvider.)
	Read [▶ 2749]	Reads the value (via ADS) (Inherited from IValue [▶ 2745].)
	ReadAsync [▶ 2750]	Reads the value (via ADS) (Inherited from IValue [▶ 2745].)
	ResolveValue [▶ 2750]	Resolves the Value object to its primitive value. (Inherited from IValue [▶ 2745].)

	Name	Description
	TryGetArrayElement Values [▶ 2468]	Returns Array Element values. (Inherited from IArrayValue [▶ 2465] .)
	TryGetIndexValue [▶ 2468]	Tries to get the specified Array Element (Inherited from IArrayValue [▶ 2465] .)
	TryGetMemberValue [▶ 2680]	Tries to get a property/Member value. (Inherited from IStructValue [▶ 2678] .)
	TryResolveValue [▶ 2751]	Tries to resolves the Value object to its primitive value. (Inherited from IValue [▶ 2745] .)
	TrySetIndexValue [▶ 2469]	Tries to set the indexed value on Arrays (Inherited from IArrayValue [▶ 2465] .)
	TrySetMemberValue [▶ 2681]	Tries to Set a Member/Property Value (Inherited from IStructValue [▶ 2678] .)
	Write [▶ 2751]	Writes the value (via ADS) (Inherited from IValue [▶ 2745] .)
	WriteAsync [▶ 2751]	Writes the value (via ADS) (Inherited from IValue [▶ 2745] .)

Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

System.Dynamic.IDynamicMetaObjectProvider

[TwinCAT.TypeSystem.IValue \[▶ 2745\]](#)









[TwinCAT.TypeSystem.IStructValue \[▶ 2678\]](#)


[TwinCAT.TypeSystem.IArrayValue \[▶ 2465\]](#)

6.11.46.1 IDynamicValue Properties

The [IDynamicValue \[▶ 2504\]](#) type exposes the following members.

Properties

	Name	Description
	Age [▶ 2746]	Gets the age of the value (last successful read of the value) (Inherited from IValue [▶ 2745] .)
	CachedRaw [▶ 2746]	Gets the cached Raw internal Data. (Inherited from IValue [▶ 2745] .)
	DataType [▶ 2747]	Gets the data type bound to this IValue [▶ 2745] (Inherited from IValue [▶ 2745] .)
	IsPrimitive [▶ 2747]	Gets a value indicating whether this IValue [▶ 2745] is a primitive value. (Inherited from IValue [▶ 2745] .)
	ParentValue [▶ 2506]	Gets the parent value.
	Symbol [▶ 2748]	Gets the symbol bound to this IValue [▶ 2745] . (Inherited from IValue [▶ 2745] .)
	TimeStamp [▶ 2748]	Gets the Time stamp of the last successful read of the Value (local user time, UTC) (Inherited from IValue [▶ 2745] .)
	UpdateMode [▶ 2748]	Gets the update mode (not implemented yet) (Inherited from IValue [▶ 2745] .)

	Name	Description
	ValueFactory [▶ 2506]	Gets the value factory.

Reference

[IDynamicValue Interface](#) [[▶ 2504](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.46.1.1 IDynamicValue.ParentValue Property

Gets the parent value.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDynamicValue? ParentValue { get; }
```

Property Value

Type: [IDynamicValue](#) [[▶ 2504](#)]

The parent value.

Reference

[IDynamicValue Interface](#) [[▶ 2504](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.46.1.2 IDynamicValue.ValueFactory Property

Gets the value factory.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IAccessorValueFactory ValueFactory { get; }
```

Property Value

Type: [IAccessorValueFactory](#) [[▶ 3192](#)]

The value factory.

Reference













[IDynamicValue Interface](#) [[▶ 2504](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.46.2 IDynamicValue Methods

The [IDynamicValue](#) [[▶ 2504](#)] type exposes the following members.

Methods

	Name	Description
	GetMetaObject	Returns the DynamicMetaObject responsible for binding operations performed on this object. (Inherited from IDynamicMetaObjectProvider.)
	Read [▶ 2749]	Reads the value (via ADS) (Inherited from IValue [▶ 2745].)
	ReadAsync [▶ 2750]	Reads the value (via ADS) (Inherited from IValue [▶ 2745].)
	ResolveValue [▶ 2750]	Resolves the Value object to its primitive value. (Inherited from IValue [▶ 2745].)
	TryGetArrayElement Values [▶ 2468]	Returns Array Element values. (Inherited from IArrayValue [▶ 2465].)
	TryGetIndexValue [▶ 2468]	Tries to get the specified Array Element (Inherited from IArrayValue [▶ 2465].)
	TryGetMemberValue [▶ 2680]	Tries to get a property/Member value. (Inherited from IStructValue [▶ 2678].)
	TryResolveValue [▶ 2751]	Tries to resolves the Value object to its primitive value. (Inherited from IValue [▶ 2745].)
	TrySetIndexValue [▶ 2469]	Tries to set the indexed value on Arrays (Inherited from IArrayValue [▶ 2465].)
	TrySetMemberValue [▶ 2681]	Tries to Set a Member/Property Value (Inherited from IStructValue [▶ 2678].)
	Write [▶ 2751]	Writes the value (via ADS) (Inherited from IValue [▶ 2745].)
	WriteAsync [▶ 2751]	Writes the value (via ADS) (Inherited from IValue [▶ 2745].)

Reference

[IDynamicValue Interface \[\[▶ 2504\]\(#\)\]](#)

[TwinCAT.TypeSystem Namespace \[\[▶ 2083\]\(#\)\]](#)

6.11.47 IEnumType Interface

Common Enum type interface

Namespace: [TwinCAT.TypeSystem \[\[▶ 2083\]\(#\)\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax














C#

```
public interface IEnumType : IAliasType,
    IDataType, IBitSize
```








The IEnumType type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataType [▶ 2475] (Inherited from IDataType [▶ 2475].)
	BaseType [▶ 2439]	Gets the Base Type (Inherited from IAliasType [▶ 2436].)

	Name	Description
	BaseTypeName [▶ 2440]	Gets the BaseType name (Inherited from IAliasType [▶ 2436].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)
	EnumValues [▶ 2510]	Enumeration specification (if enum)
	FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from IDataType [▶ 2475].)
	Id [▶ 2480]	Gets the ID of the DataType (Inherited from IDataType [▶ 2475].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475].)
	Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Methods

	Name	Description
	Contains [▶ 2512]	Determines whether the enum values contains the specified name
	GetNames [▶ 2513]	Gets the filed names of the IEnumType.T. [▶ 2516]
	GetValues [▶ 2513]	Gets the values of the IEnumType.T. [▶ 2516]
	Parse [▶ 2514]	Parses a value name of the IEnumType.T. [▶ 2516] and returns the value (as base type)
	ToString [▶ 2514]	Returns a String that represents the specified value.
	TryParse(String, IConvertible.) [▶ 2515]	Parses the value from value name.
	TryParse(String, IEnumValue.) [▶ 2515]	Parses the value from value name.

Extension Methods

	Name	Description
	IsArrayOfPrimitives. [▶ 3025]	Overloaded.

	Name	Description
		Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer. [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive. [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve. [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]
















Also see about this

- ▶ [IDataType.IsContainer Property](#) [▶ 2480]
- ▶ [IDataType.IsPointer Property](#) [▶ 2481]
- ▶ [IDataType.IsPrimitive Property](#) [▶ 2481]
- ▶ [IDataType.IsReference Property](#) [▶ 2482]

6.11.47.1 IEnumType Properties

The [IEnumType \[▸ 2507\]](#) type exposes the following members.

Properties





	Name	Description
	Attributes [▸ 2478]	Gets the attributes of the IDataType [▸ 2475] (Inherited from IDataType [▸ 2475] .)
	BaseType [▸ 2439]	Gets the Base Type (Inherited from IAliasType [▸ 2436] .)
	BaseTypeName [▸ 2440]	Gets the BaseType name (Inherited from IAliasType [▸ 2436] .)
	BitSize [▸ 2473]	Gets the size of the IDataType [▸ 2475] in bits. (Inherited from IBitSize [▸ 2472] .)
	ByteSize [▸ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▸ 2472] .)
	Category [▸ 2478]	Gets the Data Type category (Inherited from IDataType [▸ 2475] .)
	Comment [▸ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▸ 2475] .)
	EnumValues [▸ 2510]	Enumeration specification (if enum)
	FullName [▸ 2479]	Gets the full name of the IDataType [▸ 2475] (Namespace + Name) (Inherited from IDataType [▸ 2475] .)
	Id [▸ 2480]	Gets the ID of the Data Type (Inherited from IDataType [▸ 2475] .)
	IsBitType [▸ 2474]	Gets a value indicating whether this instance is not basing on a full Data Type but instead of some sort of bit mapping (Inherited from IBitSize [▸ 2472] .)
	IsByteAligned [▸ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▸ 2472] .)
	Name [▸ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▸ 2475] .)
	Namespace [▸ 2482]	Gets the namespace string within the IDataType [▸ 2475] exists. (Inherited from IDataType [▸ 2475] .)
	Size [▸ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▸ 2474] (Inherited from IBitSize [▸ 2472] .)

Reference

[IEnumType Interface \[▸ 2507\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

Also see about this

-  [IDataType.IsContainer Property \[▸ 2480\]](#)
-  [IDataType.IsPointer Property \[▸ 2481\]](#)
-  [IDataType.IsPrimitive Property \[▸ 2481\]](#)
-  [IDataType.IsReference Property \[▸ 2482\]](#)

6.11.47.1.1 IEnumType.EnumValues Property

Enumeration specification (if enum)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IEnumValueCollection EnumValues { get; }
```

Property Value

Type: [IEnumValueCollection](#) [[▶ 2527](#)]

The enum specification.

Reference








[IEnumType Interface](#) [[▶ 2507](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.47.2 IEnumType Methods

The [IEnumType](#) [[▶ 2507](#)] type exposes the following members.

Methods

	Name	Description
	Contains [▶ 2512]	Determines whether the enum values contains the specified name
	GetNames [▶ 2513]	Gets the filed names of the IEnumType.T. [▶ 2516]
	GetValues [▶ 2513]	Gets the values of the IEnumType.T. [▶ 2516]
	Parse [▶ 2514]	Parses a value name of the IEnumType.T. [▶ 2516] and returns the value (as base type)
	ToString [▶ 2514]	Returns a String that represents the specified value.
	TryParse(String, IConvertible.) [▶ 2515]	Parses the value from value name.
	TryParse(String, IEnumValue.) [▶ 2515]	Parses the value from value name.

Extension Methods

	Name	Description
	IsArrayOfPrimitives. [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer. [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)

	Name	Description
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021] .)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021] .)
	IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021] .)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021] .)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021] .)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021] .)
	ResolvableAsPrimitive. [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021] .)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021] .)
	Resolve. [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021] .)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021] .)

Reference

[IEnumType Interface \[▶ 2507\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.47.2.1 IEnumType.Contains Method

Determines whether the enum values contains the specified name

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool Contains(
    string name
)
```


Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: Boolean
true if contains the value, otherwise, false.

Reference

[IEnumType Interface](#) [[▶ 2507](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.47.2.2 IEnumType.GetNames Method

Gets the filed names of the [IEnumType.T.](#) [[▶ 2516](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string[] GetNames()
```

Return Value

Type: .String.
System.String[].

Reference

[IEnumType Interface](#) [[▶ 2507](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.47.2.3 IEnumType.GetValues Method

Gets the values of the [IEnumType.T.](#) [[▶ 2516](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IConvertible[] GetValues()
```

Return Value

Type: .IConvertible.
T[].

Reference

[IEnumType Interface](#) [[▶ 2507](#)]

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.47.2.4 IEnumType.Parse Method

Parses a value name of the [IEnumType.T. \[▸ 2516\]](#) and returns the value (as base type)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IConvertible Parse(  
    string name  
)
```

Parameters

name	Type: System.String The value name.
------	--

Return Value

Type: IConvertible
T.

Reference

[IEnumType Interface \[▸ 2507\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.47.2.5 IEnumType.ToString Method

Returns a String that represents the specified value.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string ToString(  
    IConvertible val  
)
```

Parameters

val	Type: System.IConvertible The value.
-----	---

Return Value

Type: String
A String that represents this value.



Reference

[IEnumType Interface \[▸ 2507\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.47.2.6 IEnumType.TryParse Method

Overload List

	Name	Description
	TryParse(String, IConvertible.) [▶ 2515]	Parses the value from value name.
	TryParse(String, IEnumValue.) [▶ 2515]	Parses the value from value name.

Reference

[IEnumType Interface \[▶ 2507\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.47.2.6.1 IEnumType.TryParse Method (String, IConvertible.)

Parses the value from value name.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool TryParse(
    string name,
    out IConvertible?? value
)
```

Parameters

name	Type: System.String The value name.
value	Type: System.IConvertible. The value.

Return Value

Type: Boolean
true if value name was found, false otherwise.

Reference

[IEnumType Interface \[▶ 2507\]](#)

[TryParse Overload \[▶ 2515\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.47.2.6.2 IEnumType.TryParse Method (String, IEnumValue.)

Parses the value from value name.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool TryParse(
    string name,
    out IEnumValue?? value
)
```

Parameters

name	Type: System.String The value name.
value	Type: TwinCAT.TypeSystem.IEnumValue [▸ 2524] . The value.

Return Value

Type: Boolean
true if value name was found, false otherwise.

Reference

[IEnumType Interface \[▸ 2507\]](#)

[TryParse Overload \[▸ 2515\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.48 IEnumType.T. Interface

Interface representing an enum type

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#





```
public interface IEnumType<T> : IAliasType,
    IDataType, IBitSize
where T : IConvertible
```












Type Parameters

T	Base type of the Enum
---	-----------------------







The IEnumType.T. type exposes the following members.

Properties

	Name	Description
	Attributes [▸ 2478]	Gets the attributes of the IDataType [▸ 2475] (Inherited from IDataType [▸ 2475] .)
	BaseType [▸ 2439]	Gets the Base Type (Inherited from IAliasType [▸ 2436] .)
	BaseTypeName [▸ 2440]	Gets the BaseType name (Inherited from IAliasType [▸ 2436] .)
	BitSize [▸ 2473]	Gets the size of the IDataType [▸ 2475] in bits. (Inherited from IBitSize [▸ 2472] .)

	Name	Description
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)
	EnumValues [▶ 2519]	Enumeration specification (if enum)
	FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from IDataType [▶ 2475].)
	Id [▶ 2480]	Gets the ID of the DataType (Inherited from IDataType [▶ 2475].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475].)
	Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Methods

	Name	Description
	Contains [▶ 2521]	Determines whether the enum values contains the specified name
	GetNames [▶ 2521]	Gets the filed names of the IEnumType.T.
	GetValues [▶ 2522]	Gets the values of the IEnumType.T.
	Parse [▶ 2522]	Parses a name of the IEnumType.T. and returns the value (as base type)
	ToString [▶ 2523]	Returns a String that represents the specified value.
	TryParse [▶ 2523]	Parses the value from value name.

Extension Methods





	Name	Description
	IsArrayOfPrimitives. [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer. [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded.

	Name	Description
		Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive. [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve. [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]



Also see about this














-  [IDataType.IsContainer Property](#) [[▶ 2480](#)]
-  [IDataType.IsPointer Property](#) [[▶ 2481](#)]
-  [IDataType.IsPrimitive Property](#) [[▶ 2481](#)]
-  [IDataType.IsReference Property](#) [[▶ 2482](#)]

6.11.48.1 IEnumType.T. Properties

The [IEnumType.T.](#) [[▶ 2516](#)] generic type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataType [▶ 2475] (Inherited from IDataType [▶ 2475].)
	BaseType [▶ 2439]	Gets the Base Type (Inherited from IAliasType [▶ 2436].)





	Name	Description
	BaseTypeName [▶ 2440]	Gets the BaseType name (Inherited from IAliasType [▶ 2436].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)
	EnumValues [▶ 2519]	Enumeration specification (if enum)
	FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from IDataType [▶ 2475].)
	Id [▶ 2480]	Gets the ID of the DataType (Inherited from IDataType [▶ 2475].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475].)
	Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Reference

[IEnumType.T. Interface](#) [▶ 2516]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

Also see about this

-  [IDataType.IsContainer Property](#) [▶ 2480]
-  [IDataType.IsPointer Property](#) [▶ 2481]
-  [IDataType.IsPrimitive Property](#) [▶ 2481]
-  [IDataType.IsReference Property](#) [▶ 2482]

6.11.48.1.1 IEnumType.T..EnumValues Property

Enumeration specification (if enum)

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IEnumValueCollection EnumValues { get; }
```

Property Value

Type: [IEnumerable](#) [[2527](#)]

The enum specification.

Reference







[IEnumerable](#). Interface [[2516](#)]

[TwinCAT.TypeSystem](#) Namespace [[2083](#)]

6.11.48.2 IEnumerable.T. Methods

The [IEnumerable.T](#) [[2516](#)] generic type exposes the following members.

Methods

	Name	Description
	Contains [2521]	Determines whether the enum values contains the specified name
	GetNames [2521]	Gets the filed names of the IEnumerable.T [2516]
	GetValues [2522]	Gets the values of the IEnumerable.T [2516]
	Parse [2522]	Parses a name of the IEnumerable.T [2516] and returns the value (as base type)
	ToString [2523]	Returns a String that represents the specified value.
	TryParse [2523]	Parses the value from value name.

Extension Methods

	Name	Description
	IsArrayOfPrimitives [3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [3021].)
	IsArrayOfPrimitives(Boolean) [3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [3021].)
	IsContainer [3027]	Overloaded. Gets a value indicating whether this IDataType [2475] is a container type (Defined by DataTypeExtension [3021].)
	IsContainer(Boolean) [3028]	Overloaded. Gets a value indicating whether this IDataType [2475] is a container type (Defined by DataTypeExtension [3021].)
	IsPointer [3031]	Gets a value indicating whether this IDataType [2475] is a pointer type (Defined by DataTypeExtension [3021].)
	IsPrimitive [3033]	Overloaded. Gets a value indicating whether this IDataType [2475] is primitive (Defined by DataTypeExtension [3021].)
	IsPrimitive(Boolean) [3034]	Overloaded. Gets a value indicating whether this IDataType [2475] is primitive (Defined by DataTypeExtension [3021].)
	IsReference [3036]	Gets a value indicating whether this IDataType [2475] is a reference type (Defined by DataTypeExtension [3021].)

	Name	Description
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve . [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[IEnumType.T. Interface](#) [[▶ 2516](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.48.2.1 IEnumType.T..Contains Method

Determines whether the enum values contains the specified name

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool Contains(
    string name
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: Boolean
true if contains the value, otherwise, false.

Reference

[IEnumType.T. Interface](#) [[▶ 2516](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.48.2.2 IEnumType.T..GetNames Method

Gets the filed names of the [IEnumType.T.](#) [[▶ 2516](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string[] GetNames()
```

Return Value

Type: [.String](#).
[System.String\[\]](#).

Reference

[IEnumType.T. Interface](#) [[▶ 2516](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.48.2.3 IEnumType.T..GetValues Method

Gets the values of the [IEnumType.T.](#) [[▶ 2516](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
T[] GetValues()
```

Return Value

Type: [.T](#) [[▶ 2516](#)].
[T\[\]](#).

Reference

[IEnumType.T. Interface](#) [[▶ 2516](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.48.2.4 IEnumType.T..Parse Method

Parses a name of the [IEnumType.T.](#) [[▶ 2516](#)] and returns the value (as base type)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
T Parse(  
    string name  
)
```

Parameters

name	Type: System.String The name.
------	--

Return Value

Type: [T](#) [[▶ 2516](#)]

T.

Reference

[IEnumerable.T Interface](#) [[▶ 2516](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.48.2.5 IEnumType.T..ToString Method

Returns a String that represents the specified value.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string ToString(  
    T val  
)
```

Parameters

val	Type: T [▶ 2516] The value.
-----	--

Return Value

Type: String

A String that represents this value.

Reference

[IEnumerable.T Interface](#) [[▶ 2516](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.48.2.6 IEnumType.T..TryParse Method

Parses the value from value name.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool TryParse(  
    string name,  
    out T value  
)
```

Parameters

name	Type: System.String The value name.
------	--

value	Type: T [2516]. The value.
-------	---

Return Value

Type: Boolean
true if value name was found, false otherwise.

Reference

[IEnumType.T. Interface](#) [[2516](#)]

[TwinCAT.TypeSystem Namespace](#) [[2083](#)]

6.11.49 IEnumValue Interface

Generic interface for EnumValues

Namespace: [TwinCAT.TypeSystem](#) [[2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229






Syntax

C#

```
public interface IEnumValue
```

The IEnumValue type exposes the following members.

Properties

	Name	Description
	ManagedBaseType [2525]	Gets the enumeration base type (sint,byte,short,ushort,int,uint,Int64,UInt64 supported)
	Name [2525]	Gets the name of the Enumeration Value (value as string)
	Primitive [2526]	Gets the (Primitive, BaseType) Value of the enumeration as object
	RawValue [2526]	Gets the raw value of the enumeration (as byte array)
	Size [2526]	Gets the size of the Enum value (in bytes)



Reference




[TwinCAT.TypeSystem Namespace](#) [[2083](#)]

6.11.49.1 IEnumValue Properties

The [IEnumValue](#) [[2524](#)] type exposes the following members.

Properties

	Name	Description
	ManagedBaseType [2525]	Gets the enumeration base type (sint,byte,short,ushort,int,uint,Int64,UInt64 supported)
	Name [2525]	Gets the name of the Enumeration Value (value as string)

	Name	Description
	Primitive [▶ 2526]	Gets the (Primitive, BaseType) Value of the enumeration as object
	RawValue [▶ 2526]	Gets the raw value of the enumeration (as byte array)
	Size [▶ 2526]	Gets the size of the Enum value (in bytes)

Reference

[IEnumValue Interface](#) [[▶ 2524](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.49.1.1 IEnumValue.ManagedBaseType Property

Gets the enumeration base type (sint,byte,short,ushort,int,uint,Int64,UInt64 supported)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Type ManagedBaseType { get; }
```

Property Value

Type: Type

The type of the base.

Reference

[IEnumValue Interface](#) [[▶ 2524](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.49.1.2 IEnumValue.Name Property

Gets the name of the Enumeration Value (value as string)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string Name { get; }
```

Property Value

Type: String

The name.

Reference

[IEnumValue Interface](#) [[▶ 2524](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.49.1.3 IEnumValue.Primitive Property

Gets the (Primitive, BaseType) Value of the enumeration as object

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object Primitive { get; }
```

Property Value

Type: Object
The object value.

Reference

[IEnumValue Interface \[▸ 2524\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.49.1.4 IEnumValue.RawValue Property

Gets the raw value of the enumeration (as byte array)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
byte[] RawValue { get; }
```

Property Value

Type: .Byte.
The raw value.

Reference

[IEnumValue Interface \[▸ 2524\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.49.1.5 IEnumValue.Size Property

Gets the size of the Enum value (in bytes)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Size { get; }
```

Property Value

Type: Int32
The size.

Reference

[IEnumValue Interface |▸ 2524|](#)

[TwinCAT.TypeSystem Namespace |▸ 2083|](#)

6.11.50 IEnumValueCollection Interface

Interface IEnumValueCollection

Namespace: [TwinCAT.TypeSystem |▸ 2083|](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229




Syntax

C#









```
public interface IEnumValueCollection : IEnumValueCollection<IEnumValue, IConvertible>,
    ICollection<IEnumValue>, IEnumerable<IEnumValue>, IEnumerable
```




The IEnumValueCollection type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.IEnumValue ▸ 2524 ..)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.IEnumValue ▸ 2524 ..)
	Item ▸ 2530 	Gets or sets the element at the specified index. (Inherited from IEnumValueCollection.TEnumValue, TValue. ▸ 2529 .)

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.IEnumValue ▸ 2524 ..)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.IEnumValue ▸ 2524 ..)
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.IEnumValue ▸ 2524 ..)
	Contains(String) ▸ 2532 	Determines whether [contains] [the specified name]. (Inherited from IEnumValueCollection.TEnumValue, TValue. ▸ 2529 .)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.IEnumValue ▸ 2524 ..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.IEnumValue ▸ 2524 ..)
	GetNames ▸ 2533 	Gets the Value Names. (Inherited from IEnumValueCollection.TEnumValue, TValue. ▸ 2529 .)
	GetValues ▸ 2533 	Gets the values. (Inherited from IEnumValueCollection.TEnumValue, TValue. ▸ 2529 .)

	Name	Description
	Parse [▶ 2533]	Parses the specified name. (Inherited from IEnumValueCollection.TEnumValue, TValue. ▶ 2529 .)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.IEnumValue ▶ 2524 ..)
	TryParse(String, TValue.) [▶ 2535]	Tries to parse the string value of the Enum. (Inherited from IEnumValueCollection.TEnumValue, TValue. ▶ 2529 .)




Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.50.1 IEnumValueCollection Properties

The [IEnumValueCollection](#) [[▶ 2527](#)] type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.IEnumValue ▶ 2524 ..)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.IEnumValue ▶ 2524 ..)
	Item [▶ 2530]	Gets or sets the element at the specified index. (Inherited from IEnumValueCollection.TEnumValue, TValue. ▶ 2529 .)

Reference








[IEnumValueCollection Interface](#) [[▶ 2527](#)]





[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.50.2 IEnumValueCollection Methods

The [IEnumValueCollection](#) [[▶ 2527](#)] type exposes the following members.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.IEnumValue ▶ 2524 ..)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.IEnumValue ▶ 2524 ..)
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.IEnumValue ▶ 2524 ..)
	Contains(String) [▶ 2532]	Determines whether [contains] [the specified name]. (Inherited from IEnumValueCollection.TEnumValue, TValue. ▶ 2529 .)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.IEnumValue ▶ 2524 ..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.IEnumValue ▶ 2524 ..)
	GetNames [▶ 2533]	Gets the Value Names. (Inherited from IEnumValueCollection.TEnumValue, TValue. ▶ 2529 .)

	Name	Description
	GetValues [▶ 2533]	Gets the values. (Inherited from IEnumValueCollection.TEnumValue, TValue. [▶ 2529].)
	Parse [▶ 2533]	Parses the specified name. (Inherited from IEnumValueCollection.TEnumValue, TValue. [▶ 2529].)
	Remove	Removes the first occurrence of a specific object from the ICollection.T. (Inherited from ICollection.IEnumValue [▶ 2524].)
	TryParse(String, TValue.) [▶ 2535]	Tries to parse the string value of the Enum. (Inherited from IEnumValueCollection.TEnumValue, TValue. [▶ 2529].)

Reference

[IEnumValueCollection Interface](#) [[▶ 2527](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.51 IEnumValueCollection.TEnumValue, TValue. Interface

Interface for collections of [IEnumValues](#) [[▶ 2524](#)]. Implements the [ICollection.T.](#)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#




```
public interface IEnumValueCollection<TEnumValue, TValue> : ICollection<TEnumValue>,
    IEnumerable<TEnumValue>, IEnumerable
where TEnumValue : class, IEnumValue
where TValue : IConvertible
```

Type Parameters




TEnumValue	TValue
------------	--------










The [IEnumValueCollection.TEnumValue, TValue.](#) type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T. (Inherited from ICollection.TEnumValue.)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.TEnumValue.)
	Item [▶ 2530]	Gets or sets the element at the specified index.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T. (Inherited from ICollection.TEnumValue.)
	Clear	Removes all items from the ICollection.T. (Inherited from ICollection.TEnumValue.)
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.TEnumValue.)

	Name	Description
	Contains(String) [▶ 2532]	Determines whether [contains] [the specified name].
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.TEnumValue..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.TEnumValue..)
	GetNames [▶ 2533]	Gets the Value Names.
	GetValues [▶ 2533]	Gets the values.
	Parse [▶ 2533]	Parses the specified name.
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.TEnumValue..)
	TryParse(String, TEnumValue.) [▶ 2534]	Tries to parse the string value of the Enum.
	TryParse(String, TValue.) [▶ 2535]	Tries to parse the string value of the Enum.

Reference




[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

System.Collections.Generic.ICollection.T.

6.11.51.1 IEnumValueCollection.TEnumValue, TValue. Properties

The [IEnumValueCollection.TEnumValue, TValue.](#) [▶ 2529] generic type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.TEnumValue [▶ 2529].)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.TEnumValue [▶ 2529].)
	Item [▶ 2530]	Gets or sets the element at the specified index.

Reference

[IEnumValueCollection.TEnumValue, TValue. Interface](#) [▶ 2529]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.51.1.1 IEnumValueCollection.TEnumValue, TValue..Item Property

Gets or sets the element at the specified index.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
TValue this[
    string name
] { get; }
```

Parameters

name	Type: System.String The name of the value
------	--

Return Value

Type: [TValue](#) [[▶ 2529](#)]
EnumValue<T>.

Exceptions

Exception	Condition
NotImplementedException	
NotImplementedException	

Reference











[IEnumerable.TEnumValue, TValue. Interface](#) [[▶ 2529](#)]



[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.51.2 IEnumValueCollection.TEnumValue, TValue. Methods

The [IEnumerable.TEnumValue, TValue.](#) [[▶ 2529](#)] generic type exposes the following members.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.TEnumValue [▶ 2529]..)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.TEnumValue [▶ 2529]..)
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.TEnumValue [▶ 2529]..)
	Contains(String) [▶ 2532]	Determines whether [contains] [the specified name].
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.TEnumValue [▶ 2529]..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.TEnumValue [▶ 2529]..)
	GetNames [▶ 2533]	Gets the Value Names.
	GetValues [▶ 2533]	Gets the values.
	Parse [▶ 2533]	Parses the specified name.
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.TEnumValue [▶ 2529]..)

	Name	Description
	<code>TryParse(String, TEnumValue.)</code> [▶ 2534]	Tries to parse the string value of the Enum.
	<code>TryParse(String, TValue.)</code> [▶ 2535]	Tries to parse the string value of the Enum.



Reference

[IEnumerable{T}.TEnumValue, TValue. Interface \[▶ 2529\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.51.2.1 IEnumerable{T}.TEnumValue, TValue..Contains Method

Overload List

	Name	Description
	<code>Contains(T)</code>	Determines whether the ICollection{T}.T. contains a specific value. (Inherited from ICollection{T}.TEnumValue [▶ 2529].)
	<code>Contains(String)</code> [▶ 2532]	Determines whether [contains] [the specified name].

Reference

[IEnumerable{T}.TEnumValue, TValue. Interface \[▶ 2529\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.51.2.1.1 IEnumerable{T}.TEnumValue, TValue..Contains Method (String)

Determines whether [contains] [the specified name].

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool Contains(
    string value
)
```

Parameters

value	Type: System.String Value
-------	------------------------------

Return Value

Type: Boolean
true if [contains] [the specified name]; otherwise, false.

Reference

[IEnumerable{T}.TEnumValue, TValue. Interface \[▶ 2529\]](#)

[Contains Overload \[▶ 2532\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.51.2.2 IEnumValueCollection.TEnumValue, TValue..GetNames Method

Gets the Value Names.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string[] GetNames()
```

Return Value

Type: .String.
System.String[].

Reference

[IEnumValueCollection.TEnumValue, TValue. Interface \[▶ 2529\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.51.2.3 IEnumValueCollection.TEnumValue, TValue..GetValues Method

Gets the values.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
TValue[] GetValues()
```

Return Value

Type: .TValue [\[▶ 2529\]](#).
T[].

Reference

[IEnumValueCollection.TEnumValue, TValue. Interface \[▶ 2529\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.51.2.4 IEnumValueCollection.TEnumValue, TValue..Parse Method

Parses the specified name.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
TValue Parse(
    string name
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: [TValue](#) [[▶ 2529](#)]



T.

Reference

[IEnumerable.TEnumValue, TValue. Interface](#) [[▶ 2529](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.51.2.5 IEnumerable.TEnumValue, TValue..TryParse Method**Overload List**

	Name	Description
	TryParse(String, TEnumValue.) [▶ 2534]	Tries to parse the string value of the Enum.
	TryParse(String, TValue.) [▶ 2535]	Tries to parse the string value of the Enum.

Reference

[IEnumerable.TEnumValue, TValue. Interface](#) [[▶ 2529](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.51.2.5.1 IEnumerable.TEnumValue, TValue..TryParse Method (String, TEnumValue.)

Tries to parse the string value of the Enum.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
bool TryParse(
    string strValue,
    out TEnumValue? value
)
```

Parameters

strValue	Type: System.String The Value in string representation.
value	Type: TEnumValue [▶ 2529]. The value.

Return Value

Type: Boolean
true if the string could be parsed successfully, false otherwise.

Reference

[IEnumerable{TEnumValue, TValue}. Interface \[▶ 2529\]](#)

[TryParse Overload \[▶ 2534\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.51.2.5.2 IEnumValueCollection.TEnumValue, TValue..TryParse Method (String, TValue.)

Tries to parse the string value of the Enum.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
bool TryParse(
    string strValue,
    out TValue value
)
```

Parameters

strValue	Type: System.String The Value in string representation.
value	Type: TValue [▶ 2529]. The value.

Return Value

Type: Boolean
true if the string was parsed successfully, false otherwise.

Reference

[IEnumerable{TEnumValue, TValue}. Interface \[▶ 2529\]](#)

[TryParse Overload \[▶ 2534\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.52 IField Interface

Specifies a single field/member of a [Struct DataType \[▶ 2671\]](#).

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public interface IField : IAttributedInstance, IInstance,
    IBitSize
```

The IField type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full IDataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	ParentType [▶ 2538]	Gets the Parent Struct/Union of this IField.
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	TypeName [▶ 2553]	Gets the name of the IDataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

















Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.52.1 IField Properties

The [IField](#) [[▶ 2535](#)] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full IDataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($BitSize \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	ParentType [▶ 2538]	Gets the Parent Struct/Union of this IField [▶ 2535].
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	TypeName [▶ 2553]	Gets the name of the IDataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Reference[IField Interface \[► 2535\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.52.1 IField.ParentType Property**Gets the Parent Struct/Union of this [IField \[► 2535\]](#).**Namespace:** [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
IDataType? ParentType { get; }
```

Property ValueType: [IDataType \[► 2475\]](#)

The type of the parent.

Reference[IField Interface \[► 2535\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.52.2 IField Methods**The [IField \[► 2535\]](#) type exposes the following members.**Extension Methods**






	Name	Description
	ReferencesExternalData [► 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [► 3021] .)

Reference[IField Interface \[► 2535\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.53 IFieldCollection Interface**Interface [IFieldCollection](#) Implements the [IInstanceCollection.T. \[► 2554\]](#)**Namespace:** [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**
















```
public interface IFieldCollection : IInstanceCollection<IField>,
    IList<IField>, ICollection<IField>, IEnumerable<IField>,
    IEnumerable
```


The IFieldCollection type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.IField ▶ 2535 ..)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.IField ▶ 2535 ..)
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.IField ▶ 2535 ..)
	Item.String. ▶ 2556	Gets the Instance ▶ 2549 with the specified instance path. (Inherited from InstanceCollection.T. ▶ 2554 .)
	Mode ▶ 2556	Gets the InstanceCollectionMode ▶ 2573 . (Inherited from InstanceCollection.T. ▶ 2554 .)

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.IField ▶ 2535 ..)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.IField ▶ 2535 ..)
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.IField ▶ 2535 ..)
	Contains(String) ▶ 2558	Determines whether this collection contains an instance with the specified instance path. (Inherited from InstanceCollection.T. ▶ 2554 .)
	ContainsName ▶ 2558	Determines whether this collection contains an instance with the specified instance name. (Inherited from InstanceCollection.T. ▶ 2554 .)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.IField ▶ 2535 ..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.IField ▶ 2535 ..)
	GetInstance ▶ 2559	Gets the Instance ▶ 2549 by instance path. (Inherited from InstanceCollection.T. ▶ 2554 .)
	GetInstanceByName ▶ 2559	Gets the Instance ▶ 2549 by instance name. (Inherited from InstanceCollection.T. ▶ 2554 .)
	IndexOf	Determines the index of a specific item in the IList.T.. (Inherited from IList.IField ▶ 2535 ..)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.IField ▶ 2535 ..)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.IField ▶ 2535 ..)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.IField ▶ 2535 ..)
	TryGetInstance ▶ 2560	Tries to get the specified instance. (Inherited from InstanceCollection.T. ▶ 2554 .)
	TryGetInstanceByName ▶ 2560	Tries to get the specified instance by name. (Inherited from InstanceCollection.T. ▶ 2554 .)

	Name	Description
	TryGetMember [▶ 2541]	Gets the Field/Member with the specified name from the collection.

Reference






[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

[TwinCAT.TypeSystem.InstanceCollection.T.](#) [▶ 2554]

6.11.53.1 IFieldCollection Properties

The [IFieldCollection](#) [▶ 2538] type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.IField [▶ 2535].)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.IField [▶ 2535].)
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.IField [▶ 2535].)
	Item.String. [▶ 2556]	Gets the Instance [▶ 2549] with the specified instance path. (Inherited from InstanceCollection.T. [▶ 2554].)
	Mode [▶ 2556]	Gets the InstanceCollectionMode [▶ 2573]. (Inherited from InstanceCollection.T. [▶ 2554].)

Reference







[IFieldCollection Interface](#) [▶ 2538]











[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.53.2 IFieldCollection Methods

The [IFieldCollection](#) [▶ 2538] type exposes the following members.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.IField [▶ 2535].)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.IField [▶ 2535].)
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.IField [▶ 2535].)
	Contains(String) [▶ 2558]	Determines whether this collection contains an instance with the specified instance path. (Inherited from InstanceCollection.T. [▶ 2554].)
	ContainsName [▶ 2558]	Determines whether this collection contains an instance with the specified instance name. (Inherited from InstanceCollection.T. [▶ 2554].)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.IField [▶ 2535].)

	Name	Description
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.IField [▶ 2535] ..)
	GetInstance [▶ 2559]	Gets the Instance [▶ 2549] by instance path. (Inherited from InstanceCollection.T. [▶ 2554] .)
	GetInstanceByName [▶ 2559]	Gets the Instance [▶ 2549] by instance name. (Inherited from InstanceCollection.T. [▶ 2554] .)
	IndexOf	Determines the index of a specific item in the IList.T. (Inherited from IList.IField [▶ 2535] ..)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.IField [▶ 2535] ..)
	Remove	Removes the first occurrence of a specific object from the ICollection.T. (Inherited from ICollection.IField [▶ 2535] ..)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.IField [▶ 2535] ..)
	TryGetInstance [▶ 2560]	Tries to get the specified instance. (Inherited from InstanceCollection.T. [▶ 2554] .)
	TryGetInstanceByName [▶ 2560]	Tries to get the specified instance by name. (Inherited from InstanceCollection.T. [▶ 2554] .)
	TryGetMember [▶ 2541]	Gets the Field/Member with the specified name from the collection.

Reference

[IFieldCollection Interface \[▶ 2538\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.53.2.1 IFieldCollection.TryGetMember Method

Gets the Field/Member with the specified name from the collection.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool TryGetMember(
    string fieldName,
    out IField?? symbol
)
```

Parameters

fieldName	Type: System.String Name of the field/member.
symbol	Type: TwinCAT.TypeSystem.IField [▶ 2535] . The symbol with the specified name.

Return Value

Type: Boolean
true if found, false otherwise.

Reference

[IFieldCollection Interface](#) [► 2538]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.54 IGenericTypeMarshaler Interface

Interface IGenericTypeMarshaler Implements the [ITypeMarshaler](#) [► 2731]

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229







Syntax

C#

```
public interface IGenericTypeMarshaler : ITypeMarshaler
```

The IGenericTypeMarshaler type exposes the following members.

Methods

	Name	Description
	CanMarshal [► 2732]	Determines whether ADS can marshal the specified managed data type. (Inherited from ITypeMarshaler [► 2731].)
	CanMarshalValue [► 2736]	Determines whether ADS can marshal the specified value (Inherited from ITypeMarshaler [► 2731].)
	Marshal [► 2734]	Marshals the specified value to the specified destination memory / span. (Inherited from ITypeMarshaler [► 2731].)
	MarshalValueSize [► 2736]	Gets the byte size of the value when marshalled. (Inherited from ITypeMarshaler [► 2731].)
	Unmarshal(Type, ReadOnlySpan<Byte>, Encoding, Object.) [► 2735]	Unmarshals the specified managed type from memory / span (Inherited from ITypeMarshaler [► 2731].)
	Unmarshal.T.(ReadOnlySpan<Byte>, Encoding, T.) [► 2544]	Unmarshals the source data to an managed value of the specified type.

Remarks




The IGenericTypeMarshaler extends the [ITypeMarshaler](#) [► 2731] by generic methods.

Reference

[TwinCAT.TypeSystem Namespace](#) [► 2083]

[TwinCAT.TypeSystem.ITypeMarshaler](#) [► 2731]







Also see about this

-  [ITypeMarshaler.CanMarshal Method \(Object\)](#) [► 2733]
-  [ITypeMarshaler.CanMarshal Method \(Type\)](#) [► 2733]
-  [ITypeMarshaler.MarshalSize Method](#) [► 2735]

6.11.54.1 IGenericTypeMarshaler Methods

The [IGenericTypeMarshaler](#) [▶ 2542] type exposes the following members.

Methods




	Name	Description
	CanMarshal [▶ 2732]	Determines whether ADS can marshal the specified managed data type. (Inherited from ITypeMarshaler [▶ 2731].)
	CanMarshalValue [▶ 2736]	Determines whether ADS can marshal the specified value (Inherited from ITypeMarshaler [▶ 2731].)
	Marshal [▶ 2734]	Marshals the specified value to the specified destination memory / span. (Inherited from ITypeMarshaler [▶ 2731].)
	MarshalValueSize [▶ 2736]	Gets the byte size of the value when marshalled. (Inherited from ITypeMarshaler [▶ 2731].)
	Unmarshal(Type, ReadOnlySpan.Byte, Encoding, Object.) [▶ 2735]	Unmarshals the specified managed type from memory / span (Inherited from ITypeMarshaler [▶ 2731].)
	Unmarshal.T (ReadOnlySpan.Byte, Encoding, T.) [▶ 2544]	Unmarshals the source data to an managed value of the specified type.

Reference

[IGenericTypeMarshaler Interface](#) [▶ 2542]



[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

Also see about this

-  [ITypeMarshaler.CanMarshal Method \(Object\)](#) [▶ 2733]
-  [ITypeMarshaler.CanMarshal Method \(Type\)](#) [▶ 2733]
-  [ITypeMarshaler.MarshalSize Method](#) [▶ 2735]

6.11.54.1.1 IGenericTypeMarshaler.Unmarshal Method

Overload List

	Name	Description
	Unmarshal.T (ReadOnlySpan.Byte, Encoding, T.) [▶ 2544]	Unmarshals the source data to an managed value of the specified type.
	Unmarshal(Type, ReadOnlySpan.Byte, Encoding, Object.) [▶ 2735]	Unmarshals the specified managed type from memory / span (Inherited from ITypeMarshaler [▶ 2731].)

Reference

[IGenericTypeMarshaler Interface](#) [▶ 2542]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.54.1.1.1 IGenericTypeMarshaler.Unmarshal.T. Method (ReadOnlySpan`1, Void, Byte)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
int Unmarshal<T>(
    ReadOnlySpan source,
    void encoding,
    byte value
)
```

Parameters

source	Type: ReadOnlySpan
encoding	Type: System.Void
value	Type: System.Byte

Type Parameters

T

Return Value

Type: [Int32](#)

Reference

[IGenericTypeMarshaler Interface](#) [[▶ 2542](#)]

[Unmarshal Overload](#) [[▶ 2543](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.54.1.1.2 IGenericTypeMarshaler.Unmarshal.T. Method (ReadOnlySpan.Byte., Encoding, T.)

Unmarshals the source data to an managed value of the specified type.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Unmarshal<T>(
    ReadOnlySpan<byte> source,
    Encoding encoding,
    out T value
)
```


Parameters

source	Type: System.ReadOnlySpan.Byte. The source data.
encoding	Type: System.Text.Encoding The encoding.
value	Type: T. The value.

Type Parameters

T	The values type.
---	------------------

Return Value

Type: Int32
Number of unmarshalled bytes.

Reference

- [IGenericTypeMarshaler Interface \[▶ 2542\]](#)
- [Unmarshal Overload \[▶ 2543\]](#)
- [TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.55 IHierarchicalSymbol Interface

Bindable Symbol interface (for internal use only)

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229








Syntax

C#

```
public interface IHierarchicalSymbol : ISymbol,
    IAttributedInstance, IInstance, IBitSize
```


The IHierarchicalSymbol type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469] .)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472] .)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472] .)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691] .)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549] .)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549] . (Inherited from IInstance [▶ 2549] .)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549] .)

	Name	Description
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is a container type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Methods

	Name	Description
	SetParent [▶ 2548]	Sets the parent of the Symbol

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference


[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.55.1 IHierarchicalSymbol Properties

The [IHierarchicalSymbol](#) [[▶ 2545](#)] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)

	Name	Description
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469] .)

Reference


[IHierarchicalSymbol Interface](#) [\[▶ 2545\]](#)

[TwinCAT.TypeSystem Namespace](#) [\[▶ 2083\]](#)

6.11.55.2 IHierarchicalSymbol Methods

The [IHierarchicalSymbol](#) [\[▶ 2545\]](#) type exposes the following members.

Methods

	Name	Description
	SetParent [▶ 2548]	Sets the parent of the Symbol

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021] .)

Reference

[IHierarchicalSymbol Interface](#) [\[▶ 2545\]](#)

[TwinCAT.TypeSystem Namespace](#) [\[▶ 2083\]](#)

6.11.55.2.1 IHierarchicalSymbol.SetParent Method

Sets the parent of the Symbol

Namespace: [TwinCAT.TypeSystem](#) [\[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void SetParent(
    ISymbol symbol
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol.
--------	--

Reference

[IHierarchicalSymbol Interface](#) [\[▶ 2545\]](#)

[TwinCAT.TypeSystem Namespace](#) [\[▶ 2083\]](#)

6.11.56 Instance Interface

Interface specifying instance objects.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229














Syntax

C#

```
public interface IInstance : IBitSize
```

The IInstance type exposes the following members.

Properties

	Name	Description
	BitSize [▸ 2473]	Gets the size of the IDataType [▸ 2475] in bits. (Inherited from IBitSize [▸ 2472].)
	ByteSize [▸ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▸ 2472].)
	Comment [▸ 2550]	Gets the comment of the IInstance
	DataType [▸ 2551]	Gets the IDataType [▸ 2475] of the IInstance.
	InstanceName [▸ 2551]	Gets the name of the instance (without periods (.))
	InstancePath [▸ 2552]	Gets the relative / absolute access path to the instance (with periods (.))
	IsBitType [▸ 2474]	Gets a value indicating whether this instance is not basing on a full IDataType but instead of some sort of bit mapping (Inherited from IBitSize [▸ 2472].)
	IsByteAligned [▸ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▸ 2472].)
	IsPointer [▸ 2552]	Indicates that the IInstance represents a Pointer type (Pointer TO)
	IsReference [▸ 2552]	Indicates that the IInstance represents a Reference type (REFERENCE TO)
	IsStatic [▸ 2553]	Gets a value indicating whether this IInstance is static.
	Size [▸ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▸ 2474] (Inherited from IBitSize [▸ 2472].)
	TypeName [▸ 2553]	Gets the name of the IDataType [▸ 2475] that is used for this IInstance.

Extension Methods

	Name	Description
	ReferencesExternalData [▸ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▸ 3021].)














Reference

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.56.1 Instance Properties

The [IInstance](#) [[▶ 2549](#)] type exposes the following members.

Properties

	Name	Description
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549]
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549].
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.))
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.))
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static.
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549].

Reference

[IInstance Interface](#) [[▶ 2549](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.56.1.1 Instance.Comment Property

Gets the comment of the [IInstance](#) [[▶ 2549](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string Comment { get; }
```

Property Value

Type: String
The comment.

Reference

[IInstance Interface \[▸ 2549\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.56.1.2 Instance.DataType Property

Gets the [IDataType \[▸ 2475\]](#) of the [IInstance \[▸ 2549\]](#).

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDataType? DataType { get; }
```

Property Value

Type: [IDataType \[▸ 2475\]](#)
The type of the data.

Remarks

The DataType can be unresolved in rare circumstances and therefore could have value **null**.

Reference

[IInstance Interface \[▸ 2549\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.56.1.3 Instance.InstanceName Property

Gets the name of the instance (without periods (.))

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string InstanceName { get; }
```

Property Value

Type: String
The name of the instance.

Reference

[IInstance Interface \[▸ 2549\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.56.1.4 Instance.InstancePath Property

Gets the relative / absolute access path to the instance (with periods (..))

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string InstancePath { get; }
```

Property Value

Type: String

The instance path.

Remarks

If this path is relative or absolute depends on the context. [IMember](#) [[▶ 2561](#)] are using relative paths, [ISymbol](#) [[▶ 2691](#)]s are using absolute ones.

Reference

[IInstance Interface](#) [[▶ 2549](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.56.1.5 Instance.IsPointer Property

Indicates that the [IInstance](#) [[▶ 2549](#)] represents a Pointer type (Pointer TO)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsPointer { get; }
```

Property Value

Type: Boolean

true if is ReferenceTo, otherwise false.

Reference

[IInstance Interface](#) [[▶ 2549](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.56.1.6 Instance.IsReference Property

Indicates that the [IInstance](#) [[▶ 2549](#)] represents a Reference type (REFERENCE TO)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsReference { get; }
```

Property Value

Type: Boolean
true if is ReferenceTo, otherwise false.

Reference

[IInstance Interface](#) [► 2549]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.56.1.7 IInstance.IsStatic Property

Gets a value indicating whether this [IInstance](#) [► 2549] is static.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsStatic { get; }
```

Property Value

Type: Boolean
true if this instance is static; otherwise, false.

Reference

[IInstance Interface](#) [► 2549]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.56.1.8 IInstance.TypeName Property

Gets the name of the [DataType](#) [► 2475] that is used for this [IInstance](#) [► 2549].

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string TypeName { get; }
```

Property Value

Type: String
The name of the type.

Reference

[IInstance Interface](#) [► 2549]

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.56.2 Instance Methods

The [IInstance \[▸ 2549\]](#) type exposes the following members.

Extension Methods

	Name	Description
	ReferencesExternalData [▸ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▸ 3021] .)

Reference

[IInstance Interface \[▸ 2549\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.57 InstanceCollection.T. Interface

Generic InstanceCollection interface.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#






```
public interface IInstanceCollection<T> : IList<T>,
    ICollection<T>, IEnumerable<T>, IEnumerable
where T : class, IInstance
```

Type Parameters


T















The [IInstanceCollection.T.](#) type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T. (Inherited from ICollection.T.)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.T.)
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.T.)
	Item.String. [▸ 2556]	Gets the IInstance [▸ 2549] with the specified instance path.
	Mode [▸ 2556]	Gets the InstanceCollectionMode [▸ 2573] .

Methods

	Name	Description
	Add	Adds an item to the ICollection.T. (Inherited from ICollection.T.)

	Name	Description
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.T..)
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.T..)
	<u>Contains(String)</u> [▶ 2558]	Determines whether this collection contains an instance with the specified instance path.
	<u>ContainsName</u> [▶ 2558]	Determines whether this collection contains an instance with the specified instance name.
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.T..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.T..)
	<u>GetInstance</u> [▶ 2559]	Gets the <u>Instance</u> [▶ 2549] by instance path.
	<u>GetInstanceByName</u> [▶ 2559]	Gets the <u>Instance</u> [▶ 2549] by instance name.
	IndexOf	Determines the index of a specific item in the IList.T.. (Inherited from IList.T..)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.T..)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.T..)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.T..)
	<u>TryGetInstance</u> [▶ 2560]	Tries to get the specified instance.
	<u>TryGetInstanceByName</u> [▶ 2560]	Tries to get the specified instance by name.






Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.57.1 InstanceCollection.T. Properties

The InstanceCollection.T. [▶ 2554] generic type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.T [▶ 2554]..)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.T [▶ 2554]..)
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.T [▶ 2554]..)
	<u>Item.String.</u> [▶ 2556]	Gets the <u>Instance</u> [▶ 2549] with the specified instance path.
	<u>Mode</u> [▶ 2556]	Gets the <u>InstanceCollectionMode</u> [▶ 2573].



Reference

[InstanceCollection.T. Interface \[▶ 2554\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.57.1.1 InstanceCollection.T..Item Property

Overload List

	Name	Description
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.T [▸ 2554]..)
	Item.String. [▸ 2556]	Gets the IInstance [▸ 2549] with the specified instance path.

Reference

[IInstanceCollection.T. Interface \[▸ 2554\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.57.1.1.1 IInstanceCollection.T..Item Property (String)

Gets the [IInstance](#) [[▸ 2549](#)] with the specified instance path.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
T this[
    string instancePath
] { get; }
```

Parameters

instancePath	Type: System.String
--------------	---------------------

Property Value

Type: [T](#) [[▸ 2554](#)]

Reference

[IInstanceCollection.T. Interface \[▸ 2554\]](#)

[Item Overload \[▸ 2556\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.57.1.2 IInstanceCollection.T..Mode Property

Gets the [InstanceCollectionMode](#) [[▸ 2573](#)].

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
InstanceCollectionMode Mode { get; }
```

Property Value

Type: [InstanceCollectionMode](#) [▶ 2573]

The mode.

Reference
















[ICollection.T. Interface](#) [▶ 2554]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]



6.11.57.2 InstanceCollection.T. Methods

The [ICollection.T.](#) [▶ 2554] generic type exposes the following members.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.T [▶ 2554].)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.T [▶ 2554].)
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.T [▶ 2554].)
	Contains(String) [▶ 2558]	Determines whether this collection contains an instance with the specified instance path.
	ContainsName [▶ 2558]	Determines whether this collection contains an instance with the specified instance name.
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.T [▶ 2554].)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.T [▶ 2554].)
	GetInstance [▶ 2559]	Gets the Instance [▶ 2549] by instance path.
	GetInstanceByName [▶ 2559]	Gets the Instance [▶ 2549] by instance name.
	IndexOf	Determines the index of a specific item in the IList.T.. (Inherited from IList.T [▶ 2554].)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.T [▶ 2554].)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.T [▶ 2554].)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.T [▶ 2554].)
	TryGetInstance [▶ 2560]	Tries to get the specified instance.
	TryGetInstanceByName [▶ 2560]	Tries to get the specified instance by name.

Reference[ICollection.T. Interface](#) [► 2554][TwinCAT.TypeSystem Namespace](#) [► 2083]**6.11.57.2.1 InstanceCollection.T..Contains Method****Overload List**

	Name	Description
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.T [► 2554]..)
	Contains(String) [► 2558]	Determines whether this collection contains an instance with the specified instance path.

Reference[ICollection.T. Interface](#) [► 2554][TwinCAT.TypeSystem Namespace](#) [► 2083]**6.11.57.2.1.1 InstanceCollection.T..Contains Method (String)**

Determines whether this collection contains an instance with the specified instance path.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
bool Contains(
    string instancePath
)
```

Parameters

instancePath	Type: System.String The instance path.
--------------	---

Return Value

Type: Boolean

true if this collection contains the specified instance path; otherwise, false.

Reference[ICollection.T. Interface](#) [► 2554][Contains Overload](#) [► 2558][TwinCAT.TypeSystem Namespace](#) [► 2083]**6.11.57.2.2 InstanceCollection.T..ContainsName Method**

Determines whether this collection contains an instance with the specified instance name.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool ContainsName(  
    string instanceName  
)
```

Parameters

instanceName	Type: System.String Name of the instance.
--------------	--

Return Value

Type: Boolean

true if this collection contains the specified instance path; otherwise, false.

Reference

[IInstanceCollection.T. Interface](#) [► 2554]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.57.2.3 IInstanceCollection.T..GetInstance Method

Gets the [IInstance](#) [► 2549] by instance path.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
T GetInstance(  
    string instancePath  
)
```

Parameters

instancePath	Type: System.String The instance path.
--------------	---

Return Value

Type: [T](#) [► 2554]

T.

Reference

[IInstanceCollection.T. Interface](#) [► 2554]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.57.2.4 IInstanceCollection.T..GetInstanceByName Method

Gets the [IInstance](#) [► 2549] by instance name.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```

IList<T> GetInstanceByName (
    string instanceName
)

```

Parameters

instanceName	Type: System.String Name of the instance.
--------------	--

Return Value

Type: [IList.T](#) [[▶ 2554](#)].
IList<T>.

Reference

[InstanceCollection.T. Interface](#) [[▶ 2554](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.57.2.5 InstanceCollection.T..TryGetInstance Method

Tries to get the specified instance.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```

bool TryGetInstance (
    string instancePath,
    out T? symbol
)

```

Parameters

instancePath	Type: System.String The instance path.
symbol	Type: T [▶ 2554]. The symbol.

Return Value

Type: Boolean
true if the [Instance](#) [[▶ 2549](#)] is found; otherwise, false

Reference

[InstanceCollection.T. Interface](#) [[▶ 2554](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.57.2.6 InstanceCollection.T..TryGetInstanceByName Method

Tries to get the specified instance by name.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool TryGetInstanceByName(
    string instanceName,
    out IList<T>? symbols
)
```

Parameters

instanceName	Type: System.String Name of the instance.
symbols	Type: System.Collections.Generic.IList.T [▶ 2554]. The found symbols.

Return Value

Type: Boolean

true if the [IInstance](#) [[▶ 2549](#)] is found; otherwise, false

Reference

[IInstanceCollection.T. Interface](#) [[▶ 2554](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.58 IMember Interface

Specifies a single field/member of a [Struct DataType](#) [[▶ 2671](#)].

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229







Syntax














C#

```
public interface IMember : IField,
    IAttributedInstance, IInstance, IBitSize
```

The IMember type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitOffset [▶ 2563]	Gets the bit offset.
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteOffset [▶ 2564]	Gets the byte offset.
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)

	Name	Description
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Offset [▶ 2564]	Gets the offset of the IMember within the parent IStructType [▶ 2671] in bits or bytes dependent on IsBitType .
	ParentType [▶ 2538]	Gets the Parent Struct/Union of this IField [▶ 2535]. (Inherited from IField [▶ 2535].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)



Reference


















[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.58.1 IMember Properties

The [IMember](#) [[▶ 2561](#)] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitOffset [▶ 2563]	Gets the bit offset.

	Name	Description
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteOffset [▶ 2564]	Gets the byte offset.
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549] . (Inherited from IInstance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full IDataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Offset [▶ 2564]	Gets the offset of the IMember [▶ 2561] within the parent IStructType [▶ 2671] in bits or bytes dependent on IsBitType .
	ParentType [▶ 2538]	Gets the Parent Struct/Union of this IField [▶ 2535] . (Inherited from IField [▶ 2535].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549] . (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Reference

[IMember Interface \[▶ 2561\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.58.1.1 IMember.BitOffset Property

Gets the bit offset.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
int BitOffset { get; }
```

Property Value

Type: Int32
The bit offset.

Reference

[IMember Interface](#) [► 2561]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.58.1.2 IMember.ByteOffset Property

Gets the byte offset.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int ByteOffset { get; }
```

Property Value

Type: Int32
The byte offset.

Reference

[IMember Interface](#) [► 2561]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.58.1.3 IMember.Offset Property

Gets the offset of the [IMember](#) [► 2561] within the parent [IStructType](#) [► 2671] in bits or bytes dependent on [IsBitType](#).

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Offset { get; }
```

Property Value

Type: Int32
The bit offset.

Reference

[IMember Interface](#) [► 2561]

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.58.2 IMember Methods

The [IMember \[▶ 2561\]](#) type exposes the following members.

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021] .)

Reference

[IMember Interface \[▶ 2561\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.59 IMemberCollection Interface

Interface [IMemberCollection](#) Implements the [IInstanceCollection.T. \[▶ 2554\]](#)

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229








Syntax

C#

```
public interface IMemberCollection : IInstanceCollection<IMember>,
    IList<IMember>, ICollection<IMember>, IEnumerable<IMember>,
    IEnumerable
```

The [IMemberCollection](#) type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.IMember [▶ 2561] ..)
	Instances [▶ 2567]	Gets the Instance members (non static)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.IMember [▶ 2561] ..)
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.IMember [▶ 2561] ..)
	Item.String. [▶ 2556]	Gets the IInstance [▶ 2549] with the specified instance path. (Inherited from IInstanceCollection.T. [▶ 2554] .)
	Mode [▶ 2556]	Gets the InstanceCollectionMode [▶ 2573] . (Inherited from IInstanceCollection.T. [▶ 2554] .)
	Statics [▶ 2567]	Gets the Static Members

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.IMember [▶ 2561]..)
	CalcSize [▶ 2569]	Calculates the Byte Size of the IMemberCollection
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.IMember [▶ 2561]..)
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.IMember [▶ 2561]..)
	Contains(String) [▶ 2558]	Determines whether this collection contains an instance with the specified instance path. (Inherited from InstanceCollection.T. [▶ 2554]..)
	ContainsName [▶ 2558]	Determines whether this collection contains an instance with the specified instance name. (Inherited from InstanceCollection.T. [▶ 2554]..)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.IMember [▶ 2561]..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.IMember [▶ 2561]..)
	GetInstance [▶ 2559]	Gets the Instance [▶ 2549] by instance path. (Inherited from InstanceCollection.T. [▶ 2554]..)
	GetInstanceByName [▶ 2559]	Gets the Instance [▶ 2549] by instance name. (Inherited from InstanceCollection.T. [▶ 2554]..)
	IndexOf	Determines the index of a specific item in the IList.T.. (Inherited from IList.IMember [▶ 2561]..)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.IMember [▶ 2561]..)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.IMember [▶ 2561]..)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.IMember [▶ 2561]..)
	TryGetInstance [▶ 2560]	Tries to get the specified instance. (Inherited from InstanceCollection.T. [▶ 2554]..)
	TryGetInstanceByName [▶ 2560]	Tries to get the specified instance by name. (Inherited from InstanceCollection.T. [▶ 2554]..)
	TryGetMember [▶ 2569]	Gets the member with the specified name from the collection.

Reference


[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]







[TwinCAT.TypeSystem.InstanceCollection.T.](#) [[▶ 2554](#)]

6.11.59.1 IMemberCollection Properties

The IMemberCollection [[▶ 2565](#)] type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.IMember [▶ 2561]..)

	Name	Description
	Instances [▶ 2567]	Gets the Instance members (non static)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.IMember [▶ 2561]..)
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.IMember [▶ 2561]..)
	Item.String. [▶ 2556]	Gets the IInstance [▶ 2549] with the specified instance path. (Inherited from IInstanceCollection.T. [▶ 2554].)
	Mode [▶ 2556]	Gets the InstanceCollectionMode [▶ 2573]. (Inherited from IInstanceCollection.T. [▶ 2554].)
	Statics [▶ 2567]	Gets the Static Members

Reference

[IMemberCollection Interface](#) [[▶ 2565](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.59.1.1 IMemberCollection.Instances Property

Gets the Instance members (non static)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IInstanceCollection<IMember> Instances { get; }
```

Property Value

Type: [IInstanceCollection](#) [[▶ 2554](#)].[IMember](#) [[▶ 2561](#)].
The instances.

Reference

[IMemberCollection Interface](#) [[▶ 2565](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.59.1.2 IMemberCollection.Statics Property

Gets the Static Members

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IInstanceCollection<IMember> Statics { get; }
```

Property Value

Type: [ICollection](#) [▶ 2554].[IMember](#) [▶ 2561].
The statics.

Reference






[IMemberCollection Interface](#) [▶ 2565]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.59.2 IMemberCollection Methods

The [IMemberCollection](#) [▶ 2565] type exposes the following members.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection. IMember [▶ 2561]..)
	CalcSize [▶ 2569]	Calculates the Byte Size of the IMemberCollection [▶ 2565]
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection. IMember [▶ 2561]..)
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection. IMember [▶ 2561]..)
	Contains(String) [▶ 2558]	Determines whether this collection contains an instance with the specified instance path. (Inherited from ICollection.T. [▶ 2554].)
	ContainsName [▶ 2558]	Determines whether this collection contains an instance with the specified instance name. (Inherited from ICollection.T. [▶ 2554].)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection. IMember [▶ 2561]..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable . IMember [▶ 2561]..)
	GetInstance [▶ 2559]	Gets the Instance [▶ 2549] by instance path. (Inherited from ICollection.T. [▶ 2554].)
	GetInstanceByName [▶ 2559]	Gets the Instance [▶ 2549] by instance name. (Inherited from ICollection.T. [▶ 2554].)
	IndexOf	Determines the index of a specific item in the IList.T.. (Inherited from IList . IMember [▶ 2561]..)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList . IMember [▶ 2561]..)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection. IMember [▶ 2561]..)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList . IMember [▶ 2561]..)
	TryGetInstance [▶ 2560]	Tries to get the specified instance. (Inherited from ICollection.T. [▶ 2554].)
	TryGetInstanceByName [▶ 2560]	Tries to get the specified instance by name. (Inherited from ICollection.T. [▶ 2554].)
	TryGetMember [▶ 2569]	Gets the member with the specified name from the collection.

Reference

[IMemberCollection Interface \[► 2565\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.59.2.1 IMemberCollection.TryGetMember Method

Gets the member with the specified name from the collection.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool TryGetMember(  
    string memberName,  
    out IMember?? symbol  
)
```

Parameters

memberName	Type: System.String Name of the member.
symbol	Type: TwinCAT.TypeSystem.IMember [► 2561] . The symbol.

Return Value

Type: Boolean
true if found, false otherwise.

Reference

[IMemberCollection Interface \[► 2565\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.59.2.2 IMemberCollection.CalcSize Method

Calculates the Byte Size of the [IMemberCollection \[► 2565\]](#)

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int CalcSize()
```

Return Value

Type: Int32
System.Int32.

Remarks

This takes only the instance fields/members into account.

Reference

[IMemberCollection Interface](#) [[▶ 2565](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.60 INamespaceCollection Interface

Interface INamespaceCollection

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax

C#







```
public interface INamespaceCollection : INamespaceCollection<IDataType>,
    ICollection<INamespace<IDataType>>, IEnumerable<INamespace<IDataType>>,
    IEnumerable
```

The INamespaceCollection type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.INamespace [▶ 3102].IDataType [▶ 2475]...)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.INamespace [▶ 3102].IDataType [▶ 2475]...)

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.INamespace [▶ 3102].IDataType [▶ 2475]...)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.INamespace [▶ 3102].IDataType [▶ 2475]...)
	Contains	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.INamespace [▶ 3102].IDataType [▶ 2475]...)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.INamespace [▶ 3102].IDataType [▶ 2475]...)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.INamespace [▶ 3102].IDataType [▶ 2475]...)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.INamespace [▶ 3102].IDataType [▶ 2475]...)



Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.60.1 INamespaceCollection Properties

The [INamespaceCollection](#) [[▶ 2570](#)] type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.INamespace [▶ 3102].IDataType [▶ 2475]...)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.INamespace [▶ 3102].IDataType [▶ 2475]...)

Reference







[INamespaceCollection Interface \[▶ 2570\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.60.2 INamespaceCollection Methods

The [INamespaceCollection \[▶ 2570\]](#) type exposes the following members.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.INamespace [▶ 3102].IDataType [▶ 2475]...)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.INamespace [▶ 3102].IDataType [▶ 2475]...)
	Contains	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.INamespace [▶ 3102].IDataType [▶ 2475]...)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.INamespace [▶ 3102].IDataType [▶ 2475]...)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.INamespace [▶ 3102].IDataType [▶ 2475]...)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.INamespace [▶ 3102].IDataType [▶ 2475]...)

Reference

[INamespaceCollection Interface \[▶ 2570\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.61 INamespaceCollection.T. Interface

Interface `INamespaceCollection`

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#



```
public interface INamespaceCollection<T> : ICollection<INamespace<T>>,
    IEnumerable<INamespace<T>>, IEnumerable
where T : class, IDatatype
```

Type Parameters







T

The `INamespaceCollection.T`. type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the <code>ICollection.T</code> . (Inherited from <code>ICollection.INamespace [▶ 3102].T...</code>)
	IsReadOnly	Gets a value indicating whether the <code>ICollection.T</code> is read-only. (Inherited from <code>ICollection.INamespace [▶ 3102].T...</code>)

Methods

	Name	Description
	Add	Adds an item to the <code>ICollection.T</code> . (Inherited from <code>ICollection.INamespace [▶ 3102].T...</code>)
	Clear	Removes all items from the <code>ICollection.T</code> . (Inherited from <code>ICollection.INamespace [▶ 3102].T...</code>)
	Contains	Determines whether the <code>ICollection.T</code> contains a specific value. (Inherited from <code>ICollection.INamespace [▶ 3102].T...</code>)
	CopyTo	Copies the elements of the <code>ICollection.T</code> to an Array, starting at a particular Array index. (Inherited from <code>ICollection.INamespace [▶ 3102].T...</code>)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from <code>IEnumerable.INamespace [▶ 3102].T...</code>)
	Remove	Removes the first occurrence of a specific object from the <code>ICollection.T</code> . (Inherited from <code>ICollection.INamespace [▶ 3102].T...</code>)



Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.61.1 INamespaceCollection.T. Properties

The `INamespaceCollection.T`. [▶ 2571] generic type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the <code>ICollection.T</code> . (Inherited from <code>ICollection.INamespace [▶ 3102].T [▶ 2571]...</code>)
	IsReadOnly	Gets a value indicating whether the <code>ICollection.T</code> is read-only. (Inherited from <code>ICollection.INamespace [▶ 3102].T [▶ 2571]...</code>)

Reference







[INamespaceCollection.T. Interface \[▶ 2571\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.61.2 INamespaceCollection.T. Methods

The `INamespaceCollection.T`. [▶ 2571] generic type exposes the following members.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.INamespace [▶ 3102].T [▶ 2571]...)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.INamespace [▶ 3102].T [▶ 2571]...)
	Contains	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.INamespace [▶ 3102].T [▶ 2571]...)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.INamespace [▶ 3102].T [▶ 2571]...)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.INamespace [▶ 3102].T [▶ 2571]...)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.INamespace [▶ 3102].T [▶ 2571]...)

Reference

[INamespaceCollection.T. Interface \[▶ 2571\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.62 InstanceCollectionMode Enumeration

Enum InstanceCollectionMode

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum InstanceCollectionMode
```

Members

	Member name	Value	Description
	Names	0	InstanceCollection{T} is organized with InstanceNames instead of Instance Paths
	Path	1	InstanceCollection{T} is organized with InstancePaths in a flat list
	PathHierarchy	2	InstanceCollection{T} is organized with InstancePaths in a Hierarchy (Only Root objects appearing)

Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.63 InsufficientAccessRightsException Class

Insufficient rights for access

Inheritance Hierarchy

System.Object
 System.Exception
 TwinCAT.AdsException [[▶ 61](#)]
 TwinCAT.TypeSystem.SymbolException [[▶ 2933](#)]
 TwinCAT.TypeSystem.InsufficientAccessRightsException

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229


Syntax

C#











```
[SerializableAttribute]
public sealed class InsufficientAccessRightsException : SymbolException
```

The InsufficientAccessRightsException type exposes the following members.



Constructors





	Name	Description
	InsufficientAccessRightsException [▶ 2575]	Initializes a new instance of the InsufficientAccessRightsException class.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	InstancePath [▶ 2944]	Gets the instance path. (Inherited from SymbolException [▶ 2933].)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	Symbol [▶ 2944]	Gets the symbol. (Inherited from SymbolException [▶ 2933].)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)

	Name	Description
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData [▶ 2945]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from SymbolException [▶ 2933] .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.63.1 InsufficientAccessRightsException Constructor

Initializes a new instance of the [InsufficientAccessRightsException \[▶ 2573\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public InsufficientAccessRightsException(
    IValueSymbol symbol,
    SymbolAccessRights requested
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.IValueSymbol [▶ 2775] The symbol.
requested	Type: TwinCAT.TypeSystem.SymbolAccessRights [▶ 2928] The requested.

Reference





[InsufficientAccessRightsException Class \[▶ 2573\]](#)







[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.63.2 InsufficientAccessRightsException Properties

The [InsufficientAccessRightsException \[▶ 2573\]](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)

	Name	Description
	InstancePath [▶ 2944]	Gets the instance path. (Inherited from SymbolException [▶ 2933].)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	Symbol [▶ 2944]	Gets the symbol. (Inherited from SymbolException [▶ 2933].)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Reference







[InsufficientAccessRightsException Class](#) [[▶ 2573](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.63.3 InsufficientAccessRightsException Methods

The [InsufficientAccessRightsException](#) [[▶ 2573](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData [▶ 2945]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from SymbolException [▶ 2933].)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Reference

[InsufficientAccessRightsException Class](#) [[▶ 2573](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.64 IOversamplingArrayInstance Interface

Interface [IOversamplingArrayInstance](#)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax







C#

```
public interface IOversamplingArrayInstance : IArrayInstance,
    ISymbol, IAttributedInstance, IInstance, IBitSize
```


The IOversamplingArrayInstance type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	Dimensions [▶ 2456]	Gets the dimensions as read only collection. (Inherited from IArrayInstance [▶ 2453].)
	Elements [▶ 2456]	Gets the contained Array Elements as read only collection. (Inherited from IArrayInstance [▶ 2453].)
	ElementType [▶ 2457]	Gets the type of the contained elements. (Inherited from IArrayInstance [▶ 2453].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($BitSize \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Item [▶ 2457]	Gets the ISymbol [▶ 2691] with the specified indices. (Inherited from IArrayInstance [▶ 2453].)

	Name	Description
	OversamplingElement [▶ 2579]	Gets the oversampling element.
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from Instance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from AttributedInstance [▶ 2469].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference










[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

[TwinCAT.TypeSystem.IArrayInstance](#) [[▶ 2453](#)]

6.11.64.1 IOversamplingArrayInstance Properties

The [IOversamplingArrayInstance](#) [[▶ 2576](#)] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from AttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the Instance [▶ 2549] (Inherited from Instance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549]. (Inherited from Instance [▶ 2549].)
	Dimensions [▶ 2456]	Gets the dimensions as read only collection. (Inherited from IArrayInstance [▶ 2453].)
	Elements [▶ 2456]	Gets the contained Array Elements as read only collection. (Inherited from IArrayInstance [▶ 2453].)
	ElementType [▶ 2457]	Gets the type of the contained elements. (Inherited from IArrayInstance [▶ 2453].)

	Name	Description
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is a container type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Item [▶ 2457]	Gets the ISymbol [▶ 2691] with the specified indices. (Inherited from IArrayInstance [▶ 2453].)
	OversamplingElement [▶ 2579]	Gets the oversampling element.
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from AttributedInstance [▶ 2469].)

Reference

[IOversamplingArrayInstance](#) Interface [▶ 2576]

[TwinCAT.TypeSystem](#) Namespace [▶ 2083]

6.11.64.1.1 IOversamplingArrayInstance.OversamplingElement Property

Gets the oversampling element.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISymbol OversamplingElement { get; }
```

Property Value

Type: [ISymbol](#) [▶ 2691]

The oversampling element.

Reference

[IOversamplingArrayInstance Interface](#) [▶ 2576]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.64.2 IOversamplingArrayInstance Methods

The [IOversamplingArrayInstance](#) [▶ 2576] type exposes the following members.

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[IOversamplingArrayInstance Interface](#) [▶ 2576]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.65 IPointerInstance Interface

Interface representing an instance of an [IPointerType](#) [▶ 2584]

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax

C#

```
public interface IPointerInstance : ISymbol,
    IAttributedInstance, IInstance, IBitSize
```

The [IPointerInstance](#) type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)

	Name	Description
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the Instance [▶ 2549] (Inherited from Instance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549] . (Inherited from Instance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from Instance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from Instance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the Instance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from Instance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the Instance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from Instance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this Instance [▶ 2549] is static. (Inherited from Instance [▶ 2549].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Reference [▶ 2583]	Gets the resolved reference of Pointer / Reference (or NULL if PVOID)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549] . (Inherited from Instance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference








[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.65.1 IPointerInstance Properties

The [IPointerInstance](#) [[▶ 2580](#)] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)

	Name	Description
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Reference [▶ 2583]	Gets the resolved reference of Pointer / Reference (or NULL if PVOID)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Reference

[IPointerInstance Interface](#) [[▶ 2580](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.65.1.1 IPointerInstance.Reference Property

Gets the resolved reference of Pointer / Reference (or NULL if PVOID)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISymbol? Reference { get; }
```

Property Value

Type: [ISymbol](#) [[▶ 2691](#)]

The reference symbol or NULL if PVOID Pointer.

Reference

[IPointerInstance Interface](#) [[▶ 2580](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.65.2 IPointerInstance Methods

The [IPointerInstance](#) [[▶ 2580](#)] type exposes the following members.

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[IPointerInstance Interface \[▶ 2580\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.66 IPointerType Interface

Interface representing a pointer type

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229















Syntax

C#

```
public interface IPointerType : IDataTypeInfo,
    IBitSize
```

The IPointerType type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataTypeInfo [▶ 2475] (Inherited from IDataTypeInfo [▶ 2475] .)
	BitSize [▶ 2473]	Gets the size of the IDataTypeInfo [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472] .)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472] .)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataTypeInfo [▶ 2475] .)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataTypeInfo [▶ 2475] .)
	FullName [▶ 2479]	Gets the full name of the IDataTypeInfo [▶ 2475] (Namespace + Name) (Inherited from IDataTypeInfo [▶ 2475] .)
	Id [▶ 2480]	Gets the ID of the Data Type (Inherited from IDataTypeInfo [▶ 2475] .)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full IDataTypeInfo but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472] .)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472] .)
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataTypeInfo [▶ 2475] .)
	Namespace [▶ 2482]	Gets the namespace string within the IDataTypeInfo [▶ 2475] exists. (Inherited from IDataTypeInfo [▶ 2475] .)
	ReferencedType [▶ 2586]	Gets the the referenced type.
	ReferenceTypeName [▶ 2587]	Gets the name of the referenced datatype
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472] .)

Extension Methods

Name	Description
IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]















Also see about this

- [IDataType.IsContainer Property](#) [[▶ 2480](#)]
- [IDataType.IsPointer Property](#) [[▶ 2481](#)]
- [IDataType.IsPrimitive Property](#) [[▶ 2481](#)]
- [IDataType.IsReference Property](#) [[▶ 2482](#)]

6.11.66.1 IPointerType Properties

The [IPointerType](#) [[▶ 2584](#)] type exposes the following members.

Properties





	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataType [▶ 2475] (Inherited from IDataType [▶ 2475].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)
	FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from IDataType [▶ 2475].)
	Id [▶ 2480]	Gets the ID of the Data Type (Inherited from IDataType [▶ 2475].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full Data Type but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475].)
	Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475].)
	ReferencedType [▶ 2586]	Gets the the referenced type.
	ReferenceTypeName [▶ 2587]	Gets the name of the referenced datatype
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Reference

[IPointerType Interface](#) [[▶ 2584](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

Also see about this

-  [IDataType.IsContainer Property](#) [[▶ 2480](#)]
-  [IDataType.IsPointer Property](#) [[▶ 2481](#)]
-  [IDataType.IsPrimitive Property](#) [[▶ 2481](#)]
-  [IDataType.IsReference Property](#) [[▶ 2482](#)]

6.11.66.1.1 IPointerType.ReferencedType Property

Gets the the referenced type.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDataType? ReferencedType { get; }
```

Property Value

Type: [IDataType \[▸ 2475\]](#)

The datatype of the reference or **null** if not resolved.

Reference

[IPointerType Interface \[▸ 2584\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.66.1.2 IPointerType.ReferenceTypeName Property

Gets the name of the referenced datatype

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string ReferenceTypeName { get; }
```

Property Value

Type: String

The name of the reference datatype.

Reference

[IPointerType Interface \[▸ 2584\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.66.2 IPointerType Methods

The [IPointerType \[▸ 2584\]](#) type exposes the following members.

Extension Methods

	Name	Description
	IsArrayOfPrimitives [▸ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▸ 3021] .)
	IsArrayOfPrimitives(Boolean) [▸ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▸ 3021] .)
	IsContainer [▸ 3027]	Overloaded.

Name	Description
	Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive. [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
Resolve. [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[IPointerType Interface](#) [[▶ 2584](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.67 IPrimitiveType Interface

Interface [IPrimitiveType](#)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229














Syntax

C#

```
public interface IPrimitiveType : IDataTypeInfo,
    IBitSize
```

The [IPrimitiveType](#) type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataType [▶ 2475] (Inherited from IDataType [▶ 2475].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)
	FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from IDataType [▶ 2475].)
	Id [▶ 2480]	Gets the ID of the Data Type (Inherited from IDataType [▶ 2475].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full Data Type but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475].)
	Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475].)
	PrimitiveFlags [▶ 2591]	Indicates types of different PrimitiveTypes with flags.
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Extension Methods

	Name	Description
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)





	Name	Description
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve . [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

[TwinCAT.TypeSystem.IDataType](#) [▶ 2475]






Also see about this









-  [IDataType.IsContainer Property](#) [▶ 2480]
-  [IDataType.IsPointer Property](#) [▶ 2481]
-  [IDataType.IsPrimitive Property](#) [▶ 2481]
-  [IDataType.IsReference Property](#) [▶ 2482]

6.11.67.1 IPrimitiveType Properties

The [IPrimitiveType](#) [▶ 2588] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataType [▶ 2475] (Inherited from IDataType [▶ 2475].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)





	Name	Description
	FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from IDataType [▶ 2475] .)
	Id [▶ 2480]	Gets the ID of the Data Type (Inherited from IDataType [▶ 2475] .)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full Data Type but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472] .)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from IBitSize [▶ 2472] .)
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475] .)
	Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475] .)
	PrimitiveFlags [▶ 2591]	Indicates types of different PrimitiveTypes with flags.
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472] .)

Reference

[IPrimitiveType Interface \[▶ 2588\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

Also see about this

-  [IDataType.IsContainer Property \[▶ 2480\]](#)
-  [IDataType.IsPointer Property \[▶ 2481\]](#)
-  [IDataType.IsPrimitive Property \[▶ 2481\]](#)
-  [IDataType.IsReference Property \[▶ 2482\]](#)

6.11.67.1.1 IPrimitiveType.PrimitiveFlags Property

Indicates types of different PrimitiveTypes with flags.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
PrimitiveTypeFlags PrimitiveFlags { get; }
```

Property Value

Type: [PrimitiveTypeFlags \[▶ 2812\]](#)

Reference

[IPrimitiveType Interface \[▶ 2588\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.67.2 IPrimitiveType Methods

The [IPrimitiveType](#) [[▶ 2588](#)] type exposes the following members.

Extension Methods

Name	Description
IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[IPrimitiveType Interface](#) [[▶ 2588](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.68 IProcessImageAddress Interface

Interface describing a Process Image Address

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229









Syntax

C#

```
public interface IProcessImageAddress : IBitSize
```

The IProcessImageAddress type exposes the following members.

Properties

	Name	Description
	BitSize [▸ 2473]	Gets the size of the IDataType [▸ 2475] in bits. (Inherited from IBitSize [▸ 2472].)
	ByteSize [▸ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▸ 2472].)
	IndexGroup [▸ 2594]	Gets the index group of the Symbol
	IndexOffset [▸ 2594]	Gets the index offset of the Symbol
	IsBitType [▸ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▸ 2472].)
	IsByteAligned [▸ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▸ 2472].)
	IsVirtual [▸ 2595]	Gets a value indicating whether this instance is virtual.
	Size [▸ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▸ 2474] (Inherited from IBitSize [▸ 2472].)





Reference





[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.68.1 IProcessImageAddress Properties

The [IProcessImageAddress \[▸ 2593\]](#) type exposes the following members.

Properties

	Name	Description
	BitSize [▸ 2473]	Gets the size of the IDataType [▸ 2475] in bits. (Inherited from IBitSize [▸ 2472].)
	ByteSize [▸ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▸ 2472].)
	IndexGroup [▸ 2594]	Gets the index group of the Symbol
	IndexOffset [▸ 2594]	Gets the index offset of the Symbol

	Name	Description
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsVirtual [▶ 2595]	Gets a value indicating whether this instance is virtual.
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Reference

[IProcessImageAddress Interface](#) [[▶ 2593](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.68.1.1 IProcessImageAddress.IndexGroup Property

Gets the index group of the Symbol

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
uint IndexGroup { get; }
```

Property Value

Type: UInt32
The index group.

Reference

[IProcessImageAddress Interface](#) [[▶ 2593](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.68.1.2 IProcessImageAddress.IndexOffset Property

Gets the index offset of the Symbol

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
uint IndexOffset { get; }
```

Property Value

Type: UInt32
The index offset.

Reference

[IProcessImageAddress Interface \[► 2593\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.68.1.3 IProcessImageAddress.IsVirtual Property

Gets a value indicating whether this instance is virtual.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsVirtual { get; }
```

Property Value

Type: Boolean

true if this instance is virtual; otherwise, false.

Remarks

Virtual symbols are only organizational elements within the Symbols Hierarchy and cannot be accessed separately by IndexGroup/IndexOffset, Value Read/Writes, notifications or handles.

Reference

[IProcessImageAddress Interface \[► 2593\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.69 IReferenceInstance Interface

Interface representing an instance of an [IReferenceType \[► 2600\]](#)

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229





Syntax

C#

```
public interface IReferenceInstance : ISymbol,
    IAttributedInstance, IInstance, IBitSize
```

The IReferenceInstance type exposes the following members.

Properties

	Name	Description
	Attributes [► 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [► 2469] .)
	BitSize [► 2473]	Gets the size of the IDataType [► 2475] in bits. (Inherited from IBitSize [► 2472] .)
	ByteSize [► 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [► 2472] .)
	Category [► 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [► 2691] .)

	Name	Description
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is a container type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	ReferencedType [▶ 2598]	Gets the referenced type
	ResolvedByteSize [▶ 2599]	Get the ByteSize of the (completely) resolved Symbol
	ResolvedCategory [▶ 2599]	Gets the Category of the (completely) resolved Symbol.
	ResolvedType [▶ 2599]	Gets the (completely) resolved type
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference











[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.69.1 IReferenceInstance Properties

The [IReferenceInstance](#) [[▶ 2595](#)] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($BitSize \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)

	Name	Description
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	ReferencedType [▶ 2598]	Gets the referenced type
	ResolvedByteSize [▶ 2599]	Get the ByteSize of the (completely) resolved Symbol
	ResolvedCategory [▶ 2599]	Gets the Category of the (completely) resolved Symbol.
	ResolvedType [▶ 2599]	Gets the (completely) resolved type
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Reference

[IReferenceInstance Interface](#) [[▶ 2595](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.69.1.1 IReferenceInstance.ReferencedType Property

Gets the referenced type

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDataType? ReferencedType { get; }
```

Property Value

Type: [IDataType](#) [[▶ 2475](#)]

The type of the referenced type

Remarks

This is no complete resolvement, only the next level. The referenced type can be a reference again.

Reference

[IReferenceInstance Interface](#) [[▶ 2595](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.69.1.2 IReferenceInstance.ResolvedByteSize Property

Get the ByteSize of the (completely) resolved Symbol

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int ResolvedByteSize { get; }
```

Property Value

Type: Int32

The byte size of the resolved type.

Reference

[IReferenceInstance Interface](#) [[▶ 2595](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.69.1.3 IReferenceInstance.ResolvedCategory Property

Gets the Category of the (completely) resolved Symbol.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
DataTypeCategory ResolvedCategory { get; }
```

Property Value

Type: [DataTypeCategory](#) [[▶ 2111](#)]

The resolved category.

Reference

[IReferenceInstance Interface](#) [[▶ 2595](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.69.1.4 IReferenceInstance.ResolvedType Property

Gets the (completely) resolved type

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDataType? ResolvedType { get; }
```

Property ValueType: [IDataType](#) [[▶ 2475](#)]

The type of the resolved symbol

Reference[IReferenceInstance Interface](#) [[▶ 2595](#)][TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]**6.11.69.2 IReferenceInstance Methods**The [IReferenceInstance](#) [[▶ 2595](#)] type exposes the following members.**Extension Methods**

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)







Reference[IReferenceInstance Interface](#) [[▶ 2595](#)][TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]**6.11.70 IReferenceType Interface**












Interface representing a reference type

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public interface IReferenceType : IDataType,
    IBitSize
```

The [IReferenceType](#) type exposes the following members.**Properties**

	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataType [▶ 2475] (Inherited from IDataType [▶ 2475].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)
	FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from IDataType [▶ 2475].)

	Name	Description
	Id [▶ 2480]	Gets the ID of the DataType (Inherited from IDataType [▶ 2475].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475].)
	Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475].)
	ReferencedType [▶ 2603]	Gets the the referenced type.
	ReferencedTypeNa me [▶ 2604]	Gets the name of the referenced type.
	ResolvedByteSize [▶ 2604]	Get the ByteSize of the (completely) resolved Symbol
	ResolvedCategory [▶ 2604]	Gets the Category of the (completely) resolved Symbol.
	ResolvedType [▶ 2605]	Gets the (completely) resolved type
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Extension Methods





	Name	Description
	IsArrayOfPrimitives. [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer. [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)

	Name	Description
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]








Also see about this











-  [IDataType.IsContainer Property](#) [[▶ 2480](#)]
-  [IDataType.IsPointer Property](#) [[▶ 2481](#)]
-  [IDataType.IsPrimitive Property](#) [[▶ 2481](#)]
-  [IDataType.IsReference Property](#) [[▶ 2482](#)]

6.11.70.1 IReferenceType Properties

The [IReferenceType](#) [[▶ 2600](#)] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataType [▶ 2475] (Inherited from IDataType [▶ 2475].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)
	FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from IDataType [▶ 2475].)
	Id [▶ 2480]	Gets the ID of the Data Type (Inherited from IDataType [▶ 2475].)





	Name	Description
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475].)
	Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475].)
	ReferencedType [▶ 2603]	Gets the the referenced type.
	ReferencedTypeNa me [▶ 2604]	Gets the name of the referenced type.
	ResolvedByteSize [▶ 2604]	Get the ByteSize of the (completely) resolved Symbol
	ResolvedCategory [▶ 2604]	Gets the Category of the (completely) resolved Symbol.
	ResolvedType [▶ 2605]	Gets the (completely) resolved type
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Reference

[IReferenceType Interface](#) [[▶ 2600](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

Also see about this

-  [IDataType.IsContainer Property](#) [[▶ 2480](#)]
-  [IDataType.IsPointer Property](#) [[▶ 2481](#)]
-  [IDataType.IsPrimitive Property](#) [[▶ 2481](#)]
-  [IDataType.IsReference Property](#) [[▶ 2482](#)]

6.11.70.1.1 IReferenceType.ReferencedType Property

Gets the the referenced type.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDataType? ReferencedType { get; }
```

Property Value

Type: [IDataType](#) [[▶ 2475](#)]

The data type of the referenced type or **null** if not resolved.

Reference

[IReferenceType Interface \[► 2600\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.70.1.2 IReferenceType.ReferencedTypeName Property

Gets the name of the referenced type.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string ReferencedTypeName { get; }
```

Property Value

Type: String

The name of the referenced type.

Reference

[IReferenceType Interface \[► 2600\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.70.1.3 IReferenceType.ResolvedByteSize Property

Get the ByteSize of the (completely) resolved Symbol

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int ResolvedByteSize { get; }
```

Property Value

Type: Int32

The size of the resolved byte.

Reference

[IReferenceType Interface \[► 2600\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.70.1.4 IReferenceType.ResolvedCategory Property

Gets the Category of the (completely) resolved Symbol.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
DataTypeCategory ResolvedCategory { get; }
```

Property Value

Type: [DataTypeCategory](#) [▶ 2111]

The resolved category.

Reference

[IReferenceType Interface](#) [▶ 2600]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.70.1.5 IReferenceType.ResolvedType Property

Gets the (completely) resolved type

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDataType? ResolvedType { get; }
```

Property Value

Type: [IDataType](#) [▶ 2475]

The type of the resolved symbol

Reference

[IReferenceType Interface](#) [▶ 2600]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.70.2 IReferenceType Methods

The [IReferenceType](#) [▶ 2600] type exposes the following members.

Extension Methods

	Name	Description
	IsArrayOfPrimitives . [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer . [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)

	Name	Description
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021] .)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021] .)
	IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021] .)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021] .)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021] .)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021] .)
	ResolvableAsPrimitive. [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021] .)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021] .)
	Resolve. [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021] .)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021] .)

Reference

[IReferenceType Interface \[▶ 2600\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.71 IRpcCallableInstance Interface

Interface for an RPC callable PLC Method (Remote procedure call)

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#






```
public interface IRpcCallableInstance
```

The IRpcCallableInstance type exposes the following members.

Properties

	Name	Description
	RpcMethods [▶ 2607]	Gets the Method descriptions for the IRpcCallableType [▶ 2624]

Methods

	Name	Description
	InvokeRpcMethod(String, .Object.) [▶ 2609]	Invokes the specified RPC Method
	InvokeRpcMethod(String, .Object., .Object.) [▶ 2610]	Invokes the specified RPC Method
	InvokeRpcMethodAsync [▶ 2614]	Invokes the specified RPC Method asynchronously
	TryInvokeRpcMethod(String, .Object., Object.) [▶ 2619]	Invokes the specified RPC Method
	TryInvokeRpcMethod(String, .Object., Object., Object.) [▶ 2620]	Invokes the specified RPC Method


Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.71.1 IRpcCallableInstance Properties

The [IRpcCallableInstance](#) [[▶ 2606](#)] type exposes the following members.

Properties

	Name	Description
	RpcMethods [▶ 2607]	Gets the Method descriptions for the IRpcCallableType [▶ 2624]

Reference

[IRpcCallableInstance Interface](#) [[▶ 2606](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.71.1.1 IRpcCallableInstance.RpcMethods Property

Gets the Method descriptions for the [IRpcCallableType](#) [[▶ 2624](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IRpcMethodCollection RpcMethods { get; }
```

Property Value

Type: [IRpcMethodCollection](#) [[▸ 2629](#)]

The methods.

Reference











[IRpcCallableInstance Interface](#) [[▸ 2606](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.71.2 IRpcCallableInstance Methods

The [IRpcCallableInstance](#) [[▸ 2606](#)] type exposes the following members.

Methods

	Name	Description
 	InvokeRpcMethod(String, .Object.) [▸ 2609]	Invokes the specified RPC Method
 	InvokeRpcMethod(String, .Object., .Object.) [▸ 2610]	Invokes the specified RPC Method
 	InvokeRpcMethodAsync [▸ 2614]	Invokes the specified RPC Method asynchronously
 	TryInvokeRpcMethod(String, .Object., .Object.) [▸ 2619]	Invokes the specified RPC Method
 	TryInvokeRpcMethod(String, .Object., .Object., Object.) [▸ 2620]	Invokes the specified RPC Method



Reference

[IRpcCallableInstance Interface](#) [[▸ 2606](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.71.2.1 IRpcCallableInstance.InvokeRpcMethod Method

Overload List

	Name	Description
	InvokeRpcMethod(String, .Object.) [▶ 2609]	Invokes the specified RPC Method
	InvokeRpcMethod(String, .Object., .Object..) [▶ 2610]	Invokes the specified RPC Method

Reference

[IRpcCallableInstance Interface](#) [[▶ 2606](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.71.2.1.1 IRpcCallableInstance.InvokeRpcMethod Method (String, .Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object? InvokeRpcMethod(
    string methodName,
    Object[]? inParameters
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object. The input parameters or NULL

Return Value

Type: Object
The return value of the Method (as object).

Remarks

This method only supports primitive data types as inParameters. Any available outparameters will be ignored. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
```

```

/// Defines the entry point of the application.
/// </summary>
/// <param name="args">The arguments.</param>
static void Main(string[] args)
{
    // Get the AdsAddress from command-line arguments
    AmsAddress address = ArgParser.Parse(args);

    using (AdsClient client = new AdsClient())
    {
        //client.Synchronize = false;

        // Connect to the target device
        client.Connect(address);

        SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
        ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

        // Get the Symbols (Dynamic Symbols)

        IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
the PLC Program

        // Call a Method that has the following signature (within MAIN Program)
        /* {attribute 'TcRpcEnable'}
        METHOD PUBLIC M_Add : INT
        VAR_INPUT
            i1 : INT := 0;
            i2 : INT := 0;
        END_VAR
        */

        short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

        // Call a Method that has no parameter and returns VOID
        main.InvokeRpcMethod("M_Method1", new object[] {});

        //Browsing RpcMethods
        foreach(IRpcMethod method in main.RpcMethods)
        {
            string methodName = method.Name;

            foreach(IRpcMethodParameter parameter in method.Parameters)
            {
                string parameterName = parameter.Name;
                string parameterType = parameter.TypeName;
            }
        }
    }
}

```

Reference

[IRpcCallableInstance Interface \[► 2606\]](#)

[InvokeRpcMethod Overload \[► 2609\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.71.2.1.2 IRpcCallableInstance.InvokeRpcMethod Method (String, .Object., .Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object? InvokeRpcMethod(
    string methodName,
    Object[]? inParameters,
    out Object[]? outParameters
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object. The input parameters or NULL
outParameters	Type: .System.Object.. The output parameters.

Return Value

Type: Object
The return value of the Method (as object).

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [► 2094] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});
        }
    }
}
```

```

// Call a Method that has no parameter and returns VOID
main.InvokeRpcMethod("M_Method1", new object[] {});

//Browsing RpcMethods
foreach(IRpcMethod method in main.RpcMethods)
{
    string methodName = method.Name;

    foreach(IRpcMethodParameter parameter in method.Parameters)
    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
}
}

```

Reference

[IRpcCallableInstance Interface](#) [► 2606]

[InvokeRpcMethod Overload](#) [► 2609]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.71.2.1.3 IRpcCallableInstance.InvokeRpcMethod Method (String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, .Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```

Object InvokeRpcMethod(
    string methodName,
    Object[] inParameters,
    AnyTypeSpecifier[] outSpecifiers,
    AnyTypeSpecifier retSpecifier,
    out Object[] outParameters
)

```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object. The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [► 2094]. The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [► 2094] The ret specifier (specifying the return value) or NULL.
outParameters	Type: .System.Object. The out parameters.

Return Value

Type: [Object](#)
The return value of the Method (as object).

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```

Reference

[IRpcCallableInstance Interface \[► 2606\]](#)

[InvokeRpcMethod Overload \[► 2609\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.71.2.2 IRpcCallableInstance.InvokeRpcMethodAsync Method

Invokes the specified RPC Method asynchronously

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultRpcMethodAccess> InvokeRpcMethodAsync (
    string methodName,
    Object[]? inParameters,
    CancellationToken cancel
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: System.Object. The parameters.
cancel	Type: System.Threading.CancellationToken The cancellation token

Return Value

Type: [Task.ResultRpcMethodAccess](#) [[▸ 3213](#)].

A task that represents the asynchronous 'InvokeRpcMethod' operation. The [ResultRpcMethodAccess](#) [[▸ 3213](#)] results contains the return value ([ReturnValue](#) [[▸ 3216](#)]) together with the output parameters. The succeeded communication is indicated by the [ErrorCode](#) property ([ErrorCode](#) [[▸ 3202](#)]) after the communication.

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [[▸ 2094](#)] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
```

```

the PLC Program

// Call a Method that has the following signature (within MAIN Program)
/* {attribute 'TcRpcEnable'}
METHOD PUBLIC M_Add : INT
VAR_INPUT
    i1 : INT := 0;
    i2 : INT := 0;
END_VAR
*/

short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

// Call a Method that has no parameter and returns VOID
main.InvokeRpcMethod("M_Method1", new object[] {});

//Browsing RpcMethods
foreach(IRpcMethod method in main.RpcMethods)
{
    string methodName = method.Name;

    foreach(IRpcMethodParameter parameter in method.Parameters)
    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
}

```

Reference

[IRpcCallableInstance Interface \[► 2606\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.71.2.2.1 IRpcCallableInstance.InvokeRpcMethodAsync Method (String, .Object., CancellationToken)

Invokes the specified RPC Method asynchronously

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```

Task<ResultRpcMethodAccess> InvokeRpcMethodAsync (
    string methodName,
    Object[] inParameters,
    CancellationToken cancel
)

```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
cancel	Type: System.Threading.CancellationToken The cancellation token

Return Value

Type: [Task.ResultRpcMethodAccess](#) [▸ 3213].

A task that represents the asynchronous 'InvokeRpcMethod' operation. The [ResultRpcMethodAccess](#) [▸ 3213] results contains the return value ([ReturnValue](#) [▸ 3216]) together with the output parameters. The succeeded communication is indicated by the [ErrorCode](#) property ([ErrorCode](#) [▸ 3202]) after the communication.

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [▸ 2094] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to `byte[]` arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```


Reference

[IRpcCallableInstance Interface](#) [► 2606]

[InvokeRpcMethodAsync Overload](#) [► 2614]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.71.2.2 IRpcCallableInstance.InvokeRpcMethodAsync Method (String, .Object., .AnyTypeSpecifier., AnyTypeSpecifier, CancellationToken)

Invokes the specified RPC Method asynchronously

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934 addedca3e72bc0ea15da1c14

Syntax

C#

```
Task<ResultRpcMethodAccess> InvokeRpcMethodAsync (
    string methodName,
    Object[] inParameters,
    AnyTypeSpecifier[] outSpecifiers,
    AnyTypeSpecifier retSpecifier,
    CancellationToken cancel
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object . The parameters.
outSpecifiers	Type: .TwinCAT.TypeSystem.AnyTypeSpecifier [► 2094]. The out specifiers (specifying the out types) or NULL.
retSpecifier	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [► 2094] The ret specifier (specifying the return value) or NULL.
cancel	Type: System.Threading.CancellationToken The cancellation token

Return Value

Type: [Task.ResultRpcMethodAccess](#) [► 3213].

A task that represents the asynchronous 'InvokeRpcMethod' operation. The [ResultRpcMethodAccess](#) [► 3213] results contains the return value ([ReturnValue](#) [► 3216]) together with the output parameters. The succeeded communication is indicated by the [ErrorCode](#) property ([ErrorCode](#) [► 3202]) after the communication.

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```

class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}

```

Reference



[IRpcCallableInstance Interface \[► 2606\]](#)

[InvokeRpcMethodAsync Overload \[► 2614\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.71.2.3 IRpcCallableInstance.TryInvokeRpcMethod Method

Overload List

	Name	Description
	TryInvokeRpcMethod(String, .Object., Object.) [▶ 2619]	Invokes the specified RPC Method
	TryInvokeRpcMethod(String, .Object., .Object., Object.) [▶ 2620]	Invokes the specified RPC Method

Reference

[IRpcCallableInstance Interface](#) [[▶ 2606](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.71.2.3.1 IRpcCallableInstance.TryInvokeRpcMethod Method (String, .Object., Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int TryInvokeRpcMethod(
    string methodName,
    Object[]? inParameters,
    out Object?? retValue
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object. The parameters.
retValue	Type: System.Object. The return value of the RPC method as object.

Return Value

Type: Int32

The result value of the call (ErrorCode). 0 means Succeeded.

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [[▶ 2094](#)] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] { (short) 3, (short) 4 });

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```

Reference

[IRpcCallableInstance Interface \[► 2606\]](#)

[TryInvokeRpcMethod Overload \[► 2619\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.71.2.3.2 IRpcCallableInstance.TryInvokeRpcMethod Method (String, .Object., .Object., Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int TryInvokeRpcMethod(
    string methodName,
    Object[]? inParameters,
    out Object[]? outParameters,
    out Object?? retValue
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object. The parameters.
outParameters	Type: .System.Object.. The out parameters.
retValue	Type: System.Object. The return value of the RPC method as object.

Return Value

Type: Int32
The result value of the call (ErrorCode). 0 means Succeeded.

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [[▶ 2094](#)] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
```


outParameters Type: System.Object.
The out parameters.

retValue Type: System.Object.
The return value of the RPC method./>

Return Value

Type: Int32
AdsErrorCode.

Remarks

The RpcMethod optionally support In-Parameters, Out-Parameters and Return values. Therefore the parameters inParameters, outParameters, outSpecifiers, retSpecifier are allowed to be empty or NULL. In case of using primitive datatypes, the type specifier parameters (outSpecifiers and retSpecifier) are not necessary and should not be set.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IRpcStructInstance main = (IRpcStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```


Syntax

C#

```
IRpcMethodCollection RpcMethods { get; }
```

Property Value

Type: [IRpcMethodCollection](#) [[▶ 2629](#)]
 The RPC methods.

Remarks

The DataType (Structure) must be marked with the PlcAttribute 'TcRpcEnable' to enable RpcMethods, otherwise RpcMethods are not passed through to the ADS symbolic information.

Reference

[IRpcCallableType Interface](#) [[▶ 2624](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.73 IRpcMethod Interface

Interface describes an RPC Method

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229









Syntax

C#

```
public interface IRpcMethod
```

The IRpcMethod type exposes the following members.

Properties

	Name	Description
	Comment [▶ 2626]	Gets the Method comment.
	InParameters [▶ 2626]	Gets the In-Parameters of the IRpcMethod
	IsVoid [▶ 2627]	Gets a value indicating whether this IRpcMethod has no return parameter
	Name [▶ 2627]	Gets the name of the method
	OutParameters [▶ 2628]	Gets the Out-Parameters of the IRpcMethod
	Parameters [▶ 2628]	Gets all parameters (In, Out and ref parameters) of the .
	ReturnType [▶ 2628]	Gets the return type.
	ReturnTypeSize [▶ 2629]	Gets the size of the return type in bytes.









Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.73.1 IRpcMethod Properties

The [IRpcMethod](#) [[▶ 2625](#)] type exposes the following members.

Properties

	Name	Description
	Comment [▶ 2626]	Gets the Method comment.
	InParameters [▶ 2626]	Gets the In-Parameters of the IRpcMethod [▶ 2625]
	IsVoid [▶ 2627]	Gets a value indicating whether this IRpcMethod [▶ 2625] has no return parameter
	Name [▶ 2627]	Gets the name of the method
	OutParameters [▶ 2628]	Gets the Out-Parameters of the IRpcMethod [▶ 2625]
	Parameters [▶ 2628]	Gets all parameters (In, Out and ref parameters) of the . [▶ 2625]
	ReturnType [▶ 2628]	Gets the return type.
	ReturnTypeSize [▶ 2629]	Gets the size of the return type in bytes.

Reference

[IRpcMethod Interface](#) [[▶ 2625](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.73.1.1 IRpcMethod.Comment Property

Gets the Method comment.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string Comment { get; }
```

Property Value

Type: String
The comment.

Reference

[IRpcMethod Interface](#) [[▶ 2625](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.73.1.2 IRpcMethod.InParameters Property

Gets the In-Parameters of the [IRpcMethod](#) [[▶ 2625](#)]

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IRpcMethodParameterCollection InParameters { get; }
```

Property Value

Type: [IRpcMethodParameterCollection](#) [▸ 2639]

The In- and Ref-Parameters

Reference

[IRpcMethod Interface](#) [▸ 2625]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.73.1.3 IRpcMethod.IsVoid Property

Gets a value indicating whether this [IRpcMethod](#) [▸ 2625] has no return parameter

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsVoid { get; }
```

Property Value

Type: Boolean

true if this instance is void; otherwise, false.

Reference

[IRpcMethod Interface](#) [▸ 2625]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.73.1.4 IRpcMethod.Name Property

Gets the name of the method

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string Name { get; }
```

Property Value

Type: String

The name.

Reference

[IRpcMethod Interface](#) [[▶ 2625](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.73.1.5 IRpcMethod.OutParameters Property

Gets the Out-Parameters of the [IRpcMethod](#) [[▶ 2625](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IRpcMethodParameterCollection OutParameters { get; }
```

Property Value

Type: [IRpcMethodParameterCollection](#) [[▶ 2639](#)]

The out- and ref-parameters.

Remarks

The Out-Parameters doesn't include the Ref Parameters. These are included in the [InParameters](#) [[▶ 2626](#)] set.

Reference

[IRpcMethod Interface](#) [[▶ 2625](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.73.1.6 IRpcMethod.Parameters Property

Gets all parameters (In, Out and ref parameters) of the [_](#) [[▶ 2625](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IRpcMethodParameterCollection Parameters { get; }
```

Property Value

Type: [IRpcMethodParameterCollection](#) [[▶ 2639](#)]

The parameters.

Reference

[IRpcMethod Interface](#) [[▶ 2625](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.73.1.7 IRpcMethod.ReturnType Property

Gets the return type.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string ReturnType { get; }
```

Property Value

Type: String
Return type.

Reference

[IRpcMethod Interface](#) [[▶ 2625](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.73.1.8 IRpcMethod.ReturnTypeSize Property

Gets the size of the return type in bytes.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int ReturnTypeSize { get; }
```

Property Value

Type: Int32
The size of the return type.

Reference

[IRpcMethod Interface](#) [[▶ 2625](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.74 IRpcMethodCollection Interface

Interface for RPC Method collections.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229





Syntax

C#













```
public interface IRpcMethodCollection : IList<IRpcMethod>,  
    ICollection<IRpcMethod>, IEnumerable<IRpcMethod>, IEnumerable
```

The IRpcMethodCollection type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.IRpcMethod [▶ 2625]..)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.IRpcMethod [▶ 2625]..)
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.IRpcMethod [▶ 2625]..)
	Item.String. [▶ 2631]	Gets the IRpcMethod [▶ 2625] with the specified method name.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.IRpcMethod [▶ 2625]..)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.IRpcMethod [▶ 2625]..)
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.IRpcMethod [▶ 2625]..)
	Contains(String) [▶ 2633]	Determines whether this collection contains the specified method name.
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.IRpcMethod [▶ 2625]..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.IRpcMethod [▶ 2625]..)
	IndexOf	Determines the index of a specific item in the IList.T.. (Inherited from IList.IRpcMethod [▶ 2625]..)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.IRpcMethod [▶ 2625]..)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.IRpcMethod [▶ 2625]..)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.IRpcMethod [▶ 2625]..)
	TryGetMethod(Int32, IRpcMethod.) [▶ 2634]	Tries to get the specified method.
	TryGetMethod(String, IRpcMethod.) [▶ 2634]	Tries to get the specified method.





Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.74.1 IRpcMethodCollection Properties

The [IRpcMethodCollection](#) [[▶ 2629](#)] type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.IRpcMethod [▶ 2625].)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.IRpcMethod [▶ 2625].)
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.IRpcMethod [▶ 2625].)
	Item.String. [▶ 2631]	Gets the IRpcMethod [▶ 2625] with the specified method name.



Reference

[IRpcMethodCollection Interface \[\[▶ 2629\]\(#\)\]](#)

[TwinCAT.TypeSystem Namespace \[\[▶ 2083\]\(#\)\]](#)

6.11.74.1.1 IRpcMethodCollection.Item Property

Overload List

	Name	Description
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.IRpcMethod [▶ 2625].)
	Item.String. [▶ 2631]	Gets the IRpcMethod [▶ 2625] with the specified method name.

Reference

[IRpcMethodCollection Interface \[\[▶ 2629\]\(#\)\]](#)

[TwinCAT.TypeSystem Namespace \[\[▶ 2083\]\(#\)\]](#)

6.11.74.1.1.1 IRpcMethodCollection.Item Property (String)

Gets the [IRpcMethod \[\[▶ 2625\]\(#\)\]](#) with the specified method name.

Namespace: [TwinCAT.TypeSystem \[\[▶ 2083\]\(#\)\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IRpcMethod this[
    string methodName
] { get; }
```

Parameters

methodName	Type: System.String Name of the method.
------------	--

Return Value

Type: [IRpcMethod \[\[▶ 2625\]\(#\)\]](#)
RpcMethod.

Exceptions

Exception	Condition
KeyNotFoundException	

Reference

[IRpcMethodCollection Interface \[► 2629\]](#)













[Item Overload \[► 2631\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.74.2 IRpcMethodCollection Methods

The [IRpcMethodCollection \[► 2629\]](#) type exposes the following members.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.IRpcMethod [► 2625] ..)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.IRpcMethod [► 2625] ..)
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.IRpcMethod [► 2625] ..)
	Contains(String) [► 2633]	Determines whether this collection contains the specified method name.
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.IRpcMethod [► 2625] ..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.IRpcMethod [► 2625] ..)
	IndexOf	Determines the index of a specific item in the IList.T.. (Inherited from IList.IRpcMethod [► 2625] ..)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.IRpcMethod [► 2625] ..)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.IRpcMethod [► 2625] ..)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.IRpcMethod [► 2625] ..)
	TryGetMethod(Int32, IRpcMethod.) [► 2634]	Tries to get the specified method.
	TryGetMethod(String, IRpcMethod.) [► 2634]	Tries to get the specified method.



Reference

[IRpcMethodCollection Interface \[► 2629\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.74.2.1 IRpcMethodCollection.Contains Method

Overload List

	Name	Description
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.IRpcMethod [▶ 2625].)
	<u>Contains(String)</u> ▶ 2633	Determines whether this collection contains the specified method name.

Reference

[IRpcMethodCollection Interface](#) [[▶ 2629](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.74.2.1.1 IRpcMethodCollection.Contains Method (String)

Determines whether this collection contains the specified method name.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool Contains(
    string methodName
)
```

Parameters

methodName	Type: System.String Name of the method.
------------	--

Return Value

Type: Boolean
true if contained.; otherwise, false.

Reference


[IRpcMethodCollection Interface](#) [[▶ 2629](#)]


[Contains Overload](#) [[▶ 2633](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.74.2.2 IRpcMethodCollection.TryGetMethod Method

Overload List

	Name	Description
	<u>TryGetMethod(Int32, IRpcMethod.)</u> ▶ 2634	Tries to get the specified method.

	Name	Description
	TryGetMethod(String, IRpcMethod.) [▶ 2634]	Tries to get the specified method.

Reference

[IRpcMethodCollection Interface](#) [▶ 2629]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.74.2.2.1 IRpcMethodCollection.TryGetMethod Method (Int32, IRpcMethod.)

Tries to get the specified method.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool TryGetMethod(
    int vTableIndex,
    out IRpcMethod?? method
)
```

Parameters

vTableIndex	Type: System.Int32 vTableIndex.
method	Type: TwinCAT.TypeSystem.IRpcMethod [▶ 2625]. The method if fund, NULL otherwise.

Return Value

Type: Boolean
true if found, false otherwise.

Reference

[IRpcMethodCollection Interface](#) [▶ 2629]

[TryGetMethod Overload](#) [▶ 2633]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.74.2.2.2 IRpcMethodCollection.TryGetMethod Method (String, IRpcMethod.)

Tries to get the specified method.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool TryGetMethod(
    string methodName,
    out IRpcMethod?? method
)
```

Parameters

methodName	Type: System.String Name of the method.
method	Type: TwinCAT.TypeSystem.IRpcMethod [▶ 2625]. The method if found, NULL otherwise.

Return Value

Type: Boolean
true if found, false otherwise.

Reference

- [IRpcMethodCollection Interface \[▶ 2629\]](#)
- [TryGetMethod Overload \[▶ 2633\]](#)
- [TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.75 IRpcMethodParameter Interface

Interface IRpcMethodParameter

Namespace: TwinCAT.TypeSystem [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229







Syntax



C#

```
public interface IRpcMethodParameter
```

The IRpcMethodParameter type exposes the following members.

Properties

	Name	Description
	HasLengthsParameter [▶ 2636]	Gets a value indicating whether this instance has a related Lengths Parameter.
	IsInput [▶ 2639]	Gets a value indicating whether this instance is input.
	IsOutput [▶ 2639]	Gets a value indicating whether this instance is output.
	LengthsParameterIndex [▶ 2637]	Gets the index of the Lengths parameter (within the MethodParameter List)
	Name [▶ 2637]	Gets the Parameter Name
	ParameterFlags [▶ 2637]	Gets the parameter flags.

	Name	Description
	Size [▸ 2638]	Gets the size of the IRpcMethodParameter
	TypeName [▸ 2638]	Gets the Data type of the Parameter









Reference

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.75.1 IRpcMethodParameter Properties

The [IRpcMethodParameter \[▸ 2635\]](#) type exposes the following members.

Properties

	Name	Description
	HasLengthIsParameter [▸ 2636]	Gets a value indicating whether this instance has a related Lengths Parameter.
	IsInput [▸ 2639]	Gets a value indicating whether this instance is input.
	IsOutput [▸ 2639]	Gets a value indicating whether this instance is output.
	LengthsParameterIndex [▸ 2637]	Gets the index of the Lengths parameter (within the MethodParameter List)
	Name [▸ 2637]	Gets the Parameter Name
	ParameterFlags [▸ 2637]	Gets the parameter flags.
	Size [▸ 2638]	Gets the size of the IRpcMethodParameter [▸ 2635]
	TypeName [▸ 2638]	Gets the Data type of the Parameter

Reference

[IRpcMethodParameter Interface \[▸ 2635\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.75.1.1 IRpcMethodParameter.HasLengthIsParameter Property

Gets a value indicating whether this instance has a related Lengths Parameter.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool HasLengthIsParameter { get; }
```

Property Value

Type: Boolean

true if this instance has a Lengths parameter; otherwise, false.

Reference

[IRpcMethodParameter Interface](#) [► 2635]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.75.1.2 IRpcMethodParameter.LengthIsParameterIndex Property

Gets the index of the LengthIs parameter (within the MethodParameter List)

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int LengthIsParameterIndex { get; }
```

Property Value

Type: Int32

The index of the length is parameter.

Remarks

This field references to the Parameter that defines the length for this generic one. Equally to the marshalling attributes of COM (sizeof, length) this enables to transport parameter of type (PVOID)

Reference

[IRpcMethodParameter Interface](#) [► 2635]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.75.1.3 IRpcMethodParameter.Name Property

Gets the Parameter Name

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string Name { get; }
```

Property Value

Type: String

The name.

Reference

[IRpcMethodParameter Interface](#) [► 2635]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.75.1.4 IRpcMethodParameter.ParameterFlags Property

Gets the parameter flags.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
MethodParamFlags ParameterFlags { get; }
```

Property Value

Type: [MethodParamFlags](#) [[▶ 2812](#)]

The parameter flags.

Reference

[IRpcMethodParameter Interface](#) [[▶ 2635](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.75.1.5 IRpcMethodParameter.Size Property

Gets the size of the [IRpcMethodParameter](#) [[▶ 2635](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Size { get; }
```

Property Value

Type: Int32

The size.

Reference

[IRpcMethodParameter Interface](#) [[▶ 2635](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.75.1.6 IRpcMethodParameter.TypeName Property

Gets the Data type of the Parameter

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string TypeName { get; }
```

Property Value

Type: String

The type.

Reference

[IRpcMethodParameter Interface](#) [► 2635]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.75.1.7 IRpcMethodParameter.IsInput Property

Gets a value indicating whether this instance is input.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsInput { get; }
```

Property Value

Type: Boolean

true if this instance is input; otherwise, false.

Reference

[IRpcMethodParameter Interface](#) [► 2635]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.75.1.8 IRpcMethodParameter.IsOutput Property

Gets a value indicating whether this instance is output.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsOutput { get; }
```

Property Value

Type: Boolean

true if this instance is output; otherwise, false.

Reference

[IRpcMethodParameter Interface](#) [► 2635]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.76 IRpcMethodParameterCollection Interface

Interface IRpcMethodParameterCollection

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229




Syntax

C#











```
public interface IRpcMethodParameterCollection : IList<IRpcMethodParameter>,
    ICollection<IRpcMethodParameter>, IEnumerable<IRpcMethodParameter>, IEnumerable
```

The IRpcMethodParameterCollection type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.IRpcMethodParameter [▶ 2635]..)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.IRpcMethodParameter [▶ 2635]..)
	Item	Gets or sets the element at the specified index. (Inherited from IList.IRpcMethodParameter [▶ 2635]..)

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.IRpcMethodParameter [▶ 2635]..)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.IRpcMethodParameter [▶ 2635]..)
	Contains	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.IRpcMethodParameter [▶ 2635]..)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.IRpcMethodParameter [▶ 2635]..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.IRpcMethodParameter [▶ 2635]..)
	GetLengthIsParameter [▶ 2641]	Gets the corresponding LengthIs parameter.
	IndexOf	Determines the index of a specific item in the IList.T.. (Inherited from IList.IRpcMethodParameter [▶ 2635]..)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.IRpcMethodParameter [▶ 2635]..)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.IRpcMethodParameter [▶ 2635]..)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.IRpcMethodParameter [▶ 2635]..)




Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.76.1 IRpcMethodParameterCollection Properties

The IRpcMethodParameterCollection [▶ 2639] type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.IRpcMethodParameter [▶ 2635]..)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.IRpcMethodParameter [▶ 2635]..)
	Item	Gets or sets the element at the specified index. (Inherited from IList.IRpcMethodParameter [▶ 2635]..)

Reference











[IRpcMethodParameterCollection Interface \[▶ 2639\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.76.2 IRpcMethodParameterCollection Methods

The IRpcMethodParameterCollection [▶ 2639] type exposes the following members.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.IRpcMethodParameter [▶ 2635]..)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.IRpcMethodParameter [▶ 2635]..)
	Contains	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.IRpcMethodParameter [▶ 2635]..)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.IRpcMethodParameter [▶ 2635]..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.IRpcMethodParameter [▶ 2635]..)
	GetLengthsParameter [▶ 2641]	Gets the corresponding Lengths parameter.
	IndexOf	Determines the index of a specific item in the IList.T.. (Inherited from IList.IRpcMethodParameter [▶ 2635]..)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.IRpcMethodParameter [▶ 2635]..)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.IRpcMethodParameter [▶ 2635]..)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.IRpcMethodParameter [▶ 2635]..)

Reference

[IRpcMethodParameterCollection Interface \[▶ 2639\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.76.2.1 IRpcMethodParameterCollection.GetLengthsParameter Method

Gets the corresponding Lengths parameter.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IRpcMethodParameter? GetLengthIsParameter(
    IRpcMethodParameter parameter
)
```

Parameters

parameter	Type: TwinCAT.TypeSystem.IRpcMethodParameter [► 2635] The value parameter
-----------	--

Return Value

Type: [IRpcMethodParameter](#) [► 2635]

The LengthIs Parameter

Reference

[IRpcMethodParameterCollection Interface](#) [► 2639]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

[IRpcMethodParameter.LengthIsParameterIndex](#) [► 2637]

[IRpcMethodParameter.HasLengthIsParameter](#) [► 2636]

6.11.77 IRpcStructInstance Interface

Note: This API is now obsolete.

Interface IRpcStructInstance

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229





Syntax

C#

```
[ObsoleteAttribute("Use IStructInstance instead!")]
public interface IRpcStructInstance : IStructInstance,
    IInterfaceInstance, ISymbol, IAttributedInstance, IInstance, IBitSize,
    IRpcCallableInstance
```






The IRpcStructInstance type exposes the following members.

Properties

	Name	Description
	Attributes [► 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [► 2469].)
	BitSize [► 2473]	Gets the size of the IDataType [► 2475] in bits. (Inherited from IBitSize [► 2472].)
	ByteSize [► 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [► 2472].)
	Category [► 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [► 2691].)

	Name	Description
	Comment [▶ 2550]	Gets the comment of the Instance [▶ 2549] (Inherited from Instance [▶ 2549] .)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549] . (Inherited from Instance [▶ 2549] .)
	HasRpcMethods [▶ 3018]	Gets a value indicating whether this instance has RPC methods (Inherited from InterfaceInstance [▶ 3014] .)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from Instance [▶ 2549] .)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from Instance [▶ 2549] .)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472] .)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472] .)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is a container type. (Inherited from ISymbol [▶ 2691] .)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691] .)
	IsPointer [▶ 2552]	Indicates that the Instance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from Instance [▶ 2549] .)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691] .)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691] .)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691] .)
	IsReference [▶ 2552]	Indicates that the Instance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from Instance [▶ 2549] .)
	IsStatic [▶ 2553]	Gets a value indicating whether this Instance [▶ 2549] is static. (Inherited from Instance [▶ 2549] .)
	MemberInstances [▶ 3018]	Gets the member instances of the Struct Instance [▶ 3014] . (Inherited from InterfaceInstance [▶ 3014] .)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691] .)
	RpcMethods [▶ 2607]	Gets the Method descriptions for the IRpcCallableType [▶ 2624] (Inherited from IRpcCallableInstance [▶ 2606] .)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472] .)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691] .)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549] . (Inherited from Instance [▶ 2549] .)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from AttributedInstance [▶ 2469] .)

Methods

	Name	Description
	InvokeRpcMethod(String, .Object.) [▶ 2609]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)
	InvokeRpcMethod(String, .Object., .Object.) [▶ 2610]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)
	InvokeRpcMethodAsync [▶ 2614]	Invokes the specified RPC Method asynchronously (Inherited from IRpcCallableInstance [▶ 2606].)
	TryInvokeRpcMethod(String, .Object., Object.) [▶ 2619]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)
	TryInvokeRpcMethod(String, .Object., .Object., Object.) [▶ 2620]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]





[TwinCAT.TypeSystem.IStructInstance](#) [[▶ 2666](#)]

[TwinCAT.TypeSystem.IRpcCallableInstance](#) [[▶ 2606](#)]

6.11.77.1 IRpcStructInstance Properties

The [IRpcStructInstance](#) [[▶ 2642](#)] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)

	Name	Description
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549] .)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549] . (Inherited from IInstance [▶ 2549] .)
	HasRpcMethods [▶ 3018]	Gets a value indicating whether this instance has RPC methods (Inherited from IInterfaceInstance [▶ 3014] .)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549] .)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549] .)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472] .)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472] .)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is a container type. (Inherited from ISymbol [▶ 2691] .)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691] .)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549] .)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691] .)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691] .)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691] .)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549] .)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549] .)
	MemberInstances [▶ 3018]	Gets the member instances of the Struct Instance [▶ 3014] . (Inherited from IInterfaceInstance [▶ 3014] .)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691] .)
	RpcMethods [▶ 2607]	Gets the Method descriptions for the IRpcCallableType [▶ 2624] (Inherited from IRpcCallableInstance [▶ 2606] .)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472] .)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691] .)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549] . (Inherited from IInstance [▶ 2549] .)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469] .)

Reference






[IRpcStructInstance Interface \[▶ 2642\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.77.2 IRpcStructInstance Methods

The [IRpcStructInstance \[▸ 2642\]](#) type exposes the following members.

Methods

	Name	Description
	InvokeRpcMethod(String, Object.) [▸ 2609]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▸ 2606] .)
	InvokeRpcMethod(String, Object., Object.) [▸ 2610]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▸ 2606] .)
	InvokeRpcMethodAsync [▸ 2614]	Invokes the specified RPC Method asynchronously (Inherited from IRpcCallableInstance [▸ 2606] .)
	TryInvokeRpcMethod(String, Object., Object.) [▸ 2619]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▸ 2606] .)
	TryInvokeRpcMethod(String, Object., Object., Object.) [▸ 2620]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▸ 2606] .)

Extension Methods

	Name	Description
	ReferencesExternalData [▸ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▸ 3021] .)

Reference

[IRpcStructInstance Interface \[▸ 2642\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.78 IStringInstance Interface

Interface [IStringInstance](#)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax



C#

```
public interface IStringInstance : ISymbol,
    IAttributedInstance, IInstance, IBitSize
```

The IStringInstance type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full IDataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsFixedLength [▶ 2649]	Gets a value indicating whether this instance is a string of static length
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)

	Name	Description
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from Instance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from AttributedInstance [▶ 2469].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]













[TwinCAT.TypeSystem.ISymbol](#) [[▶ 2691](#)]













[TwinCAT.TypeSystem.IStringType](#) [[▶ 2661](#)]

6.11.78.1 IStringInstance Properties

The [IStringInstance](#) [[▶ 2646](#)] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from AttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the Instance [▶ 2549] (Inherited from Instance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549]. (Inherited from Instance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from Instance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from Instance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsFixedLength [▶ 2649]	Gets a value indicating whether this instance is a string of static length

	Name	Description
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Reference

[IStringInstance Interface](#) [[▶ 2646](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.78.1.1 IStringInstance.IsFixedLength Property

Gets a value indicating whether this instance is a string of static length

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsFixedLength { get; }
```

Property Value

Type: Boolean
true if this instance is of static length; otherwise, false.

Reference

[IStringInstance Interface](#) [[▶ 2646](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.78.2 IStringInstance Methods

The [IStringInstance](#) [▸ 2646] type exposes the following members.

Extension Methods

	Name	Description
	ReferencesExternalData [▸ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▸ 3021].)

Reference

[IStringInstance Interface](#) [▸ 2646]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.79 IStringMarshaler Interface

Common interface for marshalling ADS string values.

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#






```
public interface IStringMarshaler
```




The [IStringMarshaler](#) type exposes the following members.

Properties

	Name	Description
	Encoding [▸ 2660]	The encoding used by this marshaler.
	StringTerminatorSize [▸ 2661]	Gets the size of the string terminator '\0'

Methods

	Name	Description
	Marshal(String, Span.Byte.) [▸ 2653]	Marshals the specified string value.
	Marshal(IStringType, String, Span.Byte.) [▸ 2654]	Marshals the specified string value.
	MarshalSize(IStringType) [▸ 2655]	Gets the marshal size of the specified string type.
	MarshalSize(String) [▸ 2655]	Gets the marshal size of the string.
	MarshalSize(Encoding, Int32) [▸ 2656]	Gets the marshal size of the string given by its length.









	Name	Description
	Unmarshal(ReadOnlySpan.Byte., String.) [▶ 2658]	Unmarshals a string from memory/span.
	Unmarshal(IStringType, ReadOnlySpan.Byte., String.) [▶ 2659]	Unmarshals the specified string type.
	Unmarshal(ReadOnlySpan.Byte., Encoding, String.) [▶ 2659]	Unmarshals a string from memory/span.

Reference



[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.79.1 IStringMarshaler Methods

Methods

	Name	Description
	Marshal(String, Span.Byte.) [▶ 2653]	Marshals the specified string value.
	Marshal(IStringType, String, Span.Byte.) [▶ 2654]	Marshals the specified string value.
	MarshalSize(IStringType) [▶ 2655]	Gets the marshal size of the specified string type.
	MarshalSize(String) [▶ 2655]	Gets the marshal size of the string.
	MarshalSize(Encoding, Int32) [▶ 2656]	Gets the marshal size of the string given by its length.
	Unmarshal(ReadOnlySpan.Byte., String.) [▶ 2658]	Unmarshals a string from memory/span.
	Unmarshal(IStringType, ReadOnlySpan.Byte., String.) [▶ 2659]	Unmarshals the specified string type.
	Unmarshal(ReadOnlySpan.Byte., Encoding, String.) [▶ 2659]	Unmarshals a string from memory/span.

Reference[IStringMarshaler Interface \[► 2650\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.79.1.1 IStringMarshaler.Marshal Method****Overload List**

	Name	Description
	Marshal(String, Span.Byte.) [► 2653]	Marshals the specified string value.
	Marshal(IStringType, String, Span.Byte.) [► 2654]	Marshals the specified string value.

Reference[IStringMarshaler Interface \[► 2650\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.79.1.1.1 IStringMarshaler.Marshal Method (String, Span`1)****Namespace:** [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14**Syntax****C#**

```
int Marshal(
    string value,
    Span destination
)
```

Parametersvalue Type: [System.String](#)destination Type: [Span](#)**Return Value**Type: [Int32](#)**Reference**[IStringMarshaler Interface \[► 2650\]](#)[Marshal Overload \[► 2652\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.79.1.1.2 IStringMarshaler.Marshal Method (IStringType, String, Span`1)

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
int Marshal(
    IStringType type,
    string value,
    Span destination
)
```

Parameters

type Type: [TwinCAT.TypeSystem.IStringType](#) [► 2661]

value Type: [System.String](#)

destination Type: [Span](#)

Return Value

Type: [Int32](#)

Reference

[IStringMarshaler Interface](#) [► 2650]

[Marshal Overload](#) [► 2652]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.79.1.1.3 IStringMarshaler.Marshal Method (String, Span.Byte)

Marshals the specified string value.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Marshal(
    string value,
    Span<byte> destination
)
```

Parameters

value	Type: System.String The value.
destination	Type: System.Span.Byte . The destination span/memory.

Return Value

Type: Int32
Number of marshalled bytes.

Reference

[IStringMarshaler Interface](#) [[▶ 2650](#)]

[Marshal Overload](#) [[▶ 2652](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.79.1.1.4 IStringMarshaler.Marshal Method (IStringType, String, Span.Byte.)

Marshals the specified string value.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
int Marshal(
    IStringType type,
    string value,
    Span<byte> destination
)
```

Parameters

type	Type: TwinCAT.TypeSystem.IStringType [▶ 2661] The string type.
value	Type: System.String The string value.
destination	Type: System.Span.Byte. The destination span/memory.

Return Value

Type: Int32
The Number of marshalled bytes.


Reference



[IStringMarshaler Interface](#) [[▶ 2650](#)]

[Marshal Overload](#) [[▶ 2652](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.79.1.2 IStringMarshaler.MarshalSize Method**Overload List**

	Name	Description
	MarshalSize(IStringType) [▶ 2655]	Gets the marshal size of the specified string type.

	Name	Description
	MarshalSize(String) [▶ 2655]	Gets the marshal size of the string.
	MarshalSize(Encoding, Int32) [▶ 2656]	Gets the marshal size of the string given by its length.

Reference

[IStringMarshaler Interface](#) [▶ 2650]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.79.1.2.1 IStringMarshaler.MarshalSize Method (String)

Gets the marshal size of the string.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int MarshalSize(
    string value
)
```

Parameters

value	Type: System.String The string value.
-------	--

Return Value

Type: Int32
Marshalling size of the string.

Reference

[IStringMarshaler Interface](#) [▶ 2650]

[MarshalSize Overload](#) [▶ 2654]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.79.1.2.2 IStringMarshaler.MarshalSize Method (IStringType)

Gets the marshal size of the specified string type.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int MarshalSize(
    IStringType stringType
)
```

Parameters

stringType	Type: TwinCAT.TypeSystem.IStringType [▶ 2661] Type of the string.
------------	--

Return Value

Type: Int32
Marshalling size of the string

Reference

[IStringMarshaler Interface](#) [▶ 2650]

[MarshalSize Overload](#) [▶ 2654]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.79.1.2.3 IStringMarshaler.MarshalSize Method (Encoding, Int32)

Gets the marshal size of the string given by its length.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
int MarshalSize(
    Encoding encoding,
    int strLen
)
```

Parameters

encoding	Type: System.Text.Encoding The encoding.
strLen	Type: System.Int32 Length of the string.

Return Value

Type: Int32
Marshalling size of the string.

Reference




[IStringMarshaler Interface](#) [▶ 2650]

[MarshalSize Overload](#) [▶ 2654]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.79.1.3 IStringMarshaler.Unmarshal Method

Overload List

	Name	Description
	Unmarshal(ReadOnlySpan<Byte>, String.) [▶ 2658]	Unmarshals a string from memory/span.
	Unmarshal(IStringType, ReadOnlySpan<Byte>, String.) [▶ 2659]	Unmarshals the specified string type.
	Unmarshal(ReadOnlySpan<Byte>, Encoding, String.) [▶ 2659]	Unmarshals a string from memory/span.

Reference

[IStringMarshaler Interface](#) [▶ 2650]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.79.1.3.1 IStringMarshaler.Unmarshal Method (ReadOnlySpan<Byte>, Void, Byte)

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
int Unmarshal(
    ReadOnlySpan source,
    void encoding,
    byte value
)
```

Parameters

source Type: [ReadOnlySpan](#)

encoding Type: [System.Void](#)

value Type: [System.Byte](#)

Return Value

Type: [Int32](#)

Reference

[IStringMarshaler Interface](#) [▶ 2650]

[Unmarshal Overload \[▶ 2657\]](#)[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.79.1.3.2 IStringMarshaler.Unmarshal Method (IStringType, ReadOnlySpan`1, Void)

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
int Unmarshal(
    IStringType stringType,
    ReadOnlySpan source,
    void value
)
```

Parameters

stringType	Type: TwinCAT.TypeSystem.IStringType [▶ 2661]
source	Type: ReadOnlySpan
value	Type: System.Void

Return Value

Type: [Int32](#)

Reference

[IStringMarshaler Interface \[▶ 2650\]](#)[Unmarshal Overload \[▶ 2657\]](#)[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.79.1.3.3 IStringMarshaler.Unmarshal Method (ReadOnlySpan.Byte., String.)

Unmarshals a string from memory/span.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Unmarshal(
    ReadOnlySpan<byte> source,
    out string value
)
```

Parameters

source	Type: System.ReadOnlySpan.Byte. The source memory/span.
--------	--

value	Type: System.String. The unmarshalled value.
-------	---

Return Value

Type: Int32

The number of unmarshalled bytes.

Reference[IStringMarshaler Interface \[► 2650\]](#)[Unmarshal Overload \[► 2657\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.79.1.3.4 IStringMarshaler.Unmarshal Method (IStringType, ReadOnlySpan.Byte., String.)**

Unmarshals the specified string type.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
int Unmarshal(
    IStringType stringType,
    ReadOnlySpan<byte> source,
    out string value
)
```

Parameters

stringType	Type: TwinCAT.TypeSystem.IStringType [► 2661] Type of the string.
source	Type: System.ReadOnlySpan.Byte. The source memory/span.
value	Type: System.String. The unmarshaled string value.

Return Value

Type: Int32

The number of unmarshalled bytes.

Reference[IStringMarshaler Interface \[► 2650\]](#)[Unmarshal Overload \[► 2657\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.79.1.3.5 IStringMarshaler.Unmarshal Method (ReadOnlySpan.Byte., Encoding, String.)**

Unmarshals a string from memory/span.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Unmarshal(
    ReadOnlySpan<byte> source,
    Encoding encoding,
    out string value
)
```

Parameters

source	Type: System.ReadOnlySpan.Byte. The source memory/span.
encoding	Type: System.Text.Encoding The encoding.
value	Type: System.String. The unmarshalled value.

Return Value

Type: Int32

The number of unmarshalled bytes.

Reference

[IStringMarshaler Interface](#) [► 2650]

[Unmarshal Overload](#) [► 2657]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.79.2 IStringMarshaler Properties

The [IStringMarshaler](#) [► 2650] type exposes the following members.

Properties

	Name	Description
	Encoding [► 2660]	The encoding used by this marshaler.
	StringTerminatorSize [► 2661]	Gets the size of the string terminator '\0'

Reference

[IStringMarshaler Interface](#) [► 2650]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.79.2.1 IStringMarshaler.Encoding Property

The encoding used by this marshaler.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Encoding Encoding { get; }
```

Property Value

Type: Encoding
The encoding.

Reference

[IStringMarshaler Interface](#) [► 2650]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.79.2.2 IStringMarshaler.StringTerminatorSize Property

Gets the size of the string terminator '\0'

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int StringTerminatorSize { get; }
```

Property Value

Type: Int32
The size of the string terminator.

Reference

[IStringMarshaler Interface](#) [► 2650]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.80 IStringType Interface

Interface representing a string [IDataType](#) [► 2475]

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax















C#

```
public interface IStringType : IDataType,
    IBitSize
```

The IStringType type exposes the following members.

Properties

	Name	Description
	Attributes [► 2478]	Gets the attributes of the IDataType [► 2475] (Inherited from IDataType [► 2475].)

	Name	Description
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472] .)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472] .)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475] .)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475] .)
	Encoding [▶ 2664]	Gets the encoding of the String (Encoding.Default (STRING) or Encoding.UNICODE (WSTRING))
	FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from IDataType [▶ 2475] .)
	Id [▶ 2480]	Gets the ID of the Data Type (Inherited from IDataType [▶ 2475] .)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full Data Type but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472] .)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472] .)
	IsFixedLength [▶ 2665]	Gets a value indicating whether the string is of fixed length.
	Length [▶ 2665]	Gets the number of characters within the string (when fixed length).
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475] .)
	Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475] .)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472] .)

Extension Methods





	Name	Description
	IsArrayOfPrimitives. [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021] .)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021] .)
	IsContainer. [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021] .)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021] .)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021] .)
	IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021] .)

	Name	Description
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve . [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]







Also see about this










-  [IDataType.IsContainer](#) Property [▶ 2480]
-  [IDataType.IsPointer](#) Property [▶ 2481]
-  [IDataType.IsPrimitive](#) Property [▶ 2481]
-  [IDataType.IsReference](#) Property [▶ 2482]

6.11.80.1 IStringType Properties

The [IStringType](#) [▶ 2661] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataType [▶ 2475] (Inherited from IDataType [▶ 2475].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)
	Encoding [▶ 2664]	Gets the encoding of the String (Encoding.Default (STRING) or Encoding.UNICODE (WSTRING))





	Name	Description
	FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from IDataType [▶ 2475].)
	Id [▶ 2480]	Gets the ID of the Data Type (Inherited from IDataType [▶ 2475].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full Data Type but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsFixedLength [▶ 2665]	Gets a value indicating whether the string is of fixed length.
	Length [▶ 2665]	Gets the number of characters within the string (when fixed length).
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475].)
	Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Reference

[IStringType Interface](#) [[▶ 2661](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

Also see about this

-  [IDataType.IsContainer Property](#) [[▶ 2480](#)]
-  [IDataType.IsPointer Property](#) [[▶ 2481](#)]
-  [IDataType.IsPrimitive Property](#) [[▶ 2481](#)]
-  [IDataType.IsReference Property](#) [[▶ 2482](#)]

6.11.80.1.1 IStringType.Encoding Property

Gets the encoding of the String (Encoding.Default (STRING) or Encoding.UNICODE (WSTRING))

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Encoding Encoding { get; }
```

Property Value

Type: Encoding
The encoding.

Reference

[IStringType Interface](#) [[▶ 2661](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.80.1.2 IStringType.IsFixedLength Property

Gets a value indicating whether the string is of fixed length.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsFixedLength { get; }
```

Property Value

Type: Boolean

true if this instance is fixed length; otherwise, false.

Reference

[IStringType Interface](#) [[▶ 2661](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.80.1.3 IStringType.Length Property

Gets the number of characters within the string (when fixed length).

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Length { get; }
```

Property Value

Type: Int32

The length if fixed length, otherwise -1

Reference

[IStringType Interface](#) [[▶ 2661](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.80.2 IStringType Methods

The [IStringType](#) [[▶ 2661](#)] type exposes the following members.

Extension Methods

	Name	Description
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)

Name	Description
IsContainer. [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive. [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
Resolve. [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[IStringType Interface](#) [[▶ 2661](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.81 IStructInstance Interface

Interface representing an instance of a [IStructType](#) [[▶ 2671](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax





C#

```
public interface IStructInstance : IInterfaceInstance,
    ISymbol, IAttributedInstance, IInstance, IBitSize, IRpcCallableInstance
```

The IStructInstance type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	HasRpcMethods [▶ 3018]	Gets a value indicating whether this instance has RPC methods (Inherited from IInterfaceInstance [▶ 3014].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($BitSize \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	MemberInstances [▶ 3018]	Gets the member instances of the Struct Instance [▶ 3014]. (Inherited from IInterfaceInstance [▶ 3014].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	RpcMethods [▶ 2607]	Gets the Method descriptions for the IRpcCallableType [▶ 2624] (Inherited from IRpcCallableInstance [▶ 2606].)

	Name	Description
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from Instance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from AttributedInstance [▶ 2469].)

Methods

	Name	Description
	InvokeRpcMethod (String, .Object.) [▶ 2609]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)
	InvokeRpcMethod (String, .Object., .Object.) [▶ 2610]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)
	InvokeRpcMethodAsync [▶ 2614]	Invokes the specified RPC Method asynchronously (Inherited from IRpcCallableInstance [▶ 2606].)
	TryInvokeRpcMethod (String, .Object., Object.) [▶ 2619]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)
	TryInvokeRpcMethod (String, .Object., Object., Object.) [▶ 2620]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)


Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]


6.11.81.1 IStructInstance Properties

The [IStructInstance](#) [[▶ 2666](#)] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from AttributedInstance [▶ 2469].)

	Name	Description
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472] .)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472] .)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691] .)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549] .)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549] . (Inherited from IInstance [▶ 2549] .)
	HasRpcMethods [▶ 3018]	Gets a value indicating whether this instance has RPC methods (Inherited from IInterfaceInstance [▶ 3014] .)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549] .)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549] .)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full IDataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472] .)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472] .)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691] .)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691] .)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549] .)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691] .)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691] .)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691] .)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549] .)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549] .)
	MemberInstances [▶ 3018]	Gets the member instances of the Struct Instance [▶ 3014] . (Inherited from IInterfaceInstance [▶ 3014] .)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691] .)
	RpcMethods [▶ 2607]	Gets the Method descriptions for the IRpcCallableType [▶ 2624] (Inherited from IRpcCallableInstance [▶ 2606] .)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472] .)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691] .)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549] . (Inherited from IInstance [▶ 2549] .)

	Name	Description
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Reference

[IStructInstance Interface](#) [▶ [2666](#)]

[TwinCAT.TypeSystem Namespace](#) [▶ [2083](#)]

6.11.81.1.1 IStructInstance.HasRpcMethods Property

Gets a value indicating whether this instance has RPC methods

Namespace: [TwinCAT.TypeSystem](#) [▶ [2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
bool HasRpcMethods { get; }
```

Property Value

Type: [Boolean](#)

true if this instance has RPC methods; otherwise, false.

Remarks

If the struct instance supports RPC Methods, then the instance class is also supporting [IRpcStructInstance](#) [▶ [2642](#)]:

Reference

[IStructInstance Interface](#) [▶ [2666](#)]

[TwinCAT.TypeSystem Namespace](#) [▶ [2083](#)]

[TwinCAT.TypeSystem.IRpcStructInstance](#) [▶ [2642](#)]

[TwinCAT.TypeSystem.IRpcMethod](#) [▶ [2625](#)]

[TwinCAT.TypeSystem.IRpcMethodParameter](#) [▶ [2635](#)]

6.11.81.1.2 IStructInstance.MemberInstances Property

Gets the member instances of the [Struct Instance](#) [▶ [2666](#)].

Namespace: [TwinCAT.TypeSystem](#) [▶ [2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
ISymbolCollection<ISymbol> MemberInstances { get; }
```

Property Value

Type: [ISymbolCollection](#) [[▶ 2700](#)].[ISymbol](#) [[▶ 2691](#)].
The member instances.

Reference

[IStructInstance Interface](#) [[▶ 2666](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.81.2 IStructInstance Methods

The [IStructInstance](#) [[▶ 2666](#)] type exposes the following members.

Methods

	Name	Description
	InvokeRpcMethod(String, Object.) [▶ 2609]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)
	InvokeRpcMethod(String, Object., Object.) [▶ 2610]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)
	InvokeRpcMethodAsync [▶ 2614]	Invokes the specified RPC Method asynchronously (Inherited from IRpcCallableInstance [▶ 2606].)
	TryInvokeRpcMethod(String, Object., Object.) [▶ 2619]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)
	TryInvokeRpcMethod(String, Object., Object., Object.) [▶ 2620]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[IStructInstance Interface](#) [[▶ 2666](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.82 IStructType Interface

Interface representing Struct data types

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax



C#

```
public interface IStructType : IInterfaceType,
    IDataTypes, IBitSize, IRpcCallableType
```

The IStructType type exposes the following members.

Properties

	Name	Description
	AllMembers [▶ 2987]	Gets all members (down the derivation hierarchy) (Inherited from IInterfaceType [▶ 2983].)
	Attributes [▶ 2478]	Gets the attributes of the IDataType [▶ 2475] (Inherited from IDataType [▶ 2475].)
	BaseType [▶ 2988]	Gets the structs Base Type (Null if not derived). (Inherited from IInterfaceType [▶ 2983].)
	BaseTypeName [▶ 2988]	Gets the the Name of the Base class (if derived) (Inherited from IInterfaceType [▶ 2983].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)
	FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from IDataType [▶ 2475].)
	HasRpcMethods [▶ 2986]	Gets a value indicating whether this instance has RPC methods (Struct types only) (Inherited from IInterfaceType [▶ 2983].)
	Id [▶ 2480]	Gets the ID of the Data Type (Inherited from IDataType [▶ 2475].)
	InterfaceImplementationNames [▶ 2987]	Gets the names of the interfaces, this IDataType [▶ 2475] implements. (Inherited from IInterfaceType [▶ 2983].)
	InterfaceImplementations [▶ 2987]	Gets the resolved interface types, this IDataType [▶ 2475] implements. (Inherited from IInterfaceType [▶ 2983].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full Data Type but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	Members [▶ 2988]	Gets a readonly collection of the Members [▶ 2561] of the IInterfaceType [▶ 2983]. (Inherited from IInterfaceType [▶ 2983].)
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475].)
	Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475].)

	Name	Description
	RpcMethods [▶ 2624]	Gets the Method descriptions for the IRpcCallableType [▶ 2624] (Inherited from IRpcCallableType [▶ 2624].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)





Extension Methods

	Name	Description
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer . [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive . [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive ve. [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve . [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

Also see about this



-  [IDataType.IsContainer Property \[▸ 2480\]](#)
-  [IDataType.IsPointer Property \[▸ 2481\]](#)
-  [IDataType.IsPrimitive Property \[▸ 2481\]](#)
-  [IDataType.IsReference Property \[▸ 2482\]](#)

6.11.82.1 IStructType Properties

The [IStructType \[▸ 2671\]](#) type exposes the following members.

Properties

	Name	Description
	AllMembers [▸ 2987]	Gets all members (down the derivation hierarchy) (Inherited from IInterfaceType [▸ 2983].)
	Attributes [▸ 2478]	Gets the attributes of the IDataType [▸ 2475] (Inherited from IDataType [▸ 2475].)
	BaseType [▸ 2988]	Gets the structs Base Type (Null if not derived). (Inherited from IInterfaceType [▸ 2983].)
	BaseTypeName [▸ 2988]	Gets the the Name of the Base class (if derived) (Inherited from IInterfaceType [▸ 2983].)
	BitSize [▸ 2473]	Gets the size of the IDataType [▸ 2475] in bits. (Inherited from IBitSize [▸ 2472].)
	ByteSize [▸ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▸ 2472].)
	Category [▸ 2478]	Gets the Data Type category (Inherited from IDataType [▸ 2475].)
	Comment [▸ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▸ 2475].)
	FullName [▸ 2479]	Gets the full name of the IDataType [▸ 2475] (Namespace + Name) (Inherited from IDataType [▸ 2475].)
	HasRpcMethods [▸ 2986]	Gets a value indicating whether this instance has RPC methods (Struct types only) (Inherited from IInterfaceType [▸ 2983].)
	Id [▸ 2480]	Gets the ID of the DataType (Inherited from IDataType [▸ 2475].)
	InterfaceImplementationNames [▸ 2987]	Gets the names of the interfaces, this IDataType [▸ 2475] implements. (Inherited from IInterfaceType [▸ 2983].)
	InterfaceImplementations [▸ 2987]	Gets the resolved interface types, this IDataType [▸ 2475] implements. (Inherited from IInterfaceType [▸ 2983].)
	IsBitType [▸ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▸ 2472].)
	IsByteAligned [▸ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▸ 2472].)
	Members [▸ 2988]	Gets a readonly collection of the Members [▸ 2561] of the IInterfaceType [▸ 2983]. (Inherited from IInterfaceType [▸ 2983].)
	Name [▸ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▸ 2475].)
	Namespace [▸ 2482]	Gets the namespace string within the IDataType [▸ 2475] exists. (Inherited from IDataType [▸ 2475].)

	Name	Description
	RpcMethods [▶ 2624]	Gets the Method descriptions for the IRpcCallableType [▶ 2624] (Inherited from IRpcCallableType [▶ 2624].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Reference

[IStructType Interface](#) [▶ 2671]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

Also see about this

 [IDataType.IsContainer Property](#) [▶ 2480]

 [IDataType.IsPointer Property](#) [▶ 2481]

 [IDataType.IsPrimitive Property](#) [▶ 2481]

 [IDataType.IsReference Property](#) [▶ 2482]

6.11.82.1.1 IStructType.AllMembers Property

Gets all members (down the derivation hierarchy)

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
IMemberCollection AllMembers { get; }
```

Property Value

Type: [IMemberCollection](#) [▶ 2565]

All members.

Reference

[IStructType Interface](#) [▶ 2671]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.82.1.2 IStructType.BaseType Property

Gets the structs Base Type (Null if not derived).

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
IDataType? BaseType { get; }
```

Property Value

Type: [IDataType](#) [▶ 2475]

Reference

[IStructType Interface \[► 2671\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.82.1.3 IStructType.BaseTypeName Property

Gets the the Name of the Base class (if derived)

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
string BaseTypeName { get; }
```

Property Value

Type: [String](#)

Empty if not derived.

Reference

[IStructType Interface \[► 2671\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.82.1.4 IStructType.HasRpcMethods Property

Gets a value indicating whether this instance has RPC methods (Struct types only)

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
bool HasRpcMethods { get; }
```

Property Value

Type: [Boolean](#)

true if this instance has RPC methods; otherwise, false.

Remarks

The DataType (Structure) must be marked with the PlcAttribute 'TcRpcEnable' to enable RpcMethods, otherwise RpcMethods are not passed through to the ADS symbolic information.

Reference

[IStructType Interface \[► 2671\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.82.1.5 IStructType.Members Property

Gets a readonly collection of the [Members](#) [▶ 2561] of the [IStructType](#) [▶ 2671].

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
IMemberCollection Members { get; }
```

Property Value

Type: [IMemberCollection](#) [▶ 2565]

The members as readonly collection.

Remarks

If the [IStructType](#) [▶ 2671] is derived, only the extended members are returned. To get all supported members down the inheritance chain, use the [AllMembers](#) [▶ 2675] property.

Reference

[IStructType Interface](#) [▶ 2671]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.82.2 IStructType Methods

The [IStructType](#) [▶ 2671] type exposes the following members.

Extension Methods

Name	Description
IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)

	Name	Description
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[IStructType Interface](#) [[▶ 2671](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.83 IStructValue Interface

Interface [IStructValue](#)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229






Syntax



C#

```
public interface IStructValue : IValue
```









The [IStructValue](#) type exposes the following members.

Properties

	Name	Description
	Age [▶ 2746]	Gets the age of the value (last successful read of the value) (Inherited from IValue [▶ 2745].)
	CachedRaw [▶ 2746]	Gets the cached Raw internal Data. (Inherited from IValue [▶ 2745].)
	DataType [▶ 2747]	Gets the data type bound to this IValue [▶ 2745] (Inherited from IValue [▶ 2745].)
	IsPrimitive [▶ 2747]	Gets a value indicating whether this IValue [▶ 2745] is a primitive value. (Inherited from IValue [▶ 2745].)
	Symbol [▶ 2748]	Gets the symbol bound to this IValue [▶ 2745]. (Inherited from IValue [▶ 2745].)

	Name	Description
	TimeStamp [▶ 2748]	Gets the Time stamp of the last successful read of the Value (local user time, UTC) (Inherited from IValue [▶ 2745].)
	UpdateMode [▶ 2748]	Gets the update mode (not implemented yet) (Inherited from IValue [▶ 2745].)

Methods

	Name	Description
	Read [▶ 2749]	Reads the value (via ADS) (Inherited from IValue [▶ 2745].)
	ReadAsync [▶ 2750]	Reads the value (via ADS) (Inherited from IValue [▶ 2745].)
	ResolveValue [▶ 2750]	Resolves the Value object to its primitive value. (Inherited from IValue [▶ 2745].)
	TryGetMemberValue [▶ 2680]	Tries to get a property/Member value.
	TryResolveValue [▶ 2751]	Tries to resolves the Value object to its primitive value. (Inherited from IValue [▶ 2745].)
	TrySetMemberValue [▶ 2681]	Tries to Set a Member/Property Value
	Write [▶ 2751]	Writes the value (via ADS) (Inherited from IValue [▶ 2745].)
	WriteAsync [▶ 2751]	Writes the value (via ADS) (Inherited from IValue [▶ 2745].)

Reference








[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

[TwinCAT.TypeSystem.IValue](#) [[▶ 2745](#)]

6.11.83.1 IStructValue Properties

The [IStructValue](#) [[▶ 2678](#)] type exposes the following members.

Properties

	Name	Description
	Age [▶ 2746]	Gets the age of the value (last successful read of the value) (Inherited from IValue [▶ 2745].)
	CachedRaw [▶ 2746]	Gets the cached Raw internal Data. (Inherited from IValue [▶ 2745].)
	DataType [▶ 2747]	Gets the data type bound to this IValue [▶ 2745] (Inherited from IValue [▶ 2745].)
	IsPrimitive [▶ 2747]	Gets a value indicating whether this IValue [▶ 2745] is a primitive value. (Inherited from IValue [▶ 2745].)
	Symbol [▶ 2748]	Gets the symbol bound to this IValue [▶ 2745]. (Inherited from IValue [▶ 2745].)
	TimeStamp [▶ 2748]	Gets the Time stamp of the last successful read of the Value (local user time, UTC) (Inherited from IValue [▶ 2745].)
	UpdateMode [▶ 2748]	Gets the update mode (not implemented yet) (Inherited from IValue [▶ 2745].)

Reference









[IStructValue Interface \[▸ 2678\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.83.2 IStructValue Methods

The [IStructValue \[▸ 2678\]](#) type exposes the following members.

Methods

	Name	Description
	Read [▸ 2749]	Reads the value (via ADS) (Inherited from IValue [▸ 2745].)
	ReadAsync [▸ 2750]	Reads the value (via ADS) (Inherited from IValue [▸ 2745].)
	ResolveValue [▸ 2750]	Resolves the Value object to its primitive value. (Inherited from IValue [▸ 2745].)
	TryGetMemberValue [▸ 2680]	Tries to get a property/Member value.
	TryResolveValue [▸ 2751]	Tries to resolves the Value object to its primitive value. (Inherited from IValue [▸ 2745].)
	TrySetMemberValue [▸ 2681]	Tries to Set a Member/Property Value
	Write [▸ 2751]	Writes the value (via ADS) (Inherited from IValue [▸ 2745].)
	WriteAsync [▸ 2751]	Writes the value (via ADS) (Inherited from IValue [▸ 2745].)

Reference

[IStructValue Interface \[▸ 2678\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.83.2.1 IStructValue.TryGetMemberValue Method

Tries to get a property/Member value.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool TryGetMemberValue(
    string name,
    out Object?? value
)
```

Parameters

name	Type: System.String The name of the member
value	Type: System.Object. The value.

Return Value

Type: Boolean
true if succeeded, otherwise false otherwise.

Reference

[IStructValue Interface \[▸ 2678\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.83.2.2 IStructValue.TrySetMemberValue Method

Tries to Set a Member/Property Value

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool TrySetMemberValue(  
    string name,  
    Object value  
)
```

Parameters

name	Type: System.String The name of the member
value	Type: System.Object The value.

Return Value

Type: Boolean
true if succeeded, otherwise false otherwise.

Reference

[IStructValue Interface \[▸ 2678\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.84 ISubRangeType Interface

Interface representing a SubRange type

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229















Syntax

C#

```
public interface ISubRangeType : IDataTypes,  
    IBitSize
```

The ISubRangeType type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataType [▶ 2475] (Inherited from IDataType [▶ 2475].)
	BaseType [▶ 2684]	Gets the the base type of the ISubRangeType
	BaseTypeName [▶ 2685]	Gets the name of the base type.
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)
	FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from IDataType [▶ 2475].)
	Id [▶ 2480]	Gets the ID of the Data Type (Inherited from IDataType [▶ 2475].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full Data Type but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475].)
	Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Extension Methods





	Name	Description
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive [▶ 3033]	Overloaded.

	Name	Description
		Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]






Also see about this










-  [IDataType.IsContainer Property](#) [[▶ 2480](#)]
-  [IDataType.IsPointer Property](#) [[▶ 2481](#)]
-  [IDataType.IsPrimitive Property](#) [[▶ 2481](#)]
-  [IDataType.IsReference Property](#) [[▶ 2482](#)]

6.11.84.1 ISubRangeType Properties

The [ISubRangeType](#) [[▶ 2681](#)] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataType [▶ 2475] (Inherited from IDataType [▶ 2475].)
	BaseType [▶ 2684]	Gets the the base type of the ISubRangeType [▶ 2681]
	BaseTypeName [▶ 2685]	Gets the name of the base type.
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)





	Name	Description
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)
	FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from IDataType [▶ 2475].)
	Id [▶ 2480]	Gets the ID of the Data Type (Inherited from IDataType [▶ 2475].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full Data Type but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475].)
	Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Reference

[ISubRangeType Interface](#) [[▶ 2681](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

Also see about this

-  [IDataType.IsContainer Property](#) [[▶ 2480](#)]
-  [IDataType.IsPointer Property](#) [[▶ 2481](#)]
-  [IDataType.IsPrimitive Property](#) [[▶ 2481](#)]
-  [IDataType.IsReference Property](#) [[▶ 2482](#)]

6.11.84.1.1 ISubRangeType.BaseType Property

Gets the the base type of the [ISubRangeType](#) [[▶ 2681](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDataType? BaseType { get; }
```

Property Value

Type: [IDataType](#) [[▶ 2475](#)]

The base datat type or **null** if not resolved.

Reference

[ISubRangeType Interface](#) [[▶ 2681](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.84.1.2 ISubRangeType.BaseTypeName Property

Gets the name of the base type.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string BaseTypeName { get; }
```

Property Value

Type: String

The name of the base type.

Reference

[ISubRangeType Interface](#) [[▶ 2681](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.84.2 ISubRangeType Methods

The [ISubRangeType](#) [[▶ 2681](#)] type exposes the following members.

Extension Methods

	Name	Description
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)

	Name	Description
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[ISubRangeType Interface](#) [[▶ 2681](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.85 ISubRangeType.T. Interface

Interface representing a SubRange type

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#





```
public interface ISubRangeType<T> : ISubRangeType,
    IDataType, IBitSize
where T : struct, new()
```













Type Parameters

T

The ISubRangeType.T. type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataType [▶ 2475] (Inherited from IDataType [▶ 2475].)
	BaseType [▶ 2684]	Gets the the base type of the ISubRangeType [▶ 2681] (Inherited from ISubRangeType [▶ 2681].)
	BaseTypeName [▶ 2685]	Gets the name of the base type. (Inherited from ISubRangeType [▶ 2681].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)

	Name	Description
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)
	FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from IDataType [▶ 2475].)
	Id [▶ 2480]	Gets the ID of the DataType (Inherited from IDataType [▶ 2475].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	LowerBound [▶ 2689]	Gets the lower bound.
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475].)
	Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	UpperBound [▶ 2690]	Gets the upper bound.

Extension Methods





	Name	Description
	IsArrayOfPrimitives. [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer. [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive. [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)

	Name	Description
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]








Also see about this










-  [IDataType.IsContainer Property](#) [[▶ 2480](#)]
-  [IDataType.IsPointer Property](#) [[▶ 2481](#)]
-  [IDataType.IsPrimitive Property](#) [[▶ 2481](#)]
-  [IDataType.IsReference Property](#) [[▶ 2482](#)]

6.11.85.1 ISubRangeType.T. Properties

The [ISubRangeType.T](#) [[▶ 2686](#)] generic type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataType [▶ 2475] (Inherited from IDataType [▶ 2475].)
	BaseType [▶ 2684]	Gets the the base type of the ISubRangeType [▶ 2681] (Inherited from ISubRangeType [▶ 2681].)
	BaseTypeName [▶ 2685]	Gets the name of the base type. (Inherited from ISubRangeType [▶ 2681].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)





	Name	Description
	FullName [▸ 2479]	Gets the full name of the IDataType [▸ 2475] (Namespace + Name) (Inherited from IDataType [▸ 2475] .)
	Id [▸ 2480]	Gets the ID of the Data Type (Inherited from IDataType [▸ 2475] .)
	IsBitType [▸ 2474]	Gets a value indicating whether this instance is not basing on a full Data Type but instead of some sort of bit mapping (Inherited from IBitSize [▸ 2472] .)
	IsByteAligned [▸ 2475]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from IBitSize [▸ 2472] .)
	LowerBound [▸ 2689]	Gets the lower bound.
	Name [▸ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▸ 2475] .)
	Namespace [▸ 2482]	Gets the namespace string within the IDataType [▸ 2475] exists. (Inherited from IDataType [▸ 2475] .)
	Size [▸ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▸ 2474] (Inherited from IBitSize [▸ 2472] .)
	UpperBound [▸ 2690]	Gets the upper bound.

Reference

[ISubRangeType.T. Interface \[▸ 2686\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

Also see about this

-  [IDataType.IsContainer Property \[▸ 2480\]](#)
-  [IDataType.IsPointer Property \[▸ 2481\]](#)
-  [IDataType.IsPrimitive Property \[▸ 2481\]](#)
-  [IDataType.IsReference Property \[▸ 2482\]](#)

6.11.85.1.1 ISubRangeType.T..LowerBound Property

Gets the lower bound.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
T LowerBound { get; }
```

Property Value

Type: [T \[▸ 2686\]](#)
The lower bound.

Reference

[ISubRangeType.T. Interface \[▸ 2686\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.85.1.2 ISubRangeType.T.UpperBound Property

Gets the upper bound.

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
T UpperBound { get; }
```

Property Value

Type: [T](#) [[▸ 2686](#)]

The upper bound.

Reference

[ISubRangeType.T. Interface](#) [[▸ 2686](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.85.2 ISubRangeType.T. Methods

The [ISubRangeType.T.](#) [[▸ 2686](#)] generic type exposes the following members.

Extension Methods

	Name	Description
	IsArrayOfPrimitives. [▸ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▸ 3021].)
	IsArrayOfPrimitives(Boolean) [▸ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▸ 3021].)
	IsContainer. [▸ 3027]	Overloaded. Gets a value indicating whether this IDataType [▸ 2475] is a container type (Defined by DataTypeExtension [▸ 3021].)
	IsContainer(Boolean) [▸ 3028]	Overloaded. Gets a value indicating whether this IDataType [▸ 2475] is a container type (Defined by DataTypeExtension [▸ 3021].)
	IsPointer [▸ 3031]	Gets a value indicating whether this IDataType [▸ 2475] is a pointer type (Defined by DataTypeExtension [▸ 3021].)
	IsPrimitive. [▸ 3033]	Overloaded. Gets a value indicating whether this IDataType [▸ 2475] is primitive (Defined by DataTypeExtension [▸ 3021].)
	IsPrimitive(Boolean) [▸ 3034]	Overloaded. Gets a value indicating whether this IDataType [▸ 2475] is primitive (Defined by DataTypeExtension [▸ 3021].)
	IsReference [▸ 3036]	Gets a value indicating whether this IDataType [▸ 2475] is a reference type (Defined by DataTypeExtension [▸ 3021].)

	Name	Description
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[ISubRangeType.T. Interface](#) [[▶ 2686](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.86 ISymbol Interface

Interface specifying Symbols (

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229







Syntax


















C#

```
public interface ISymbol : IAttributedInstance,
    IInstance, IBitSize
```

The ISymbol type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)

	Name	Description
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is a container type.
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol is persistent.
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type.
	IsReadOnly [▶ 2695]	Indicates that this instance is read only.
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive.
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Parent [▶ 2696]	Gets the parent Symbol
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)






















Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.86.1 ISymbol Properties

The [ISymbol](#) [▶ 2691] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type.
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent.
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type.
	IsReadOnly [▶ 2695]	Indicates that this instance is read only.
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive.
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Parent [▶ 2696]	Gets the parent Symbol
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691]
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Reference

[ISymbol Interface](#) [► 2691]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.86.1.1 ISymbol.Category Property

Gets the Symbol/Datatype Category

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
DataTypeCategory Category { get; }
```

Property Value

Type: [DataTypeCategory](#) [► 2111]

The category.

Reference

[ISymbol Interface](#) [► 2691]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.86.1.2 ISymbol.IsContainerType Property

Gets a value indicating whether this Symbol is acontainer type.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsContainerType { get; }
```

Property Value

Type: Boolean

true if this instance is container type; otherwise, false.

Reference

[ISymbol Interface](#) [► 2691]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.86.1.3 ISymbol.IsPersistent Property

Gets a value indicating whether this [ISymbol](#) [► 2691] is persistent.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsPersistent { get; }
```

Property Value

Type: Boolean
true if this instance is persistent; otherwise, false.

Reference

[ISymbol Interface](#) [► 2691]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.86.1.4 ISymbol.IsPrimitiveType Property

Gets a value indicating whether this instance is a primitive type.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsPrimitiveType { get; }
```

Property Value

Type: Boolean
true if this instance is primitive type; otherwise, false.

Reference

[ISymbol Interface](#) [► 2691]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.86.1.5 ISymbol.IsReadOnly Property

Indicates that this instance is read only.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsReadOnly { get; }
```

Property Value

Type: Boolean

Remarks

Actually, this Flag is restricted to TcCOM-Objects readonly Parameters. Within the PLC this is used for the ApplicationName and ProjectName of PLC instances. Write-Access on these Modules will create an [DeviceAccessDenied](#) [► 664] error.

Reference

[ISymbol Interface](#) [► 2691]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.86.1.6 ISymbol.IsRecursive Property

Gets a value indicating whether this instance is recursive.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsRecursive { get; }
```

Property Value

Type: Boolean

true if this instance is recursive; otherwise, false.

Reference

[ISymbol Interface](#) [► 2691]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.86.1.7 ISymbol.Parent Property

Gets the parent Symbol

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISymbol? Parent { get; }
```

Property Value

Type: [ISymbol](#) [► 2691]

The parent.

Reference

[ISymbol Interface](#) [► 2691]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.86.1.8 ISymbol.SubSymbols Property

Gets the SubSymbols of the [ISymbol](#) [► 2691]

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISymbolCollection<ISymbol> SubSymbols { get; }
```

Property Value

Type: [ISymbolCollection](#) [[▶ 2700](#)].[ISymbol](#) [[▶ 2691](#)].

Remarks

Used for Array, Struct, Pointer and Reference instances. Otherwise empty

Reference

[ISymbol Interface](#) [[▶ 2691](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.86.2 ISymbol Methods

The [ISymbol](#) [[▶ 2691](#)] type exposes the following members.

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[ISymbol Interface](#) [[▶ 2691](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.87 ISymbolCollection Interface

Interface [ISymbolCollection](#) Implements the [IInstanceCollection.T.](#) [[▶ 2554](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229



Syntax




C#

```
public interface ISymbolCollection : ISymbolCollection<ISymbol>,
    IInstanceCollection<ISymbol>, IList<ISymbol>, ICollection<ISymbol>,
    IEnumerable<ISymbol>, IEnumerable
```
















The [ISymbolCollection](#) type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T. (Inherited from ICollection.ISymbol [▶ 2691].)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.ISymbol [▶ 2691].)

	Name	Description
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.ISymbol [▶ 2691]..)
	Item.String. [▶ 2556]	Gets the Instance [▶ 2549] with the specified instance path. (Inherited from InstanceCollection.T. [▶ 2554].)
	Mode [▶ 2556]	Gets the InstanceCollectionMode [▶ 2573]. (Inherited from InstanceCollection.T. [▶ 2554].)

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.ISymbol [▶ 2691]..)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.ISymbol [▶ 2691]..)
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.ISymbol [▶ 2691]..)
	Contains(String) [▶ 2558]	Determines whether this collection contains an instance with the specified instance path. (Inherited from InstanceCollection.T. [▶ 2554].)
	ContainsName [▶ 2558]	Determines whether this collection contains an instance with the specified instance name. (Inherited from InstanceCollection.T. [▶ 2554].)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.ISymbol [▶ 2691]..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.ISymbol [▶ 2691]..)
	GetInstance [▶ 2559]	Gets the Instance [▶ 2549] by instance path. (Inherited from InstanceCollection.T. [▶ 2554].)
	GetInstanceByName [▶ 2559]	Gets the Instance [▶ 2549] by instance name. (Inherited from InstanceCollection.T. [▶ 2554].)
	IndexOf	Determines the index of a specific item in the IList.T.. (Inherited from IList.ISymbol [▶ 2691]..)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.ISymbol [▶ 2691]..)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.ISymbol [▶ 2691]..)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.ISymbol [▶ 2691]..)
	TryGetInstance [▶ 2560]	Tries to get the specified instance. (Inherited from InstanceCollection.T. [▶ 2554].)
	TryGetInstanceByName [▶ 2560]	Tries to get the specified instance by name. (Inherited from InstanceCollection.T. [▶ 2554].)

Reference






[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

[TwinCAT.TypeSystem.InstanceCollection.T. \[▶ 2554\]](#)

6.11.87.1 ISymbolCollection Properties

The [ISymbolCollection \[▶ 2697\]](#) type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.ISymbol [▶ 2691]..)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.ISymbol [▶ 2691]..)
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.ISymbol [▶ 2691]..)
	Item.String_ [▶ 2556]	Gets the Instance [▶ 2549] with the specified instance path. (Inherited from InstanceCollection.T. [▶ 2554].)
	Mode [▶ 2556]	Gets the InstanceCollectionMode [▶ 2573]. (Inherited from InstanceCollection.T. [▶ 2554].)

Reference














[ISymbolCollection Interface \[▶ 2697\]](#)



[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.87.2 ISymbolCollection Methods

The [ISymbolCollection \[▶ 2697\]](#) type exposes the following members.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.ISymbol [▶ 2691]..)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.ISymbol [▶ 2691]..)
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.ISymbol [▶ 2691]..)
	Contains(String) [▶ 2558]	Determines whether this collection contains an instance with the specified instance path. (Inherited from InstanceCollection.T. [▶ 2554].)
	ContainsName [▶ 2558]	Determines whether this collection contains an instance with the specified instance name. (Inherited from InstanceCollection.T. [▶ 2554].)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.ISymbol [▶ 2691]..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.ISymbol [▶ 2691]..)
	GetInstance [▶ 2559]	Gets the Instance [▶ 2549] by instance path. (Inherited from InstanceCollection.T. [▶ 2554].)
	GetInstanceByName [▶ 2559]	Gets the Instance [▶ 2549] by instance name. (Inherited from InstanceCollection.T. [▶ 2554].)
	IndexOf	Determines the index of a specific item in the IList.T.. (Inherited from IList.ISymbol [▶ 2691]..)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.ISymbol [▶ 2691]..)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.ISymbol [▶ 2691]..)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.ISymbol [▶ 2691]..)

	Name	Description
	TryGetInstance [▶ 2560]	Tries to get the specified instance. (Inherited from IInstanceCollection.T. [▶ 2554].)
	TryGetInstanceByName [▶ 2560]	Tries to get the specified instance by name. (Inherited from IInstanceCollection.T. [▶ 2554].)

Reference

[ISymbolCollection Interface](#) [▶ 2697]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.88 ISymbolCollection.T. Interface

Interface ISymbolCollection

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#






```
public interface ISymbolCollection<T> : IInstanceCollection<T>,
    IList<T>, ICollection<T>, IEnumerable<T>, IEnumerable
where T : class, ISymbol
```

Type Parameters





T












The ISymbolCollection.T. type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T. (Inherited from ICollection.T.)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.T.)
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.T.)
	Item.String. [▶ 2556]	Gets the IInstance [▶ 2549] with the specified instance path. (Inherited from IInstanceCollection.T. [▶ 2554].)
	Mode [▶ 2556]	Gets the InstanceCollectionMode [▶ 2573]. (Inherited from IInstanceCollection.T. [▶ 2554].)

Methods

	Name	Description
	Add	Adds an item to the ICollection.T. (Inherited from ICollection.T.)
	Clear	Removes all items from the ICollection.T. (Inherited from ICollection.T.)
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.T.)
	Contains(String) [▶ 2558]	Determines whether this collection contains an instance with the specified instance path. (Inherited from IInstanceCollection.T. [▶ 2554].)

	Name	Description
	ContainsName [▶ 2558]	Determines whether this collection contains an instance with the specified instance name. (Inherited from IInstanceCollection.T. [▶ 2554].)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.T.)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.T.)
	GetInstance [▶ 2559]	Gets the IInstance [▶ 2549] by instance path. (Inherited from IInstanceCollection.T. [▶ 2554].)
	GetInstanceByName [▶ 2559]	Gets the IInstance [▶ 2549] by instance name. (Inherited from IInstanceCollection.T. [▶ 2554].)
	IndexOf	Determines the index of a specific item in the IList.T. . (Inherited from IList.T.)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.T.)
	Remove	Removes the first occurrence of a specific object from the ICollection.T. . (Inherited from ICollection.T.)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.T.)
	TryGetInstance [▶ 2560]	Tries to get the specified instance. (Inherited from IInstanceCollection.T. [▶ 2554].)
	TryGetInstanceByName [▶ 2560]	Tries to get the specified instance by name. (Inherited from IInstanceCollection.T. [▶ 2554].)

Reference






[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

[TwinCAT.TypeSystem.IInstanceCollection.T.](#) [▶ 2554]

6.11.88.1 ISymbolCollection.T. Properties

The [ISymbolCollection.T.](#) [▶ 2700] generic type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T. . (Inherited from ICollection.T. [▶ 2700].)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.T. [▶ 2700].)
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.I [▶ 2700].)
	Item.String. [▶ 2556]	Gets the IInstance [▶ 2549] with the specified instance path. (Inherited from IInstanceCollection.T. [▶ 2554].)
	Mode [▶ 2556]	Gets the InstanceCollectionMode [▶ 2573]. (Inherited from IInstanceCollection.T. [▶ 2554].)

Reference
















[ISymbolCollection.T. Interface](#) [▶ 2700]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.88.2 ISymbolCollection.T. Methods

The [ISymbolCollection.T.](#) [▶ 2700] generic type exposes the following members.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T. . (Inherited from ICollection.T. [▶ 2700].)
	Clear	Removes all items from the ICollection.T. . (Inherited from ICollection.T. [▶ 2700].)
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.T. [▶ 2700].)
	Contains(String) [▶ 2558]	Determines whether this collection contains an instance with the specified instance path. (Inherited from IInstanceCollection.T. [▶ 2554].)
	ContainsName [▶ 2558]	Determines whether this collection contains an instance with the specified instance name. (Inherited from IInstanceCollection.T. [▶ 2554].)
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.T. [▶ 2700].)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.T. [▶ 2700].)
	GetInstance [▶ 2559]	Gets the IInstance [▶ 2549] by instance path. (Inherited from IInstanceCollection.T. [▶ 2554].)
	GetInstanceByName [▶ 2559]	Gets the IInstance [▶ 2549] by instance name. (Inherited from IInstanceCollection.T. [▶ 2554].)
	IndexOf	Determines the index of a specific item in the IList.T. . (Inherited from IList.T. [▶ 2700].)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.T. [▶ 2700].)
	Remove	Removes the first occurrence of a specific object from the ICollection.T. . (Inherited from ICollection.T. [▶ 2700].)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.T. [▶ 2700].)
	TryGetInstance [▶ 2560]	Tries to get the specified instance. (Inherited from IInstanceCollection.T. [▶ 2554].)
	TryGetInstanceByName [▶ 2560]	Tries to get the specified instance by name. (Inherited from IInstanceCollection.T. [▶ 2554].)

Reference

[ISymbolCollection.T. Interface](#) [▶ 2700]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.89 ISymbolFactory Interface

Symbol Factory Interface

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229




Syntax

C#











```
public interface ISymbolFactory
```

The ISymbolFactory type exposes the following members.

Properties

	Name	Description
	FactoryServices [▶ 2704]	Gets the factory services.
	HasInvalidCharacters [▶ 2704]	Gets a value indicating whether ISymbol [▶ 2691]s have invalid characters
	InvalidCharacters [▶ 2705]	Gets the invalid characters that are not allowed to appear within the Instance Name

Methods

	Name	Description
	CreateArrayElement [▶ 2706]	Creates a single Array Element
	CreateArrayElementInstances [▶ 2706]	Creates all Element Instances of the specified array parent symbol.
	CreateFieldInstance [▶ 2707]	Creates a single Instance member on a struct parent
	CreateFieldInstances [▶ 2708]	Creates the Member Instances collection for the specified parent instance
	CreateInstance [▶ 2708]	Creates the Symbol with the specified resolver
	CreateInstanceAsync [▶ 2709]	Creates the Symbol with the specified resolver
	CreateReferenceInstance [▶ 2709]	Creates the dereferenced Pointer instance
	CreateVirtualStruct [▶ 2710]	Creates the virtual structure.
	Initialize [▶ 2710]	Initializes the the ISymbolFactory.
	SetInvalidCharacters [▶ 2711]	Sets the invalid characters.




Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.89.1 ISymbolFactory Properties

The [ISymbolFactory](#) [▶ 2702] type exposes the following members.

Properties

	Name	Description
	FactoryServices [▶ 2704]	Gets the factory services.
	HasInvalidCharacters [▶ 2704]	Gets a value indicating whether ISymbol [▶ 2691]s have invalid characters
	InvalidCharacters [▶ 2705]	Gets the invalid characters that are not allowed to appear within the Instance Name

Reference

[ISymbolFactory Interface](#) [▶ 2702]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.89.1.1 ISymbolFactory.FactoryServices Property

Gets the factory services.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
ISymbolFactoryServices? FactoryServices { get; }
```

Property Value

Type: [ISymbolFactoryServices](#)
The factory services.

Reference

[ISymbolFactory Interface](#) [▶ 2702]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.89.1.2 ISymbolFactory.HasInvalidCharacters Property

Gets a value indicating whether [ISymbol](#) [▶ 2691]s have invalid characters

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
bool HasInvalidCharacters { get; }
```

Property Value

Type: Boolean
true if this instance has invalid characters; otherwise, false.

Reference

[ISymbolFactory Interface \[▶ 2702\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

[ISymbolFactory.InvalidCharacters \[▶ 2705\]](#)

6.11.89.1.3 ISymbolFactory.InvalidCharacters Property

Gets the invalid characters that are not allowed to appear within the Instance Name

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
char[] InvalidCharacters { get; }
```

Property Value

Type: .Char.
The forbidden characters.

Reference

[ISymbolFactory Interface \[▶ 2702\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)







[ISymbolFactory.SetInvalidCharacters\(.Char.\) \[▶ 2711\]](#)





[ISymbolFactory.HasInvalidCharacters \[▶ 2704\]](#)

6.11.89.2 ISymbolFactory Methods

The [ISymbolFactory \[▶ 2702\]](#) type exposes the following members.

Methods

	Name	Description
	CreateArrayElement [▶ 2706]	Creates a single Array Element
	CreateArrayElement Instances [▶ 2706]	Creates all Element Instances of the specified array parent symbol.
	CreateFieldInstance [▶ 2707]	Creates a single Instance member on a struct parent
	CreateFieldInstances [▶ 2708]	Creates the Member Instances collection for the specified parent instance
	CreateInstance [▶ 2708]	Creates the Symbol with the specified resolver
	CreateInstanceAsyn c [▶ 2709]	Creates the Symbol with the specified resolver

	Name	Description
	CreateReferenceInstance [▶ 2709]	Creates the dereferenced Pointer instance
	CreateVirtualStruct [▶ 2710]	Creates the virtual structure.
	Initialize [▶ 2710]	Initializes the the ISymbolFactory [▶ 2702].
	SetInvalidCharacters [▶ 2711]	Sets the invalid characters.

Reference

[ISymbolFactory Interface](#) [[▶ 2702](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.89.2.1 ISymbolFactory.CreateArrayElement Method

Creates a single Array Element

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISymbol CreateArrayElement(
    IArrayType arrayType,
    int[] currentIndex,
    ISymbol parent
)
```

Parameters

arrayType	Type: TwinCAT.TypeSystem.IArrayType [▶ 2459] Resolved Array type.
currentIndex	Type: <code>.System.Int32</code> . Array Index of the Element
parent	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] Array Instance

Return Value

Type: [ISymbol](#) [[▶ 2691](#)]

Array element

Reference

[ISymbolFactory Interface](#) [[▶ 2702](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.89.2.2 ISymbolFactory.CreateArrayElementInstances Method

Creates all Element Instances of the specified array parent symbol.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISymbolCollection<ISymbol> CreateArrayElementInstances (
    ISymbol parentInstance,
    IArrayType arrayType
)
```

Parameters

parentInstance	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The parent instance.
arrayType	Type: TwinCAT.TypeSystem.IArrayType [▶ 2459] Resolved array type.

Return Value

Type: [ISymbolCollection](#) [▶ 2700].[ISymbol](#) [▶ 2691].
SymbolCollection.

Reference

[ISymbolFactory Interface](#) [▶ 2702]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.89.2.3 ISymbolFactory.CreateFieldInstance Method

Creates a single Instance member on a struct parent

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISymbol CreateFieldInstance (
    IField field,
    ISymbol parent
)
```

Parameters

field	Type: TwinCAT.TypeSystem.IField [▶ 2535] Field
parent	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] Parent Struct/Alias/Union

Return Value

Type: [ISymbol](#) [▶ 2691]
Instance member

Remarks

Because the Alias type can act like a struct, the parent can be an [IAliasInstance](#) also.

Reference

[ISymbolFactory Interface](#) [▶ 2702]

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.89.2.4 ISymbolFactory.CreateFieldInstances Method

Creates the Member Instances collection for the specified parent instance

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISymbolCollection<ISymbol> CreateFieldInstances(
    ISymbol parentInstance,
    IDataType parentType
)
```

Parameters

parentInstance	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The parent instance.
parentType	Type: TwinCAT.TypeSystem.IDataType [► 2475] Parent Type (Struct/Alias/Union).

Return Value

Type: [ISymbolCollection \[► 2700\].ISymbol \[► 2691\]](#).
SymbolCollection.

Reference

[ISymbolFactory Interface \[► 2702\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.89.2.5 ISymbolFactory.CreateInstance Method

Creates the Symbol with the specified resolver

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISymbol CreateInstance(
    ISymbolInfo entry,
    ISymbol? parent
)
```

Parameters

entry	Type: TwinCAT.TypeSystem.ISymbolInfo [► 2712] Symbol Entry.
parent	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The parent.

Return Value

Type: [ISymbol](#) [[▶ 2691](#)]
Symbol instance.

Reference

[ISymbolFactory Interface](#) [[▶ 2702](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.89.2.6 ISymbolFactory.CreateInstanceAsync Method

Creates the Symbol with the specified resolver

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
Task<ResultValue<ISymbol>> CreateInstanceAsync (
    ISymbolInfo entry,
    ISymbol? parent,
    CancellationToken cancel
)
```

Parameters

entry	Type: TwinCAT.TypeSystem.ISymbolInfo [▶ 2712] Symbol Entry.
parent	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The parent.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultValue](#) [[▶ 1181](#)].[ISymbol](#) [[▶ 2691](#)].
[Task<ResultValue<ISymbol>>](#).

Reference

[ISymbolFactory Interface](#) [[▶ 2702](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.89.2.7 ISymbolFactory.CreateReferenceInstance Method

Creates the dereferenced Pointer instance

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
ISymbol? CreateReferenceInstance (
    IPointerType type,
    ISymbol parent
)
```

Parameters

type	Type: TwinCAT.TypeSystem.IPointerType [▶ 2584] Reference/Pointer type.
parent	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] Parent Instance of the reference

Return Value

Type: [ISymbol](#) [▶ 2691]
Reference/Pointer instance.

Reference

[ISymbolFactory Interface](#) [▶ 2702]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.89.2.8 ISymbolFactory.CreateVirtualStruct Method

Creates the virtual structure.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
ISymbol CreateVirtualStruct(
    string instanceName,
    string instancePath,
    ISymbol? parent
)
```

Parameters

instanceName	Type: System.String Name of the instance.
instancePath	Type: System.String The instance path.
parent	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The parent.

Return Value

Type: [ISymbol](#) [▶ 2691]
Virtual struct instance

Reference

[ISymbolFactory Interface](#) [▶ 2702]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.89.2.9 ISymbolFactory.Initialize Method

Initializes the the [ISymbolFactory](#) [▶ 2702].

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void Initialize(  
    ISymbolFactoryServices services  
)
```

Parameters

services	Type: ISymbolFactoryServices The services.
----------	---

Reference

[ISymbolFactory Interface](#) [► 2702]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.89.2.10 ISymbolFactory.SetInvalidCharacters Method

Sets the invalid characters.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void SetInvalidCharacters(  
    char[] invalidChars  
)
```

Parameters

invalidChars	Type: .System.Char. The character.
--------------	---------------------------------------

Reference

[ISymbolFactory Interface](#) [► 2702]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

[ISymbolFactory.InvalidCharacters](#) [► 2705]

6.11.90 ISymbolFactoryServicesProvider Interface

Symbol Value Access interface

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#

```
public interface ISymbolFactoryServicesProvider
```

The ISymbolFactoryServicesProvider type exposes the following members.

Properties

	Name	Description
	FactoryServices [▶ 2712]	Gets the factory services.


Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.90.1 ISymbolFactoryServicesProvider Properties

The [ISymbolFactoryServicesProvider](#) [▶ 2711] type exposes the following members.

Properties

	Name	Description
	FactoryServices [▶ 2712]	Gets the factory services.

Reference

[ISymbolFactoryServicesProvider Interface](#) [▶ 2711]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.90.1.1 ISymbolFactoryServicesProvider.FactoryServices Property

Gets the factory services.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
ISymbolFactoryServices FactoryServices { get; }
```

Property Value

Type: ISymbolFactoryServices

The factory services.

Reference

[ISymbolFactoryServicesProvider Interface](#) [▶ 2711]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.91 ISymbolInfo Interface

Interface ISymbolInfo

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax

C#

```
public interface ISymbolInfo
```

The ISymbolInfo type exposes the following members.

Properties

	Name	Description
	Name [▶ 2714]	Gets the Symbol Path
	TypeName [▶ 2714]	Gets the data type Name



Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.91.1 ISymbolInfo Properties

The [ISymbolInfo](#) [[▶ 2712](#)] type exposes the following members.

Properties

	Name	Description
	Name [▶ 2714]	Gets the Symbol Path
	TypeName [▶ 2714]	Gets the data type Name

Reference

[ISymbolInfo Interface](#) [[▶ 2712](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.91.1.1 ISymbolInfo.InstancePath Property

Gets the Symbol Path

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934 addedca3e72bc0ea15da1c14

Syntax

C#

```
string InstancePath { get; }
```

Property Value

Type: [String](#)
The path.

Reference

[ISymbolInfo Interface](#) [[▶ 2712](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.91.1.2 **ISymbolInfo.TypeName Property**

Gets the data type Name

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string TypeName { get; }
```

Property Value

Type: String

The type of the data.

Reference

[ISymbolInfo Interface](#) [[▶ 2712](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.91.1.3 **ISymbolInfo.Name Property**

Gets the Symbol Path

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string Name { get; }
```

Property Value

Type: String

The path.

Reference

[ISymbolInfo Interface](#) [[▶ 2712](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.92 **ISymbolLoader Interface**

Symbol Loader interface

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229







Syntax

C#






```
public interface ISymbolLoader : ISymbolProvider,  
    ISymbolServer
```

The ISymbolLoader type exposes the following members.

Properties

	Name	Description
	BuildInTypes [▶ 2716]	Gets the build in types.
	DataTypes [▶ 2720]	Gets the data types (Inherited from ISymbolServer [▶ 2719].)
	DefaultValueEncoding [▶ 2721]	Gets the default value encoding. (Inherited from ISymbolServer [▶ 2719].)
	RootNamespaceName [▶ 2718]	Gets the name of the root namespace (Inherited from ISymbolProvider [▶ 2717].)
	Settings [▶ 2716]	Gets or sets the access Method
	Symbols [▶ 2721]	Gets the symbols. (Inherited from ISymbolServer [▶ 2719].)

Methods

	Name	Description
	GetDataTypesAsync [▶ 2722]	Gets the data types asynchronously. (Inherited from ISymbolServer [▶ 2719].)
	GetSymbolsAsync [▶ 2723]	Gets the symbols asynchronously (Inherited from ISymbolServer [▶ 2719].)
	ResetCachedSymbolicData [▶ 2724]	Resets the cached symbolic data. (Inherited from ISymbolServer [▶ 2719].)
	TryGetDataTypes [▶ 2723]	Tries to get the symbols from the device target. (Inherited from ISymbolServer [▶ 2719].)
	TryGetSymbols [▶ 2724]	Tries to geth the symbols from the device target. (Inherited from ISymbolServer [▶ 2719].)





Reference



[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.92.1 ISymbolLoader Properties

The [ISymbolLoader](#) [[▶ 2714](#)] type exposes the following members.

Properties

	Name	Description
	BuildInTypes [▶ 2716]	Gets the build in types.
	DataTypes [▶ 2720]	Gets the data types (Inherited from ISymbolServer [▶ 2719].)
	DefaultValueEncoding [▶ 2721]	Gets the default value encoding. (Inherited from ISymbolServer [▶ 2719].)
	RootNamespaceName [▶ 2718]	Gets the name of the root namespace (Inherited from ISymbolProvider [▶ 2717].)

	Name	Description
	Settings [▶ 2716]	Gets or sets the access Method
	Symbols [▶ 2721]	Gets the symbols. (Inherited from ISymbolServer [▶ 2719].)

Reference

[ISymbolLoader Interface](#) [[▶ 2714](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.92.1.1 ISymbolLoader.BuildInTypes Property

Gets the build in types.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDataValueCollection BuildInTypes { get; }
```

Property Value

Type: [IDataValueCollection](#) [[▶ 2484](#)]

The build in types.

Reference

[ISymbolLoader Interface](#) [[▶ 2714](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.92.1.2 ISymbolLoader.Settings Property

Gets or sets the access Method

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISymbolLoaderSettings Settings { get; }
```

Property Value

Type: [ISymbolLoaderSettings](#) [[▶ 104](#)]

The access method.

Reference






[ISymbolLoader Interface](#) [[▶ 2714](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.92.2 ISymbolLoader Methods

The [ISymbolLoader](#) [▶ 2714] type exposes the following members.

Methods

	Name	Description
	GetDataTypesAsync [▶ 2722]	Gets the data types asynchronously. (Inherited from ISymbolServer [▶ 2719].)
	GetSymbolsAsync [▶ 2723]	Gets the symbols asynchronously (Inherited from ISymbolServer [▶ 2719].)
	ResetCachedSymbolicData [▶ 2724]	Resets the cached symbolic data. (Inherited from ISymbolServer [▶ 2719].)
	TryGetDataTypes [▶ 2723]	Tries to get the symbols from the device target. (Inherited from ISymbolServer [▶ 2719].)
	TryGetSymbols [▶ 2724]	Tries to geth the symbols from the device target. (Inherited from ISymbolServer [▶ 2719].)

Reference

[ISymbolLoader Interface](#) [▶ 2714]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.93 ISymbolProvider Interface

Symbol Provider interface.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229





Syntax

C#






```
public interface ISymbolProvider : ISymbolServer
```

The [ISymbolProvider](#) type exposes the following members.

Properties

	Name	Description
	DataTypes [▶ 2720]	Gets the data types (Inherited from ISymbolServer [▶ 2719].)
	DefaultValueEncoding [▶ 2721]	Gets the default value encoding. (Inherited from ISymbolServer [▶ 2719].)
	RootNamespaceName [▶ 2718]	Gets the name of the root namespace
	Symbols [▶ 2721]	Gets the symbols. (Inherited from ISymbolServer [▶ 2719].)

Methods

	Name	Description
	GetDataTypesAsync [▶ 2722]	Gets the data types asynchronously. (Inherited from ISymbolServer [▶ 2719].)
	GetSymbolsAsync [▶ 2723]	Gets the symbols asynchronously (Inherited from ISymbolServer [▶ 2719].)
	ResetCachedSymbolicData [▶ 2724]	Resets the cached symbolic data. (Inherited from ISymbolServer [▶ 2719].)
	TryGetDataTypes [▶ 2723]	Tries to get the symbols from the device target. (Inherited from ISymbolServer [▶ 2719].)
	TryGetSymbols [▶ 2724]	Tries to geth the symbols from the device target. (Inherited from ISymbolServer [▶ 2719].)





Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.93.1 ISymbolProvider Properties

The [ISymbolProvider](#) [▶ 2717] type exposes the following members.

Properties

	Name	Description
	DataTypes [▶ 2720]	Gets the data types (Inherited from ISymbolServer [▶ 2719].)
	DefaultValueEncoding [▶ 2721]	Gets the default value encoding. (Inherited from ISymbolServer [▶ 2719].)
	RootNamespaceName [▶ 2718]	Gets the name of the root namespace
	Symbols [▶ 2721]	Gets the symbols. (Inherited from ISymbolServer [▶ 2719].)

Reference

[ISymbolProvider Interface](#) [▶ 2717]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.93.1.1 ISymbolProvider.RootNamespaceName Property

Gets the name of the root namespace

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
string RootNamespaceName { get; }
```

Property Value

Type: String
The namespace.

Reference






[ISymbolProvider Interface \[▸ 2717\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.93.2 ISymbolProvider Methods

The [ISymbolProvider \[▸ 2717\]](#) type exposes the following members.

Methods

	Name	Description
	GetDataTypesAsync [▸ 2722]	Gets the data types asynchronously. (Inherited from ISymbolServer [▸ 2719].)
	GetSymbolsAsync [▸ 2723]	Gets the symbols asynchronously (Inherited from ISymbolServer [▸ 2719].)
	ResetCachedSymbolicData [▸ 2724]	Resets the cached symbolic data. (Inherited from ISymbolServer [▸ 2719].)
	TryGetDataTypes [▸ 2723]	Tries to get the symbols from the device target. (Inherited from ISymbolServer [▸ 2719].)
	TryGetSymbols [▸ 2724]	Tries to geth the symbols from the device target. (Inherited from ISymbolServer [▸ 2719].)

Reference

[ISymbolProvider Interface \[▸ 2717\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.94 ISymbolServer Interface

Symbol Server Interface

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax



C#

```
public interface ISymbolServer
```






The [ISymbolServer](#) type exposes the following members.

Properties

	Name	Description
	DataTypes [▸ 2720]	Gets the data types

	Name	Description
	DefaultValueEncoding [▸ 2721]	Gets the default value encoding.
	Symbols [▸ 2721]	Gets the symbols.

Methods

	Name	Description
	GetDataTypesAsync [▸ 2722]	Gets the data types asynchronously.
	GetSymbolsAsync [▸ 2723]	Gets the symbols asynchronously
	ResetCachedSymbolicData [▸ 2724]	Resets the cached symbolic data.
	TryGetDataTypes [▸ 2723]	Tries to get the symbols from the device target.
	TryGetSymbols [▸ 2724]	Tries to geth the symbols from the device target.




Reference

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.94.1 ISymbolServer Properties

The [ISymbolServer](#) [[▸ 2719](#)] type exposes the following members.

Properties

	Name	Description
	DataTypes [▸ 2720]	Gets the data types
	DefaultValueEncoding [▸ 2721]	Gets the default value encoding.
	Symbols [▸ 2721]	Gets the symbols.

Reference

[ISymbolServer Interface](#) [[▸ 2719](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.94.1.1 ISymbolServer.DataTypes Property

Gets the data types

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
IDataValueCollection<IDataType> DataTypes { get; }
```

Property Value

Type: [IDataValueCollection](#) [[▶ 2486](#)].[IDataType](#) [[▶ 2475](#)].

The data types.

Remarks

This property reads the `DataTypes` synchronously, if the data is not available yet. For performance reasons, the asynchronous counterpart [GetDataTypesAsync\(CancellationToken\)](#) [[▶ 2722](#)] should be preferred for the first call.

Reference

[ISymbolServer Interface](#) [[▶ 2719](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

[ISymbolServer.GetDataTypesAsync\(CancellationToken\)](#) [[▶ 2722](#)]

6.11.94.1.2 ISymbolServer.DefaultValueEncoding Property

Gets the default value encoding.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Encoding DefaultValueEncoding { get; }
```

Property Value

Type: `Encoding`

The default value encoding.

Reference

[ISymbolServer Interface](#) [[▶ 2719](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.94.1.3 ISymbolServer.Symbols Property

Gets the symbols.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISymbolCollection<ISymbol> Symbols { get; }
```

Property Value

Type: [ISymbolCollection](#) [[▸ 2700](#)].[ISymbol](#) [[▸ 2691](#)].
The symbols.

Remarks

This property reads the Symbol information synchronously, if the data is not available yet. For performance reasons, the asynchronous counterpart [GetSymbolsAsync\(CancellationTokentoken\)](#) [[▸ 2723](#)] should be preferred for the first call.

Reference

[ISymbolServer Interface](#) [[▸ 2719](#)]






[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

[ISymbolServer.GetSymbolsAsync\(CancellationTokentoken\)](#) [[▸ 2723](#)]

6.11.94.2 ISymbolServer Methods

The [ISymbolServer](#) [[▸ 2719](#)] type exposes the following members.

Methods

	Name	Description
	GetDataTypesAsync [▸ 2722]	Gets the data types asynchronously.
	GetSymbolsAsync [▸ 2723]	Gets the symbols asynchronously
	ResetCachedSymbolicData [▸ 2724]	Resets the cached symbolic data.
	TryGetDataTypes [▸ 2723]	Tries to get the symbols from the device target.
	TryGetSymbols [▸ 2724]	Tries to geth the symbols from the device target.

Reference

[ISymbolServer Interface](#) [[▸ 2719](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.94.2.1 ISymbolServer.GetDataTypesAsync Method

Gets the data types asynchronously.

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultDataTypes> GetDataTypesAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultDataTypes](#) [[▸ 2872](#)].

A task that represents the asynchronous 'GetDataTypes' operation. The [ResultDataTypes](#) [[▸ 2872](#)] parameter contains the data types ([DataTypes](#) [[▸ 2874](#)]) and the [ErrorCode](#) [[▸ 1120](#)] after execution.

Reference

[ISymbolServer Interface](#) [[▸ 2719](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

[ISymbolServer.DataTypes](#) [[▸ 2720](#)]

6.11.94.2.2 ISymbolServer.GetSymbolsAsync Method

Gets the symbols asynchronously

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultSymbols> GetSymbolsAsync(  
    CancellationToken cancel  
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultSymbols](#) [[▸ 2880](#)].

A task that represents the asynchronous 'GetDataTypes' operation. The [ResultSymbols](#) [[▸ 2880](#)] parameter contains the data types ([Symbols](#) [[▸ 2886](#)]) and the [ErrorCode](#) [[▸ 1120](#)] after execution.

Reference

[ISymbolServer Interface](#) [[▸ 2719](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

[ISymbolServer.Symbols](#) [[▸ 2721](#)]

6.11.94.2.3 ISymbolServer.TryGetDataTypes Method

Tries to get the symbols from the device target.

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryGetDataTypes(
    out IDataValueCollection<IDataType>? dataTypes
)
```

Parameters

dataTypes	Type: TwinCAT.TypeSystem.IDataValueCollection [2486]. IDataType [2475]. The data types.
-----------	--

Return Value

Type: [AdsErrorCode](#) [[664](#)]
AdsErrorCode.

Reference

[ISymbolServer Interface](#) [[2719](#)]

[TwinCAT.TypeSystem Namespace](#) [[2083](#)]

6.11.94.2.4 ISymbolServer.TryGetSymbols Method

Tries to geth the symbols from the device target.

Namespace: [TwinCAT.TypeSystem](#) [[2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
AdsErrorCode TryGetSymbols(
    out ISymbolCollection<ISymbol>? symbols
)
```

Parameters

symbols	Type: TwinCAT.TypeSystem.ISymbolCollection [2700]. ISymbol [2691]. The symbols.
---------	--

Return Value

Type: [AdsErrorCode](#) [[664](#)]
AdsErrorCode.

Reference

[ISymbolServer Interface](#) [[2719](#)]

[TwinCAT.TypeSystem Namespace](#) [[2083](#)]

6.11.94.2.5 ISymbolServer.ResetCachedSymbolicData Method

Resets the cached symbolic data.

Namespace: [TwinCAT.TypeSystem](#) [[2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void ResetCachedSymbolicData()
```

Reference

[ISymbolServer Interface \[▸ 2719\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.95 ITypeAttribute Interface

Interface for ADS attributes

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax

C#

```
public interface ITypeAttribute
```

The ITypeAttribute type exposes the following members.

Properties

	Name	Description
	Name [▸ 2725]	Name of the Attribute
	Value [▸ 2726]	Gets the value of the attribute



Reference

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.95.1 ITypeAttribute Properties

The [ITypeAttribute \[▸ 2725\]](#) type exposes the following members.

Properties

	Name	Description
	Name [▸ 2725]	Name of the Attribute
	Value [▸ 2726]	Gets the value of the attribute

Reference

[ITypeAttribute Interface \[▸ 2725\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.95.1.1 ITypeAttribute.Name Property

Name of the Attribute

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string Name { get; }
```

Property Value

Type: String
The name.

Reference

[ITypeAttribute Interface](#) [▸ 2725]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.95.1.2 ITypeAttribute.Value Property

Gets the value of the attribute

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string Value { get; }
```

Property Value

Type: String
The value.

Reference

[ITypeAttribute Interface](#) [▸ 2725]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.96 ITypeAttributeCollection Interface

Interface ITypeAttributeCollection

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229





Syntax

C#













```
public interface ITypeAttributeCollection : IList<ITypeAttribute>,  
    ICollection<ITypeAttribute>, IEnumerable<ITypeAttribute>, IEnumerable
```

The ITypeAttributeCollection type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.ITypeAttribute [▶ 2725]..)
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.ITypeAttribute [▶ 2725]..)
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.ITypeAttribute [▶ 2725]..)
	Item.String. [▶ 2728]	Gets the String with the specified name.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.ITypeAttribute [▶ 2725]..)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.ITypeAttribute [▶ 2725]..)
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.ITypeAttribute [▶ 2725]..)
	Contains(String) [▶ 2730]	Determines whether this ITypeAttributeCollection contains the specified attribute.
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.ITypeAttribute [▶ 2725]..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.ITypeAttribute [▶ 2725]..)
	IndexOf	Determines the index of a specific item in the IList.T.. (Inherited from IList.ITypeAttribute [▶ 2725]..)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.ITypeAttribute [▶ 2725]..)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.ITypeAttribute [▶ 2725]..)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.ITypeAttribute [▶ 2725]..)
	TryGetAttribute [▶ 2730]	Tries to get the specified ITypeAttribute [▶ 2725]
	TryGetValue [▶ 2731]	Tries to get the specified Attribute value.


Reference




[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.96.1 ITypeAttributeCollection Properties

The [ITypeAttributeCollection \[▶ 2726\]](#) type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ICollection.T.. (Inherited from ICollection.ITypeAttribute [▶ 2725]..)

	Name	Description
	IsReadOnly	Gets a value indicating whether the ICollection.T. is read-only. (Inherited from ICollection.ITypeAttribute [▶ 2725].)
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.ITypeAttribute [▶ 2725].)
	Item.String. [▶ 2728]	Gets the String with the specified name.



Reference

[ITypeAttributeCollection Interface \[\[▶ 2726\]\(#\)\]](#)

[TwinCAT.TypeSystem Namespace \[\[▶ 2083\]\(#\)\]](#)

6.11.96.1.1 ITypeAttributeCollection.Item Property

Overload List

	Name	Description
	Item.Int32.	Gets or sets the element at the specified index. (Inherited from IList.ITypeAttribute [▶ 2725].)
	Item.String. [▶ 2728]	Gets the String with the specified name.

Reference

[ITypeAttributeCollection Interface \[\[▶ 2726\]\(#\)\]](#)

[TwinCAT.TypeSystem Namespace \[\[▶ 2083\]\(#\)\]](#)

6.11.96.1.1.1 ITypeAttributeCollection.Item Property (String)

Gets the String with the specified name.

Namespace: [TwinCAT.TypeSystem \[\[▶ 2083\]\(#\)\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string this[
    string name
] { get; }
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: String
System.String.

Reference

[ITypeAttributeCollection Interface \[\[▶ 2726\]\(#\)\]](#)













[Item Overload \[\[▶ 2728\]\(#\)\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.96.2 ITypeAttributeCollection Methods

The [ITypeAttributeCollection \[▶ 2726\]](#) type exposes the following members.

Methods

	Name	Description
	Add	Adds an item to the ICollection.T.. (Inherited from ICollection.ITypeAttribute [▶ 2725]..)
	Clear	Removes all items from the ICollection.T.. (Inherited from ICollection.ITypeAttribute [▶ 2725]..)
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.ITypeAttribute [▶ 2725]..)
	Contains(String) [▶ 2730]	Determines whether this ITypeAttributeCollection [▶ 2726] contains the specified attribute.
	CopyTo	Copies the elements of the ICollection.T. to an Array, starting at a particular Array index. (Inherited from ICollection.ITypeAttribute [▶ 2725]..)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable.ITypeAttribute [▶ 2725]..)
	IndexOf	Determines the index of a specific item in the IList.T.. (Inherited from IList.ITypeAttribute [▶ 2725]..)
	Insert	Inserts an item to the IList.T. at the specified index. (Inherited from IList.ITypeAttribute [▶ 2725]..)
	Remove	Removes the first occurrence of a specific object from the ICollection.T.. (Inherited from ICollection.ITypeAttribute [▶ 2725]..)
	RemoveAt	Removes the IList.T. item at the specified index. (Inherited from IList.ITypeAttribute [▶ 2725]..)
	TryGetAttribute [▶ 2730]	Tries to get the specified ITypeAttribute [▶ 2725]
	TryGetValue [▶ 2731]	Tries to get the specified Attribute value.



Reference

[ITypeAttributeCollection Interface \[▶ 2726\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.96.2.1 ITypeAttributeCollection.Contains Method

Overload List

	Name	Description
	Contains(T)	Determines whether the ICollection.T. contains a specific value. (Inherited from ICollection.ITypeAttribute [▶ 2725]..)
	Contains(String) [▶ 2730]	Determines whether this ITypeAttributeCollection [▶ 2726] contains the specified attribute.

Reference

[ITypeAttributeCollection Interface \[▶ 2726\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.96.2.1.1 ITypeAttributeCollection.Contains Method (String)

Determines whether this [ITypeAttributeCollection](#) [▸ 2726] contains the specified attribute.

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool Contains(  
    string name  
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: Boolean

true if [contains] [the specified name]; otherwise, false.

Reference

[ITypeAttributeCollection Interface](#) [▸ 2726]

[Contains Overload](#) [▸ 2729]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.96.2.2 ITypeAttributeCollection.TryGetAttribute Method

Tries to get the specified [ITypeAttribute](#) [▸ 2725]

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool TryGetAttribute(  
    string name,  
    out ITypeAttribute?? attribute  
)
```

Parameters

name	Type: System.String The name of the ITypeAttribute [▸ 2725].
attribute	Type: TwinCAT.TypeSystem.ITypeAttribute [▸ 2725]. The attribute.

Return Value

Type: Boolean

true if found, false otherwise.

Reference

[ITypeAttributeCollection Interface \[▶ 2726\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.96.2.3 ITypeAttributeCollection.TryGetValue Method

Tries to get the specified Attribute value.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool TryGetValue(
    string name,
    out string?? value
)
```

Parameters

name	Type: System.String The name.
value	Type: System.String. The value.

Return Value

Type: Boolean
true if the value is found, false otherwise.

Reference

[ITypeAttributeCollection Interface \[▶ 2726\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.97 ITypeMarshaler Interface

Interface ITypeMarshaler

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax




C#

```
public interface ITypeMarshaler
```

The ITypeMarshaler type exposes the following members.

Methods

	Name	Description
	CanMarshal [▶ 2732]	Determines whether ADS can marshal the specified managed data type.
	CanMarshalValue [▶ 2736]	Determines whether ADS can marshal the specified value

	Name	Description
	Marshal [▶ 2734]	Marshals the specified value to the specified destination memory / span.
	MarshalValueSize [▶ 2736]	Gets the byte size of the value when marshalled.
	Unmarshal [▶ 2735]	Unmarshals the specified managed type from memory / span




Remarks

The `ITypeMarshaler` is the common base interface for marshalling classes. It support marshalling / unmarshalling of primitive managed values.

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]






Also see about this

-  [ITypeMarshaler.CanMarshal Method \(Object\)](#) [[▶ 2733](#)]
-  [ITypeMarshaler.CanMarshal Method \(Type\)](#) [[▶ 2733](#)]
-  [ITypeMarshaler.MarshalSize Method](#) [[▶ 2735](#)]

6.11.97.1 ITypeMarshaler Methods

The `ITypeMarshaler` [[▶ 2731](#)] type exposes the following members.

Methods




	Name	Description
	CanMarshal [▶ 2732]	Determines whether ADS can marshal the specified managed data type.
	CanMarshalValue [▶ 2736]	Determines whether ADS can marshal the specified value
	Marshal [▶ 2734]	Marshals the specified value to the specified destination memory / span.
	MarshalValueSize [▶ 2736]	Gets the byte size of the value when marshalled.
	Unmarshal [▶ 2735]	Unmarshals the specified managed type from memory / span

Reference

[ITypeMarshaler Interface](#) [[▶ 2731](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

Also see about this

-  [ITypeMarshaler.CanMarshal Method \(Object\)](#) [[▶ 2733](#)]
-  [ITypeMarshaler.CanMarshal Method \(Type\)](#) [[▶ 2733](#)]
-  [ITypeMarshaler.MarshalSize Method](#) [[▶ 2735](#)]

6.11.97.1.1 ITypeMarshaler.CanMarshal Method

Determines whether ADS can marshal the specified managed data type.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
bool CanMarshal(  
    Type type  
)
```

Parameters

type	Type: System.Type The Managed data type.
------	---

Return Value

Type: Boolean

true if this instance can marshal the specified managed type; otherwise, false.

Reference

[ITypeMarshaler Interface](#) [► 2731]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.97.1.1.1 ITypeMarshaler.CanMarshal Method (Object)

Determines whether ADS can marshal the specified value

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
bool CanMarshal(  
    Object value  
)
```

Parameters

value	Type: System.Object The value.
-------	---

Return Value

Type: [Boolean](#)

true if this instance can marshal the specified value; otherwise, false.

Reference

[ITypeMarshaler Interface](#) [► 2731]

[CanMarshal Overload](#) [► 2732]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.97.1.1.2 ITypeMarshaler.CanMarshal Method (Type)

Determines whether ADS can marshal the specified managed data type.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
bool CanMarshal(
    Type type
)
```

Parameters

type Type: [System.Type](#)
The Managed data type.

Return Value

Type: [Boolean](#)
true if this instance can marshal the specified managed type; otherwise, false.

Reference

[ITypeMarshaler Interface](#) [► 2731]

[CanMarshal Overload](#) [► 2732]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.97.1.2 ITypeMarshaler.Marshal Method

Marshals the specified value to the specified destination memory / span.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Marshal(
    Object value,
    Encoding encoding,
    Span<byte> destination
)
```

Parameters

value	Type: System.Object The value to marshal.
encoding	Type: System.Text.Encoding The encoding.
destination	Type: System.Span.Byte . The destination span.

Return Value

Type: [Int32](#)
The number of marshalled bytes.

Reference

[ITypeMarshaler Interface](#) [► 2731]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.97.1.3 ITypeMarshaler.MarshalSize Method

Gets the byte size of the value when marshalled.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
int MarshalSize(
    Object value,
    Encoding encoding
)
```

Parameters

value	Type: System.Object The value.
encoding	Type: System.Text.Encoding The encoding.

Return Value

Type: [Int32](#)
The marshal size of the value.

Reference

[ITypeMarshaler Interface](#) [[▶ 2731](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.97.1.4 ITypeMarshaler.Unmarshal Method

Unmarshals the specified managed type from memory / span

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int Unmarshal(
    Type type,
    ReadOnlySpan<byte> source,
    Encoding encoding,
    out Object value
)
```

Parameters

type	Type: System.Type The type.
source	Type: System.ReadOnlySpan.Byte . The source memory/span
encoding	Type: System.Text.Encoding The encoding.
value	Type: System.Object . The created value.

Return Value

Type: Int32
Number of unmarshaled bytes.

Reference

[ITypeMarshaler Interface](#) [► 2731]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.97.1.5 ITypeMarshaler.CanMarshalValue Method

Determines whether ADS can marshal the specified value

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool CanMarshalValue(  
    Object value  
)
```

Parameters

value	Type: System.Object The value.
-------	-----------------------------------

Return Value

Type: Boolean
true if this instance can marshal the specified value; otherwise, false.

Reference

[ITypeMarshaler Interface](#) [► 2731]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.97.1.6 ITypeMarshaler.MarshalValueSize Method

Gets the byte size of the value when marshalled.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int MarshalValueSize(  
    Object value,  
    Encoding encoding  
)
```

Parameters

value	Type: System.Object The value.
-------	-----------------------------------

encoding	Type: System.Text.Encoding The encoding.
----------	---

Return Value

Type: Int32
The marshal size of the value.

Reference

[ITypeMarshaler Interface](#) [[▶ 2731](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.98 IUnionInstance Interface

Interface for an Instance of the [IUnionType](#) [[▶ 2740](#)].

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229












Syntax














C#

```
public interface IUnionInstance : ISymbol,
    IAttributedInstance, IInstance, IBitSize
```

The IUnionInstance type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	FieldInstances [▶ 2740]	Gets the field instances of the Union
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from IBitSize [▶ 2472].)

	Name	Description
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is a container type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference



[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

[TwinCAT.TypeSystem.ISymbol](#) [▶ 2691]

6.11.98.1 UnionInstance Properties

The [IUnionInstance](#) [▶ 2737] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)

	Name	Description
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the Instance [▶ 2549] (Inherited from Instance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549] . (Inherited from Instance [▶ 2549].)
	FieldInstances [▶ 2740]	Gets the field instances of the Union
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from Instance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from Instance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the Instance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from Instance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the Instance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from Instance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this Instance [▶ 2549] is static. (Inherited from Instance [▶ 2549].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549] . (Inherited from Instance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Reference

[IUnionInstance Interface \[▶ 2737\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.98.1 IUnionInstance.FieldInstances Property

Gets the field instances of the Union

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISymbolCollection<ISymbol> FieldInstances { get; }
```

Property Value

Type: [ISymbolCollection \[► 2700\]](#).[ISymbol \[► 2691\]](#).

The field instances.

Reference

[IUnionInstance Interface \[► 2737\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.98.2 IUnionInstance Methods

The [IUnionInstance \[► 2737\]](#) type exposes the following members.

Extension Methods

	Name	Description
	ReferencesExternalData [► 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [► 3021] .)

Reference

[IUnionInstance Interface \[► 2737\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.99 IUnionType Interface

Interface for an union data type.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229














Syntax

C#

```
public interface IUnionType : IDataTypes,
    IBitSize
```

The [IUnionType](#) type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataType [▶ 2475] (Inherited from IDataType [▶ 2475].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)
	Fields [▶ 2743]	Gets a readonly collection of the Members [▶ 2535] of the IUnionType .
	FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from IDataType [▶ 2475].)
	Id [▶ 2480]	Gets the ID of the Data Type (Inherited from IDataType [▶ 2475].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full Data Type but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475].)
	Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Extension Methods

	Name	Description
	IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
	IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)





	Name	Description
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataExtension [▶ 3021].)
	Resolve . [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataExtension [▶ 3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

[TwinCAT.TypeSystem.IDataType](#) [▶ 2475]






Also see about this









-  [IDataType.IsContainer Property](#) [▶ 2480]
-  [IDataType.IsPointer Property](#) [▶ 2481]
-  [IDataType.IsPrimitive Property](#) [▶ 2481]
-  [IDataType.IsReference Property](#) [▶ 2482]

6.11.99.1 UnionType Properties

The [IUnionType](#) [▶ 2740] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2478]	Gets the attributes of the IDataType [▶ 2475] (Inherited from IDataType [▶ 2475].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
	Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)





	Name	Description
	Fields [▸ 2743]	Gets a readonly collection of the Members [▸ 2535] of the IUnionType [▸ 2740] .
	FullName [▸ 2479]	Gets the full name of the IDataType [▸ 2475] (Namespace + Name) (Inherited from IDataType [▸ 2475] .)
	Id [▸ 2480]	Gets the ID of the Data Type (Inherited from IDataType [▸ 2475] .)
	IsBitType [▸ 2474]	Gets a value indicating whether this instance is not basing on a full Data Type but instead of some sort of bit mapping (Inherited from IBitSize [▸ 2472] .)
	IsByteAligned [▸ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▸ 2472] .)
	Name [▸ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▸ 2475] .)
	Namespace [▸ 2482]	Gets the namespace string within the IDataType [▸ 2475] exists. (Inherited from IDataType [▸ 2475] .)
	Size [▸ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▸ 2474] (Inherited from IBitSize [▸ 2472] .)

Reference

[IUnionType Interface \[▸ 2740\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

Also see about this

-  [IDataType.IsContainer Property \[▸ 2480\]](#)
-  [IDataType.IsPointer Property \[▸ 2481\]](#)
-  [IDataType.IsPrimitive Property \[▸ 2481\]](#)
-  [IDataType.IsReference Property \[▸ 2482\]](#)

6.11.99.1.1 IUnionType.Fields Property

Gets a readonly collection of the [Members \[▸ 2535\]](#) of the [IUnionType \[▸ 2740\]](#).

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IFieldCollection Fields { get; }
```

Property Value

Type: [IFieldCollection \[▸ 2538\]](#)

The members as readonly collection.

Remarks

If the [IStructType \[▸ 2671\]](#) is derived, only the extended members are returned. To get all supported members down the inheritance chain, use the [AllMembers \[▸ 2987\]](#) property.

Reference

[IUnionType Interface \[► 2740\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.99.2 IUnionType Methods

The [IUnionType \[► 2740\]](#) type exposes the following members.

Extension Methods

Name	Description
IsArrayOfPrimitives. [► 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [► 3021].)
IsArrayOfPrimitives(Boolean) [► 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [► 3021].)
IsContainer. [► 3027]	Overloaded. Gets a value indicating whether this IDataType [► 2475] is a container type (Defined by DataTypeExtension [► 3021].)
IsContainer(Boolean) [► 3028]	Overloaded. Gets a value indicating whether this IDataType [► 2475] is a container type (Defined by DataTypeExtension [► 3021].)
IsPointer [► 3031]	Gets a value indicating whether this IDataType [► 2475] is a pointer type (Defined by DataTypeExtension [► 3021].)
IsPrimitive. [► 3033]	Overloaded. Gets a value indicating whether this IDataType [► 2475] is primitive (Defined by DataTypeExtension [► 3021].)
IsPrimitive(Boolean) [► 3034]	Overloaded. Gets a value indicating whether this IDataType [► 2475] is primitive (Defined by DataTypeExtension [► 3021].)
IsReference [► 3036]	Gets a value indicating whether this IDataType [► 2475] is a reference type (Defined by DataTypeExtension [► 3021].)
ReferencesExternalD ata [► 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [► 3021].)
ResolvableAsPrimiti ve. [► 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [► 3021].)
ResolvableAsPrimiti ve(Boolean) [► 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [► 3021].)
Resolve. [► 3040]	Overloaded. Resolves the specified data type for AliasReference [► 3042] (Defined by DataTypeExtension [► 3021].)
Resolve(DataTypeRe solveStrategy) [► 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [► 3021].)

Reference

[IUnionType Interface \[▶ 2740\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.100 IValue Interface

Symbol Value Interface

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229








Syntax

C#







```
public interface IValue
```

The IValue type exposes the following members.

Properties

	Name	Description
	Age [▶ 2746]	Gets the age of the value (last successful read of the value)
	CachedRaw [▶ 2746]	Gets the cached Raw internal Data.
	DataType [▶ 2747]	Gets the data type bound to this IValue
	IsPrimitive [▶ 2747]	Gets a value indicating whether this IValue is a primitive value.
	Symbol [▶ 2748]	Gets the symbol bound to this IValue.
	TimeStamp [▶ 2748]	Gets the Time stamp of the last successful read of the Value (local user time, UTC)
	UpdateMode [▶ 2748]	Gets the update mode (not implemented yet)

Methods

	Name	Description
	Read [▶ 2749]	Reads the value (via ADS)
	ReadAsync [▶ 2750]	Reads the value (via ADS)
	ResolveValue [▶ 2750]	Resolves the Value object to its primitive value.
	TryResolveValue [▶ 2751]	Tries to resolves the Value object to its primitive value.
	Write [▶ 2751]	Writes the value (via ADS)
	WriteAsync [▶ 2751]	Writes the value (via ADS)








Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.100.1 IValue Properties

The [IValue](#) [▸ 2745] type exposes the following members.

Properties

	Name	Description
	Age [▸ 2746]	Gets the age of the value (last successful read of the value)
	CachedRaw [▸ 2746]	Gets the cached Raw internal Data.
	DataType [▸ 2747]	Gets the data type bound to this IValue [▸ 2745]
	IsPrimitive [▸ 2747]	Gets a value indicating whether this IValue [▸ 2745] is a primitive value.
	Symbol [▸ 2748]	Gets the symbol bound to this IValue [▸ 2745].
	TimeStamp [▸ 2748]	Gets the Time stamp of the last successful read of the Value (local user time, UTC)
	UpdateMode [▸ 2748]	Gets the update mode (not implemented yet)

Reference

[IValue Interface](#) [▸ 2745]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.100.1.1 IValue.Age Property

Gets the age of the value (last successful read of the value)

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
TimeSpan Age { get; }
```

Property Value

Type: TimeSpan

The age.

Reference

[IValue Interface](#) [▸ 2745]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

[IValue.TimeStamp](#) [▸ 2748]

6.11.100.1.2 IValue.CachedRaw Property

Gets the cached Raw internal Data.

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ReadOnlyMemory<byte> CachedRaw { get; }
```

Property Value

Type: `ReadOnlyMemory.Byte`.
The raw cached data.

Reference

[IValue Interface \[► 2745\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.100.1.3 IValue.DataType Property

Gets the data type bound to this [IValue \[► 2745\]](#).

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDataType? DataType { get; }
```

Property Value

Type: [IDataType \[► 2475\]](#)
The type of the data.

Reference

[IValue Interface \[► 2745\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.100.1.4 IValue.IsPrimitive Property

Gets a value indicating whether this [IValue \[► 2745\]](#) is a primitive value.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool IsPrimitive { get; }
```

Property Value

Type: `Boolean`
true if this instance is primitive; otherwise, false.

Reference

[IValue Interface \[► 2745\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.100.1.5 IValue.Symbol Property

Gets the symbol bound to this [IValue \[▸ 2745\]](#).

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISymbol Symbol { get; }
```

Property Value

Type: [ISymbol \[▸ 2691\]](#)

The symbol.

Reference

[IValue Interface \[▸ 2745\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.100.1.6 IValue.TimeStamp Property

Gets the Time stamp of the last successful read of the Value (local user time, UTC)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
DateTimeOffset TimeStamp { get; }
```

Property Value

Type: DateTimeOffset

The read time stamp.

Reference

[IValue Interface \[▸ 2745\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.100.1.7 IValue.UpdateMode Property

Gets the update mode (not implemented yet)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ValueUpdateMode UpdateMode { get; }
```

Property Value

Type: [ValueUpdateMode](#) [▶ 3229]

The update mode.

Reference

[IValue Interface](#) [▶ 2745]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]







Also see about this

 [ValueUpdateMode Enumeration](#) [▶ 159]

6.11.100.2 IValue Methods

The [IValue](#) [▶ 2745] type exposes the following members.

Methods

	Name	Description
	Read [▶ 2749]	Reads the value (via ADS)
	ReadAsync [▶ 2750]	Reads the value (via ADS)
	ResolveValue [▶ 2750]	Resolves the Value object to its primitive value.
	TryResolveValue [▶ 2751]	Tries to resolves the Value object to its primitive value.
	Write [▶ 2751]	Writes the value (via ADS)
	WriteAsync [▶ 2751]	Writes the value (via ADS)

Reference

[IValue Interface](#) [▶ 2745]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.100.2.1 IValue.Read Method

Reads the value (via ADS)

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
void Read()
```

Reference

[IValue Interface](#) [▶ 2745]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.100.2.2 IValue.ReadAsync Method

Reads the value (via ADS)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultAccess> ReadAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultAccess](#) [[▶ 3197](#)].

Task<ReadValueResult>.

Reference

[IValue Interface](#) [[▶ 2745](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.100.2.3 IValue.ResolveValue Method

Resolves the Value object to its primitive value.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object ResolveValue(
    bool resolveEnumToPrimitive
)
```

Parameters

resolveEnumToPrimitive	Type: System.Boolean if set to true EnumValue [▶ 2524]s are resolved to their primitives also.
------------------------	---

Return Value

Type: Object
System.Object.

Remarks

If the value is not primitive, this method returns the [IValue](#) [[▶ 2745](#)] itself.

Reference

[IValue Interface](#) [[▶ 2745](#)]

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.100.2.4 IValue.TryResolveValue Method

Tries to resolves the Value object to its primitive value.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool TryResolveValue (
    bool resolveEnumToPrimitive,
    out Object?? value
)
```

Parameters

resolveEnumToPrimitive	Type: System.Boolean if set to true EnumValue [▸ 2524] s are resolved to their primitives also.
value	Type: System.Object. The value.

Return Value

Type: Boolean

true if value can be resolved, false otherwise.

Reference

[IValue Interface \[▸ 2745\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.100.2.5 IValue.Write Method

Writes the value (via ADS)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void Write()
```

Reference

[IValue Interface \[▸ 2745\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.100.2.6 IValue.WriteAsync Method

Writes the value (via ADS)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultWriteAccess> WriteAsync(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: Task.ResultWriteAccess [▶ 3217].
Task<WriteValueResult>.

Reference

[IValue Interface](#) [▶ 2745]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.101 IValueAccessorProvider Interface

Interface IValueAccessorProvider

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public interface IValueAccessorProvider
```

The IValueAccessorProvider type exposes the following members.

Properties

	Name	Description
	ValueAccessor [▶ 2753]	Gets the value accessor.

Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.101.1 IValueAccessorProvider Properties

The [IValueAccessorProvider](#) [▶ 2752] type exposes the following members.

Properties

	Name	Description
	ValueAccessor [▶ 2753]	Gets the value accessor.

Reference

[IValueAccessorProvider Interface \[▶ 2752\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.101.1 IValueAccessorProvider.ValueAccessor Property

Gets the value accessor.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IAccessorRawValue? ValueAccessor { get; }
```

Property Value

Type: [IAccessorRawValue \[▶ 3184\]](#)

The value accessor.

Reference

[IValueAccessorProvider Interface \[▶ 2752\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.102 IValueAnySymbol Interface

Interface IValueAnySymbol

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229






Syntax

C#

```
public interface IValueAnySymbol : ISymbol,
    IAttributedInstance, IInstance, IBitSize
```




The IValueAnySymbol type exposes the following members.








Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469] .)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472] .)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472] .)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691] .)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549] .)

	Name	Description
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Methods

	Name	Description
	ReadAnyValue(Type) [▶ 2758]	Reads the value of this Value into a new created instance of the managed type
	ReadAnyValue(Type, Int32) [▶ 2759]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type
	ReadAnyValue.T.. [▶ 2759]	Reads the value of this Value into a new created instance of the managed type

	Name	Description
	ReadAnyValue.T. (Int32) [▶ 2760]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type
	ReadAnyValueAsync (Type, CancellationToken) [▶ 2761]	Reads the (AnyType) value asynchronously.
	ReadAnyValueAsync .T. (CancellationToken) [▶ 2761]	Reads the (AnyType) value asynchronously.
	UpdateAnyValue(O bject.) [▶ 2762]	Reads the value of this Value [▶ 2775] into the specified managed value.
	UpdateAnyValue(O bject., Int32) [▶ 2763]	Reads the value of this Value [▶ 2775] into the specified managed value.
	WriteAnyValue(Obj ect) [▶ 2764]	Writes the value represented by the managed value to this Value [▶ 2775]
	WriteAnyValue(Obj ect, Int32) [▶ 2764]	Writes the value represented by the managed value to this Value [▶ 2775]

Extension Methods

	Name	Description
	ReferencesExternalD ata [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021] .)

Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

[TwinCAT.TypeSystem.ISymbol \[▶ 2691\]](#)



Also see about this

-  [IValueAnySymbol.ReadAnyValueAsync Method \[▶ 2760\]](#)

6.11.102.1 IValueAnySymbol Properties

The [IValueAnySymbol \[▶ 2753\]](#) type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469] .)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472] .)

	Name	Description
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the Instance [▶ 2549] (Inherited from Instance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549] . (Inherited from Instance [▶ 2549].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from Instance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from Instance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the Instance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from Instance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the Instance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from Instance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this Instance [▶ 2549] is static. (Inherited from Instance [▶ 2549].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549] . (Inherited from Instance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Reference











[IValueAnySymbol Interface \[▶ 2753\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.102.2 IValueAnySymbol Methods

The [IValueAnySymbol](#) [▶ 2753] type exposes the following members.

Methods

	Name	Description
	ReadAnyValue(Type) [▶ 2758]	Reads the value of this Value [▶ 2753] into a new created instance of the managed type
	ReadAnyValue(Type, Int32) [▶ 2759]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type
	ReadAnyValue.T. [▶ 2759]	Reads the value of this Value [▶ 2753] into a new created instance of the managed type
	ReadAnyValue.T. (Int32) [▶ 2760]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type
	ReadAnyValueAsync (Type, CancellationToken) [▶ 2761]	Reads the (AnyType) value asynchronously.
	ReadAnyValueAsync .T. (CancellationToken) [▶ 2761]	Reads the (AnyType) value asynchronously.
	UpdateAnyValue(Object) [▶ 2762]	Reads the value of this Value [▶ 2775] into the specified managed value.
	UpdateAnyValue(Object, Int32) [▶ 2763]	Reads the value of this Value [▶ 2775] into the specified managed value.
	WriteAnyValue(Object) [▶ 2764]	Writes the value represented by the managed value to this Value [▶ 2775]
	WriteAnyValue(Object, Int32) [▶ 2764]	Writes the value represented by the managed value to this Value [▶ 2775]

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[IValueAnySymbol Interface](#) [▶ 2753]





[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

Also see about this

-  [IValueAnySymbol.ReadAnyValueAsync Method](#) [▶ 2760]

6.11.102.2.1 IValueAnySymbol.ReadAnyValue Method

Overload List

	Name	Description
	ReadAnyValue.T.. [▶ 2759]	Reads the value of this Value [▶ 2753] into a new created instance of the managed type
	ReadAnyValue.T. (Int32) [▶ 2760]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type
	ReadAnyValue(Type) [▶ 2758]	Reads the value of this Value [▶ 2753] into a new created instance of the managed type
	ReadAnyValue(Type , Int32) [▶ 2759]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type

Reference

[IValueAnySymbol Interface](#) [\[▶ 2753\]](#)

[TwinCAT.TypeSystem Namespace](#) [\[▶ 2083\]](#)

6.11.102.2.1.1 IValueAnySymbol.ReadAnyValue Method (Type)

Reads the value of this [Value](#) [\[▶ 2753\]](#) into a new created instance of the managed type

Namespace: [TwinCAT.TypeSystem](#) [\[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object ReadAnyValue(
    Type managedType
)
```

Parameters

managedType	Type: System.Type The tp.
-------------	------------------------------

Return Value

Type: Object
Read value (System.Object).

Reference

[IValueAnySymbol Interface](#) [\[▶ 2753\]](#)

[ReadAnyValue Overload](#) [\[▶ 2758\]](#)

[TwinCAT.TypeSystem Namespace](#) [\[▶ 2083\]](#)

[IValueAnySymbol.WriteAnyValue\(Object\)](#) [\[▶ 2764\]](#)

[IValueAnySymbol.UpdateAnyValue\(Object.\)](#) [\[▶ 2762\]](#)

6.11.102.2.1.2 IValueAnySymbol.ReadAnyValue Method (Type, Int32)

Reads the value of this [Value](#) [[2775](#)] into a new created instance of the managed type

Namespace: [TwinCAT.TypeSystem](#) [[2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object ReadAnyValue(
    Type managedType,
    int timeout
)
```

Parameters

managedType	Type: System.Type The tp.
timeout	Type: System.Int32 The timeout in ms.

Return Value

Type: Object

Read value (System.Object).

Reference

[IValueAnySymbol Interface](#) [[2753](#)]

[ReadAnyValue Overload](#) [[2758](#)]

[TwinCAT.TypeSystem Namespace](#) [[2083](#)]

[IValueAnySymbol.WriteAnyValue\(Object\)](#) [[2764](#)]

[IValueAnySymbol.UpdateAnyValue\(Object.\)](#) [[2762](#)]

6.11.102.2.1.3 IValueAnySymbol.ReadAnyValue.T. Method

Reads the value of this [Value](#) [[2753](#)] into a new created instance of the managed type

Namespace: [TwinCAT.TypeSystem](#) [[2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
T ReadAnyValue<T>()
```

Type Parameters

T	Type of the Value to be read.
---	-------------------------------

Return Value

Type: T

T.

Reference[IValueAnySymbol Interface \[▸ 2753\]](#)[ReadAnyValue Overload \[▸ 2758\]](#)[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)**6.11.102.2.1.4 IValueAnySymbol.ReadAnyValue.T. Method (Int32)**

Reads the value of this [Value \[▸ 2775\]](#) into a new created instance of the managed type

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
T ReadAnyValue<T>(
    int timeout
)
```

Parameters

timeout	Type: System.Int32 The timeout in ms.
---------	--

Type Parameters

T	The values type
---	-----------------

Return Value

Type: T

Read value (System.Object).

Reference[IValueAnySymbol Interface \[▸ 2753\]](#)[ReadAnyValue Overload \[▸ 2758\]](#)[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)[IValueAnySymbol.WriteAnyValue\(Object\) \[▸ 2764\]](#)[IValueAnySymbol.UpdateAnyValue\(Object.\) \[▸ 2762\]](#)**6.11.102.2.2 IValueAnySymbol.ReadAnyValueAsync Method****Overload List**

	Name	Description
	ReadAnyValueAsync.T.(CancellationTokentoken) [▸ 2761]	Reads the (AnyType) value asynchronously.

	Name	Description
	ReadAnyValueAsync (Type, CancellationTok en) [▶ 2761]	Reads the (AnyType) value asynchronously.

Reference

[IValueAnySymbol Interface](#) [▶ 2753]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

Also see about this

 [TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.102.2.2.1 IValueAnySymbol.ReadAnyValueAsync.T. Method (CancellationToken)

Reads the (AnyType) value asynchronously.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultReadValueAccess<T>> ReadAnyValueAsync<T>(
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Type Parameters

T

Return Value

Type: [Task.ResultReadValueAccess](#) [▶ 3211].T..

A task object that is representing the asynchronous 'ReadAnyValue' operation. The result will be returned in a [ResultReadValueAccess](#) [▶ 3209], which contains the [Value](#) [▶ 3213] and the [ErrorCode](#) [▶ 3202].

Reference

[IValueAnySymbol Interface](#) [▶ 2753]

[ReadAnyValueAsync Overload](#) [▶ 2760]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.102.2.2.2 IValueAnySymbol.ReadAnyValueAsync Method (Type, CancellationToken)

Reads the (AnyType) value asynchronously.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultReadValueAccess> ReadAnyValueAsync(
    Type managedType,
    CancellationToken cancel
)
```

Parameters

managedType	Type: System.Type Managed type of the value to read.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultReadValueAccess](#) [► 3209].

A task object that is representing the asynchronous 'ReadAnyValue' operation. The result will be returned in a [ResultReadValueAccess](#) [► 3209], which contains the [Value](#) [► 3213] and the [ErrorCode](#) [► 3202].

Reference



[IValueAnySymbol Interface](#) [► 2753]

[ReadAnyValueAsync Overload](#) [► 2760]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.102.2.3 IValueAnySymbol.UpdateAnyValue Method

Overload List

	Name	Description
	UpdateAnyValue(Object.) [► 2762]	Reads the value of this Value [► 2775] into the specified managed value.
	UpdateAnyValue(Object., Int32) [► 2763]	Reads the value of this Value [► 2775] into the specified managed value.

Reference

[IValueAnySymbol Interface](#) [► 2753]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.102.2.3.1 IValueAnySymbol.UpdateAnyValue Method (Object.)

Reads the value of this [Value](#) [► 2775] into the specified managed value.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void UpdateAnyValue(
    ref Object managedObject
)
```

Parameters

managedObject	Type: System.Object. The managed object.
---------------	---

Return Value

Type:
Read value (System.Object).

Reference

[IValueAnySymbol Interface \[▶ 2753\]](#)

[UpdateAnyValue Overload \[▶ 2762\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

[IValueAnySymbol.ReadAnyValue\(Type\) \[▶ 2758\]](#)

[IValueAnySymbol.WriteAnyValue\(Object\) \[▶ 2764\]](#)

6.11.102.2.3.2 IValueAnySymbol.UpdateAnyValue Method (Object., Int32)

Reads the value of this [Value \[▶ 2775\]](#) into the specified managed value.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void UpdateAnyValue(
    ref Object managedObject,
    int timeout
)
```

Parameters

managedObject	Type: System.Object. The managed object.
timeout	Type: System.Int32 The timeout.

Return Value

Type:
Read value (System.Object).

Reference

[IValueAnySymbol Interface \[▶ 2753\]](#)

[UpdateAnyValue Overload \[▶ 2762\]](#)



[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

[IValueAnySymbol.ReadAnyValue\(Type\)](#) [▶ 2758]

[IValueAnySymbol.WriteAnyValue\(Object\)](#) [▶ 2764]

6.11.102.2.4 IValueAnySymbol.WriteAnyValue Method

Overload List

	Name	Description
	WriteAnyValue(Object) [▶ 2764]	Writes the value represented by the managed value to this Value [▶ 2775]
	WriteAnyValue(Object, Int32) [▶ 2764]	Writes the value represented by the managed value to this Value [▶ 2775]

Reference

[IValueAnySymbol Interface](#) [▶ 2753]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.102.2.4.1 IValueAnySymbol.WriteAnyValue Method (Object)

Writes the value represented by the managed value to this [Value](#) [▶ 2775]

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void WriteAnyValue(
    Object managedValue
)
```

Parameters

managedValue	Type: System.Object The managed value.
--------------	---

Reference

[IValueAnySymbol Interface](#) [▶ 2753]

[WriteAnyValue Overload](#) [▶ 2764]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

[IValueAnySymbol.ReadAnyValue\(Type\)](#) [▶ 2758]

[IValueAnySymbol.UpdateAnyValue\(Object.\)](#) [▶ 2762]

6.11.102.2.4.2 IValueAnySymbol.WriteAnyValue Method (Object, Int32)

Writes the value represented by the managed value to this [Value](#) [▶ 2775]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void WriteAnyValue(
    Object managedValue,
    int timeout
)
```

Parameters

managedValue	Type: System.Object The managed value.
timeout	Type: System.Int32 The timeout in ms.

Reference

[IValueAnySymbol Interface](#) [[▶ 2753](#)]

[WriteAnyValue Overload](#) [[▶ 2764](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

[IValueAnySymbol.ReadAnyValue\(Type\)](#) [[▶ 2758](#)]

[IValueAnySymbol.UpdateAnyValue\(Object.\)](#) [[▶ 2762](#)]

6.11.103 IValueRawSymbol Interface

Interface IValueRawSymbol

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229






Syntax

C#

```
public interface IValueRawSymbol : IHierarchicalSymbol,
    ISymbol, IAttributedInstance, IInstance, IBitSize
```



The IValueRawSymbol type exposes the following members.






Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)


	Name	Description
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	HasValue [▶ 2768]	Gets a value indicating whether this IValueSymbol [▶ 2775] has a value.
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is a container type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueAccessor [▶ 2769]	Gets the value accessor.
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Methods

	Name	Description
	ReadRawValue [▶ 2770]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	ReadRawValue(Int32) [▶ 2771]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)

	Name	Description
	ReadRawValueAsynchronous [2772]	Reads the raw value of the IValueSymbol [2775] (Ads Read / Write) asynchronously.
	SetParent [2548]	Sets the parent of the Symbol (Inherited from IHierarchicalSymbol [2545].)
	WriteRawValue(Byte) [2772]	Writes the raw value of the IValueSymbol [2775] (Ads Read / Write)
	WriteRawValue(Byte, Int32) [2773]	Writes the raw value of the IValueSymbol [2775] (Ads Read / Write)
	WriteRawValueAsynchronous [2774]	Writes the raw value of the IValueSymbol [2775] (Ads Read / Write)

Events

	Name	Description
	RawValueChanged [2774]	Occurs when the RawValue of the IValueSymbol [2775] has changed.

Extension Methods

	Name	Description
	ReferencesExternalData [3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [3021].)

Reference







[TwinCAT.TypeSystem Namespace](#) [[2083](#)]

[TwinCAT.TypeSystem.IHierarchicalSymbol](#) [[2545](#)]

6.11.103.1 IValueRawSymbol Properties

The [IValueRawSymbol](#) [[2765](#)] type exposes the following members.

Properties

	Name	Description
	Attributes [2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [2469].)
	BitSize [2473]	Gets the size of the IDataType [2475] in bits. (Inherited from IBitSize [2472].)
	ByteSize [2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [2472].)
	Category [2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [2691].)
	Comment [2550]	Gets the comment of the IInstance [2549] (Inherited from IInstance [2549].)
	DataType [2551]	Gets the IDataType [2475] of the IInstance [2549]. (Inherited from IInstance [2549].)

	Name	Description
	HasValue [▶ 2768]	Gets a value indicating whether this IValueSymbol [▶ 2775] has a value.
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549] .)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549] .)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472] .)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472] .)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is a container type. (Inherited from ISymbol [▶ 2691] .)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691] .)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549] .)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691] .)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691] .)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691] .)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549] .)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549] .)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691] .)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472] .)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691] .)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549] . (Inherited from IInstance [▶ 2549] .)
	ValueAccessor [▶ 2769]	Gets the value accessor.
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469] .)

Reference

[IValueRawSymbol Interface \[▶ 2765\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.103.1.1 IValueRawSymbol.HasValue Property

Gets a value indicating whether this [IValueSymbol \[▶ 2775\]](#) has a value.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool HasValue { get; }
```

Property Value

Type: Boolean
true if this instance has value; otherwise, false.

Remarks

A VirtualSymbol does not support values, but in terms of the [IValueSymbol \[▸ 2775\]](#) definition, is a [IValueSymbol \[▸ 2775\]](#)

Reference

[IValueRawSymbol Interface \[▸ 2765\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.103.1.2 IValueRawSymbol.ValueAccessor Property

Gets the value accessor.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IAccessorRawValue? ValueAccessor { get; }
```

Property Value

Type: [IAccessorRawValue \[▸ 3184\]](#)
The value accessor.

Reference


[IValueRawSymbol Interface \[▸ 2765\]](#)







[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.103.2 IValueRawSymbol Methods

The [IValueRawSymbol \[▸ 2765\]](#) type exposes the following members.

Methods

	Name	Description
	ReadRawValue. [▸ 2770]	Reads the raw value of the IValueSymbol [▸ 2775] (Ads Read / Write)

	Name	Description
	ReadRawValue(Int32) [▶ 2771]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	ReadRawValueAsync [▶ 2772]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) asynchronously.
	SetParent [▶ 2548]	Sets the parent of the Symbol (Inherited from IHierarchicalSymbol [▶ 2545].)
	WriteRawValue(Byte) [▶ 2772]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	WriteRawValue(Byte, Int32) [▶ 2773]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	WriteRawValueAsync [▶ 2774]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)



Reference

[IValueRawSymbol Interface](#) [[▶ 2765](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.103.2.1 IValueRawSymbol.ReadRawValue Method

Overload List

	Name	Description
	ReadRawValue [▶ 2770]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)
	ReadRawValue(Int32) [▶ 2771]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write)

Reference

[IValueRawSymbol Interface](#) [[▶ 2765](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.103.2.1.1 IValueRawSymbol.ReadRawValue Method

Reads the raw value of the [IValueSymbol](#) [[▶ 2775](#)] (Ads Read / Write)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
byte[] ReadRawValue()
```

Field Value

Type: .Byte.
The raw value.

Return Value

Type: .Byte.
System.Byte[].

Reference

[IValueRawSymbol Interface \[► 2765\]](#)

[ReadRawValue Overload \[► 2770\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.103.2.1.2 IValueRawSymbol.ReadRawValue Method (Int32)

Reads the raw value of the [IValueSymbol \[► 2775\]](#) (Ads Read / Write)

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
byte[] ReadRawValue(  
    int timeout  
)
```

Parameters

timeout	Type: System.Int32 The timeout in ms.
---------	--

Field Value

Type: .Byte.
The raw value.

Return Value

Type: .Byte.
System.Byte[].

Remarks

A negative timeout indicates that the Default Timeout for the communication will be used. 0 means timeout is switched off.

Reference

[IValueRawSymbol Interface \[► 2765\]](#)

[ReadRawValue Overload \[► 2770\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.103.2.2 IValueRawSymbol.ReadRawValueAsync Method

Reads the raw value of the [IValueSymbol \[► 2775\]](#) (Ads Read / Write) asynchronously.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultReadRawAccess> ReadRawValueAsync (
    CancellationToken cancel
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Field Value

Type: [Task.ResultReadRawAccess \[► 3206\]](#).

The raw value.

Return Value

Type: [Task.ResultReadRawAccess \[► 3206\]](#).

System.Byte[].



Reference

[IValueRawSymbol Interface \[► 2765\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.103.2.3 IValueRawSymbol.WriteRawValue Method

Overload List

	Name	Description
	WriteRawValue(Byte e.) [► 2772]	Writes the raw value of the IValueSymbol [► 2775] (Ads Read / Write)
	WriteRawValue(Byte e., Int32) [► 2773]	Writes the raw value of the IValueSymbol [► 2775] (Ads Read / Write)

Reference

[IValueRawSymbol Interface \[► 2765\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.103.2.3.1 IValueRawSymbol.WriteRawValue Method (.Byte.)

Writes the raw value of the [IValueSymbol \[► 2775\]](#) (Ads Read / Write)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void WriteRawValue(  
    byte[] rawValue  
)
```

Parameters

rawValue	Type: <code>.System.Byte</code> . The value as byte array.
----------	---

Reference

[IValueRawSymbol Interface](#) [[▶ 2765](#)]

[WriteRawValue Overload](#) [[▶ 2772](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.103.2.3.2 IValueRawSymbol.WriteRawValue Method (.Byte., Int32)

Writes the raw value of the [IValueSymbol](#) [[▶ 2775](#)] (Ads Read / Write)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void WriteRawValue(  
    byte[] rawValue,  
    int timeout  
)
```

Parameters

rawValue	Type: <code>.System.Byte</code> . The value as byte array.
timeout	Type: <code>System.Int32</code> The timeout.

Field Value

Type:
The value.

Remarks

A negative timeout indicates that the Default Timeout for the communication will be used. 0 means timeout is switched off.

Reference

[IValueRawSymbol Interface](#) [[▶ 2765](#)]

[WriteRawValue Overload](#) [[▶ 2772](#)]

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.103.2.4 IValueRawSymbol.WriteRawValueAsync Method

Writes the raw value of the [IValueSymbol \[▶ 2775\]](#) (Ads Read / Write)

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultWriteAccess> WriteRawValueAsync (
    byte[] rawValue,
    CancellationToken cancel
)
```

Parameters

rawValue	Type: .System.Byte. The value as byte array.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWriteAccess \[▶ 3217\]](#).

A task that represents the asynchronous read operation. The [ResultRead \[▶ 1143\]](#) parameter contains the total number of bytes read into the buffer ([ReadBytes \[▶ 1145\]](#)) and the [ErrorCode \[▶ 1120\]](#) after execution..

Reference


[IValueRawSymbol Interface \[▶ 2765\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.103.3 IValueRawSymbol Events

The [IValueRawSymbol \[▶ 2765\]](#) type exposes the following members.

Events

	Name	Description
	RawValueChanged [▶ 2774]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed.

Reference

[IValueRawSymbol Interface \[▶ 2765\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.103.3.1 IValueRawSymbol.RawValueChanged Event

Occurs when the RawValue of the [IValueSymbol \[▶ 2775\]](#) has changed.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
event EventHandler<RawValueChangedEventArgs> RawValueChanged
```

Value

Type: System.EventHandler.RawValueChangedEventArgs [▶ 2813].

Reference

[IValueRawSymbol Interface \[▶ 2765\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.104 IValueSymbol Interface

Interface for a [ISymbol \[▶ 2691\]](#) that supports values.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229












Syntax


















C#

```
public interface IValueSymbol : IValueRawSymbol,
    IHierarchicalSymbol, ISymbol, IAttributedInstance, IInstance, IBitSize
```





The IValueSymbol type exposes the following members.












Properties

	Name	Description
	AccessRights [▶ 2780]	Gets the access rights.
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469] .)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472] .)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472] .)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691] .)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549] .)
	Connection [▶ 2781]	Gets the connection that produces values for this IValueSymbol
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549] . (Inherited from IInstance [▶ 2549] .)
	HasValue [▶ 2768]	Gets a value indicating whether this IValueSymbol has a value. (Inherited from IValueRawSymbol [▶ 2765] .)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549] .)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549] .)



	Name	Description
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
	NotificationSettings [▶ 2781]	Gets or sets the notification settings.
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	ValueAccessor [▶ 2769]	Gets the value accessor. (Inherited from IValueRawSymbol [▶ 2765].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Methods





	Name	Description
	ReadRawValue . [▶ 2770]	Reads the raw value of the IValueSymbol (Ads Read / Write) (Inherited from IValueRawSymbol [▶ 2765].)
	ReadRawValue (Int32) [▶ 2771]	Reads the raw value of the IValueSymbol (Ads Read / Write) (Inherited from IValueRawSymbol [▶ 2765].)
	ReadRawValue Asyn c [▶ 2772]	Reads the raw value of the IValueSymbol (Ads Read / Write) asynchronously. (Inherited from IValueRawSymbol [▶ 2765].)
	ReadValue . [▶ 2784]	Reads the Value of the IValueSymbol

	Name	Description
	ReadValue(Int32) [▶ 2785]	Reads the Value of the IValueSymbol
	ReadValueAsync [▶ 2785]	Reads the Value of the IValueSymbol asynchronously.
	SetParent [▶ 2548]	Sets the parent of the Symbol (Inherited from IHierarchicalSymbol [▶ 2545].)
	TryReadValue [▶ 2786]	Reads the Value of the IValueSymbol
	TryWriteValue [▶ 2787]	Writes the Value of the IValueSymbol
	WriteRawValue(Byte) [▶ 2772]	Writes the raw value of the IValueSymbol (Ads Read / Write) (Inherited from IValueRawSymbol [▶ 2765].)
	WriteRawValue(Byte, Int32) [▶ 2773]	Writes the raw value of the IValueSymbol (Ads Read / Write) (Inherited from IValueRawSymbol [▶ 2765].)
	WriteRawValueAsync [▶ 2774]	Writes the raw value of the IValueSymbol (Ads Read / Write) (Inherited from IValueRawSymbol [▶ 2765].)
	WriteValue(Object) [▶ 2788]	Writes the Value of the IValueSymbol
	WriteValue(Object, Int32) [▶ 2788]	Writes the Value of the IValueSymbol
	WriteValueAsync [▶ 2789]	Writes the Value of the IValueSymbol

Events

	Name	Description
	RawValueChanged [▶ 2774]	Occurs when the RawValue of the IValueSymbol has changed. (Inherited from IValueRawSymbol [▶ 2765].)
	ValueChanged [▶ 2790]	Occurs when the (Primitive) value of the IValueSymbol has changed.

Extension Methods

	Name	Description
	PollValues(IObservable<Unit>) [▶ 1355]	Overloaded. Poll symbol values on trigger signals. (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues(TimeSpan) [▶ 1356]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues(IObservable<Unit>, Boolean) [▶ 1357]	Overloaded. Polls symbol values on trigger signals. (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues(TimeSpan, Boolean) [▶ 1358]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by ValueSymbolExtensions [▶ 1333].)

	Name	Description
	PollValues.T. (IObservable.Unit.) [▶ 1362]	Overloaded. Poll symbol values as a value sequence on trigger signals (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
 	PollValues.T. (TimeSpan) [▶ 1363]	Overloaded. Polls the symbol as value sequence of object values with a specified period time (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
 	PollValues.T. (IObservable.Unit. , Func.ResultReadValueAccess2.IValueSymbol , Object , T .) [▶ 1365]	Overloaded. Poll symbol values on trigger signals (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues.T. (IObservable.Unit. , Boolean) [▶ 1364]	Overloaded. Poll symbol values on trigger signals (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues.T. (TimeSpan , Boolean) [▶ 1367]	Overloaded. Polls the symbol as value sequence of object values with a specified period time (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues2(IObservable.Unit.) [▶ 1368]	Overloaded. Poll symbol values as a sequence of annotated results (Value + ErrorCode) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues2(TimeSpan) [▶ 1369]	Overloaded. Poll symbol values with communication return codes. (Defined by ValueSymbolExtensions [▶ 1333].)
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
 	WhenValueChanged [▶ 1344]	Gets an observable sequence when the value of the IValueSymbol has changed. (Defined by ValueSymbolExtensions [▶ 1333].)
 	WriteValues(IObservable.Object.) [▶ 1347]	Overloaded. Subscribes the IValueSymbol to an observable sequence of values and writes them to the IValueSymbol . (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues(IObservable.Object., Action.Exception.) [▶ 1348]	Overloaded. Subscribes the IValueSymbol to an observable sequence of values and writes them to the IValueSymbol . (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues(IObservable.Object., CancellationToken) [▶ 1349]	Overloaded. Subscribes the IValueSymbol to an observable sequence of values and writes them to the IValueSymbol . (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues(IObservable.Object.,	Overloaded.

	Name	Description
	Action.Exception, CancellationToken [▶ 1350]	Subscribes the IValueSymbol to an observable sequence of values and writes them to the IValueSymbol. (Defined by ValueSymbolExtensions [▶ 1333].)

Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]












[TwinCAT.TypeSystem.IValueRawSymbol](#) [▶ 2765]

6.11.104.1 IValueSymbol Properties

The [IValueSymbol](#) [▶ 2775] type exposes the following members.

Properties

	Name	Description
	AccessRights [▶ 2780]	Gets the access rights.
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
	Connection [▶ 2781]	Gets the connection that produces values for this IValueSymbol [▶ 2775]
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
	HasValue [▶ 2768]	Gets a value indicating whether this IValueSymbol [▶ 2775] has a value. (Inherited from IValueRawSymbol [▶ 2765].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)

	Name	Description
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the Instance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from Instance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this Instance [▶ 2549] is static. (Inherited from Instance [▶ 2549].)
	NotificationSettings [▶ 2781]	Gets or sets the notification settings.
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from BitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from Instance [▶ 2549].)
	ValueAccessor [▶ 2769]	Gets the value accessor. (Inherited from IValueRawSymbol [▶ 2765].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Reference

[IValueSymbol Interface](#) [[▶ 2775](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.104.1.1 IValueSymbol.AccessRights Property

Gets the access rights.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
SymbolAccessRights AccessRights { get; }
```

Property Value

Type: [SymbolAccessRights](#) [[▶ 2928](#)]

The access rights.

Reference

[IValueSymbol Interface](#) [[▶ 2775](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.104.1.2 IValueSymbol.Connection Property

Gets the connection that produces values for this [IValueSymbol](#) [[▶ 2775](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IConnection? Connection { get; }
```

Property Value

Type: [IConnection](#) [[▶ 79](#)]

The connection object.

Reference

[IValueSymbol Interface](#) [[▶ 2775](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.104.1.3 IValueSymbol.NotificationSettings Property

Gets or sets the notification settings.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
INotificationSettings? NotificationSettings { get; set; }
```

Property Value

Type: [INotificationSettings](#) [[▶ 1098](#)]

The notification settings.

Remarks

The NotificationSettings will be inherited from [Parent](#) [[▶ 2696](#)] if the setting is not overwritten.

Reference


[IValueSymbol Interface](#) [[▶ 2775](#)]















[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.104.2 IValueSymbol Methods





The [IValueSymbol](#) [[▶ 2775](#)] type exposes the following members.













Methods

	Name	Description
	ReadRawValue . [▶ 2770]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from IValueRawSymbol [▶ 2765].)

	Name	Description
	ReadRawValue(Int32) [▶ 2771]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from IValueRawSymbol [▶ 2765].)
	ReadRawValueAsync [▶ 2772]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) asynchronously. (Inherited from IValueRawSymbol [▶ 2765].)
	ReadValue. [▶ 2784]	Reads the Value of the IValueSymbol [▶ 2775]
	ReadValue(Int32) [▶ 2785]	Reads the Value of the IValueSymbol [▶ 2775]
	ReadValueAsync [▶ 2785]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously.
	SetParent [▶ 2548]	Sets the parent of the Symbol (Inherited from IHierarchicalSymbol [▶ 2545].)
	TryReadValue [▶ 2786]	Reads the Value of the IValueSymbol [▶ 2775]
	TryWriteValue [▶ 2787]	Writes the Value of the IValueSymbol [▶ 2775]
	WriteRawValue(Byte) [▶ 2772]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from IValueRawSymbol [▶ 2765].)
	WriteRawValue(Byte, Int32) [▶ 2773]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from IValueRawSymbol [▶ 2765].)
	WriteRawValueAsync [▶ 2774]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from IValueRawSymbol [▶ 2765].)
	WriteValue(Object) [▶ 2788]	Writes the Value of the IValueSymbol [▶ 2775]
	WriteValue(Object, Int32) [▶ 2788]	Writes the Value of the IValueSymbol [▶ 2775]
	WriteValueAsync [▶ 2789]	Writes the Value of the IValueSymbol [▶ 2775]

Extension Methods

	Name	Description
	PollValues(IObservableUnit) [▶ 1355]	Overloaded. Poll symbol values on trigger signals. (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues(TimeSpan) [▶ 1356]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues(IObservableUnit, Boolean) [▶ 1357]	Overloaded. Polls symbol values on trigger signals. (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues(TimeSpan, Boolean) [▶ 1358]	Overloaded. Polls the symbol as value sequence of object values with a specified period time. (Defined by ValueSymbolExtensions [▶ 1333].)

	Name	Description
	PollValues.T. (IObservable.Unit.) [▶ 1362]	Overloaded. Poll symbol values as a value sequence on trigger signals (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
 	PollValues.T. (TimeSpan) [▶ 1363]	Overloaded. Polls the symbol as value sequence of object values with a specified period time (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
 	PollValues.T. (IObservable.Unit. , Func.ResultReadValueAccess2.IValueSymbol, Object., T.) [▶ 1365]	Overloaded. Poll symbol values on trigger signals (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues.T. (IObservable.Unit. , Boolean) [▶ 1364]	Overloaded. Poll symbol values on trigger signals (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues.T. (TimeSpan , Boolean) [▶ 1367]	Overloaded. Polls the symbol as value sequence of object values with a specified period time (typed) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues2(IObservable.Unit.) [▶ 1368]	Overloaded. Poll symbol values as a sequence of annotated results (Value + ErrorCode) (Defined by ValueSymbolExtensions [▶ 1333].)
	PollValues2(TimeSpan) [▶ 1369]	Overloaded. Poll symbol values with communication return codes. (Defined by ValueSymbolExtensions [▶ 1333].)
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
 	WhenValueChanged [▶ 1344]	Gets an observable sequence when the value of the IValueSymbol [▶ 2775] has changed. (Defined by ValueSymbolExtensions [▶ 1333].)
 	WriteValues(IObservable.Object.) [▶ 1347]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775]. (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues(IObservable.Object., Action.Exception.) [▶ 1348]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775]. (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues(IObservable.Object., Cancellation.Token) [▶ 1349]	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775]. (Defined by ValueSymbolExtensions [▶ 1333].)
	WriteValues(IObservable.Object.,	Overloaded. Subscribes the IValueSymbol [▶ 2775] to an observable sequence of values and writes them to the IValueSymbol [▶ 2775]. (Defined by ValueSymbolExtensions [▶ 1333].)

	Name	Description
	Action.Exception, CancellationToken [▶ 1350]	



Reference

[IValueSymbol Interface](#) [▶ 2775]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.104.2.1 IValueSymbol.ReadValue Method

Overload List

	Name	Description
	ReadValue. [▶ 2784]	Reads the Value of the IValueSymbol [▶ 2775]
	ReadValue(Int32) [▶ 2785]	Reads the Value of the IValueSymbol [▶ 2775]

Reference

[IValueSymbol Interface](#) [▶ 2775]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.104.2.1.1 IValueSymbol.ReadValue Method

Reads the Value of the [IValueSymbol](#) [▶ 2775]

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object ReadValue ()
```

Field Value

Type: Object
The value.

Return Value

Type: Object
System.Object.

Remarks

Calling on primitive types, a call of this method will return the primitive value. On complex types (structures and arrays) it depends on the [ISymbolLoader](#) [▶ 2714] settings what will happen. In non dynamic modes: the raw byte Array will be returned, in dynamic mode: A Value will be created on the fly.

Reference

[IValueSymbol Interface](#) [▶ 2775]

[ReadValue Overload \[▶ 2784\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.104.2.1.2 IValueSymbol.ReadValue Method (Int32)

Reads the Value of the [IValueSymbol \[▶ 2775\]](#)

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object ReadValue(  
    int timeout  
)
```

Parameters

timeout	Type: System.Int32 The timeout in ms.
---------	--

Field Value

Type: Object
The value.

Return Value

Type: Object
System.Object.

Remarks

Calling on primitive types, a call of this method will return the primitive value. On complex types (structures and arrays) it depends on the [ISymbolLoader \[▶ 2714\]](#) settings what will happen. In non dynamic modes: the raw byte Array will be returned, in dynamic mode: A Value will be created on the fly. A negative timeout indicates that the Default Timeout for the communication will be used. 0 means timeout is switched off.

Reference

[IValueSymbol Interface \[▶ 2775\]](#)

[ReadValue Overload \[▶ 2784\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.104.2.2 IValueSymbol.ReadValueAsync Method

Reads the Value of the [IValueSymbol \[▶ 2775\]](#) asynchronously.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultReadValueAccess> ReadValueAsync(  
    CancellationToken cancel  
)
```

Parameters

cancel	Type: System.Threading.CancellationToken The cancellation token.
--------	---

Return Value

Type: [Task.ResultReadValueAccess](#) [[▸ 3209](#)].

A tasks that represents the asynchronous 'ReadValue' operation. The read result is stored in the [ResultReadValueAccess](#) [[▸ 3209](#)] return value and contains the [Value](#) [[▸ 3213](#)] and the [ErrorCode](#) [[▸ 3202](#)].

Remarks

Calling on primitive types, a call of this method will return the primitive value. On complex types (structures and arrays) it depends on the [ISymbolLoader](#) [[▸ 2714](#)] settings what will happen. In non dynamic modes: the raw byte Array will be returned, in dynamic mode: A Value will be created on the fly.

Reference

[IValueSymbol Interface](#) [[▸ 2775](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.104.2.3 IValueSymbol.TryReadValue Method

Reads the Value of the [IValueSymbol](#) [[▸ 2775](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
int TryReadValue(
    int timeout,
    out Object?? value
)
```

Parameters

timeout	Type: System.Int32 The timeout in ms.
value	Type: System.Object. The symbol value.

Return Value

Type: Int32
The error code.

Remarks

Calling on primitive types, a call of this method will return the primitive value. On complex types (structures and arrays) it depends on the [ISymbolLoader](#) [[▸ 2714](#)] settings what will happen. In non dynamic modes: the raw byte Array will be returned, in dynamic mode: A Value will be created on the fly. A negative timeout indicates that the Default Timeout for the communication will be used. 0 means timeout is switched off.

Reference

[IValueSymbol Interface](#) [[▸ 2775](#)]

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.104.2.4 IValueSymbol.TryWriteValue Method

Writes the Value of the [IValueSymbol \[▶ 2775\]](#)

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int TryWriteValue(
    Object value,
    int timeout
)
```

Parameters

value	Type: System.Object The value.
timeout	Type: System.Int32 The timeout in ms.

Return Value

Type: Int32
The error code.

Remarks

Calling on primitive types, a call of this method will directly write this Value. On complex types (structs and arrays) it depends on the [ISymbolLoader \[▶ 2714\]](#) settings what will happen. In non dynamic modes: Only byte Arrays (of correct size) can be written) in dynamic mode: A Value that represents the value will be accepted also. A negative timeout indicates that the Default Timeout for the communication will be used. 0 means timeout is switched off.



Reference

[IValueSymbol Interface \[▶ 2775\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.104.2.5 IValueSymbol.WriteValue Method

Overload List

	Name	Description
	WriteValue(Object) [▶ 2788]	Writes the Value of the IValueSymbol [▶ 2775]
	WriteValue(Object, Int32) [▶ 2788]	Writes the Value of the IValueSymbol [▶ 2775]

Reference

[IValueSymbol Interface \[▶ 2775\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.104.2.5.1 IValueSymbol.WriteValue Method (Object)

Writes the Value of the [IValueSymbol](#) [[▶ 2775](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void WriteValue(
    Object value
)
```

Parameters

value	Type: System.Object The value.
-------	-----------------------------------

Remarks

Calling on primitive types, a call of this method will directly write this Value. On complex types (structs and arrays) it depends on the [ISymbolLoader](#) [[▶ 2714](#)] settings what will happen. In non dynamic modes: Only byte Arrays (of correct size) can be written) in dynamic mode: A Value that represents the value will be accepted also.

Reference

[IValueSymbol Interface](#) [[▶ 2775](#)]

[WriteValue Overload](#) [[▶ 2787](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.104.2.5.2 IValueSymbol.WriteValue Method (Object, Int32)

Writes the Value of the [IValueSymbol](#) [[▶ 2775](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
void WriteValue(
    Object value,
    int timeout
)
```

Parameters

value	Type: System.Object The value.
timeout	Type: System.Int32 The timeout in ms.

Remarks

Calling on primitive types, a call of this method will directly write this Value. On complex types (structs and arrays) it depends on the [ISymbolLoader \[▶ 2714\]](#) settings what will happen. In non dynamic modes: Only byte Arrays (of correct size) can be written) in dynamic mode: A Value that represents the value will be accepted also. A negative timeout indicates that the Default Timeout for the communication will be used. 0 means timeout is switched off.

Reference

[IValueSymbol Interface \[▶ 2775\]](#)

[WriteValue Overload \[▶ 2787\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.104.2.6 IValueSymbol.WriteValueAsync Method

Writes the Value of the [IValueSymbol \[▶ 2775\]](#)

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultWriteAccess> WriteValueAsync(
    Object value,
    CancellationToken cancel
)
```

Parameters

value	Type: System.Object The value.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWriteAccess \[▶ 3217\]](#).

A tasks that represents the asynchronous 'ReadValue' operation. The read result is stored in the [ResultWriteAccess \[▶ 3217\]](#) return value and contains the [ErrorCode \[▶ 3202\]](#).

Remarks

Calling on primitive types, a call of this method will directly write this Value. On complex types (structs and arrays) it depends on the [ISymbolLoader \[▶ 2714\]](#) settings what will happen. In non dynamic modes: Only byte Arrays (of correct size) can be written) in dynamic mode: A Value that represents the value will be accepted also.

Reference



[IValueSymbol Interface \[▶ 2775\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.104.3 IValueSymbol Events

The [IValueSymbol \[▶ 2775\]](#) type exposes the following members.

Events

	Name	Description
	RawValueChanged [▶ 2774]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed. (Inherited from IValueRawSymbol [▶ 2765] .)
	ValueChanged [▶ 2790]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed.

Reference

[IValueSymbol Interface](#) [\[▶ 2775\]](#)

[TwinCAT.TypeSystem Namespace](#) [\[▶ 2083\]](#)

6.11.104.3.1 IValueSymbol.ValueChanged Event

Occurs when the (Primitive) value of the [IValueSymbol](#) [\[▶ 2775\]](#) has changed.

Namespace: [TwinCAT.TypeSystem](#) [\[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
event EventHandler<ValueChangedEventArgs> ValueChanged
```

Value

Type: System.EventHandler.ValueChangedEventArgs [\[▶ 2971\]](#).

Reference

[IValueSymbol Interface](#) [\[▶ 2775\]](#)

[TwinCAT.TypeSystem Namespace](#) [\[▶ 2083\]](#)

6.11.105 IVirtualStructInstance Interface

Virtual Struct instance interface.

Namespace: [TwinCAT.TypeSystem](#) [\[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax

C#

```
public interface IVirtualStructInstance : IStructInstance,
    IInterfaceInstance, ISymbol, IAttributedInstance, IInstance, IBitSize,
    IRpcCallableInstance
```







The IVirtualStructInstance type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469] .)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472] .)

	Name	Description
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
	Comment [▶ 2550]	Gets the comment of the Instance [▶ 2549] (Inherited from Instance [▶ 2549].)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the Instance [▶ 2549] . (Inherited from Instance [▶ 2549].)
	HasRpcMethods [▶ 3018]	Gets a value indicating whether this instance has RPC methods (Inherited from InterfaceInstance [▶ 3014].)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from Instance [▶ 2549].)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from Instance [▶ 2549].)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
	IsPointer [▶ 2552]	Indicates that the Instance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from Instance [▶ 2549].)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
	IsReference [▶ 2552]	Indicates that the Instance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from Instance [▶ 2549].)
	IsStatic [▶ 2553]	Gets a value indicating whether this Instance [▶ 2549] is static. (Inherited from Instance [▶ 2549].)
	MemberInstances [▶ 3018]	Gets the member instances of the Struct Instance [▶ 3014] . (Inherited from InterfaceInstance [▶ 3014].)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
	RpcMethods [▶ 2607]	Gets the Method descriptions for the IRpcCallableType [▶ 2624] (Inherited from IRpcCallableInstance [▶ 2606].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549] . (Inherited from Instance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from AttributedInstance [▶ 2469].)

Methods

	Name	Description
	AddMember [▶ 2794]	Adds the member.
	InvokeRpcMethod(String, Object.) [▶ 2609]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)
	InvokeRpcMethod(String, Object., Object.) [▶ 2610]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)
	InvokeRpcMethodAsync [▶ 2614]	Invokes the specified RPC Method asynchronously (Inherited from IRpcCallableInstance [▶ 2606].)
	TryInvokeRpcMethod(String, Object., Object.) [▶ 2619]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)
	TryInvokeRpcMethod(String, Object., Object., Object.) [▶ 2620]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Remarks

Virtual struct instance are used to create a TreeView from the flat list of symbols.

Reference




[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

[TwinCAT.TypeSystem.IStructInstance](#) [[▶ 2666](#)]

6.11.105.1 IVirtualStructInstance Properties

The [IVirtualStructInstance](#) [[▶ 2790](#)] type exposes the following members.

Properties

	Name	Description
	Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
	ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)

	Name	Description
	Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691] .)
	Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549] .)
	DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549] . (Inherited from IInstance [▶ 2549] .)
	HasRpcMethods [▶ 3018]	Gets a value indicating whether this instance has RPC methods (Inherited from IInterfaceInstance [▶ 3014] .)
	InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549] .)
	InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549] .)
	IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472] .)
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472] .)
	IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691] .)
	IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691] .)
	IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549] .)
	IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691] .)
	IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691] .)
	IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691] .)
	IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549] .)
	IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549] .)
	MemberInstances [▶ 3018]	Gets the member instances of the Struct Instance [▶ 3014] . (Inherited from IInterfaceInstance [▶ 3014] .)
	Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691] .)
	RpcMethods [▶ 2607]	Gets the Method descriptions for the IRpcCallableType [▶ 2624] (Inherited from IRpcCallableInstance [▶ 2606] .)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472] .)
	SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691] .)
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549] . (Inherited from IInstance [▶ 2549] .)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469] .)

Reference







[IVirtualStructInstance Interface](#) [▸ 2790]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.105.2 IVirtualStructInstance Methods

The [IVirtualStructInstance](#) [▸ 2790] type exposes the following members.

Methods

	Name	Description
	AddMember [▸ 2794]	Adds the member.
	InvokeRpcMethod(String, .Object.) [▸ 2609]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▸ 2606].)
	InvokeRpcMethod(String, .Object., .Object.) [▸ 2610]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▸ 2606].)
	InvokeRpcMethodAsync [▸ 2614]	Invokes the specified RPC Method asynchronously (Inherited from IRpcCallableInstance [▸ 2606].)
	TryInvokeRpcMethod(String, .Object., .Object.) [▸ 2619]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▸ 2606].)
	TryInvokeRpcMethod(String, .Object., .Object., .Object.) [▸ 2620]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▸ 2606].)

Extension Methods

	Name	Description
	ReferencesExternalData [▸ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▸ 3021].)

Reference

[IVirtualStructInstance Interface](#) [▸ 2790]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.105.2.1 IVirtualStructInstance.AddMember Method

Adds the member.

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool AddMember(
    ISymbol memberInstance,
    IVirtualStructInstance parent
)
```

Parameters

memberInstance	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The member instance.
parent	Type: TwinCAT.TypeSystem.IVirtualStructInstance [▶ 2790] The parent struct instance. Usually the this pointer.

Return Value

Type: Boolean

Reference

[IVirtualStructInstance Interface](#) [[▶ 2790](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.106 MarshalException Class

Common Marshalling Exception

Inheritance Hierarchy

System.Object
 System.Exception
 [TwinCAT.AdsException](#) [[▶ 61](#)]
 TwinCAT.TypeSystem.MarshalException

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229




Syntax






C#

```
[SerializableAttribute]
public sealed class MarshalException : AdsException
```









The MarshalException type exposes the following members.

Constructors






	Name	Description
	MarshalException . [▶ 2797]	Initializes a new instance of the MarshalException class.
	MarshalException (DataType) [▶ 2798]	Initializes a new instance of the MarshalException class.
	MarshalException (String) [▶ 2798]	Initializes a new instance of the MarshalException class.


	Name	Description
	MarshalException(I DataType, Object) [▶ 2799]	Initializes a new instance of the MarshalException class.
	MarshalException(I DataType, Type) [▶ 2800]	Initializes a new instance of the MarshalException class.
	MarshalException(Str ing, Exception) [▶ 2799]	Initializes a new Instance of the AdsException class.
	MarshalException(I DataType, Type, String) [▶ 2800]	Initializes a new instance of the MarshalException class.
	MarshalException(I nstance, Type, MemberInfo) [▶ 2801]	Initializes a new instance of the MarshalException class.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)









	Name	Description
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.106.1 MarshalException Constructor

Overload List

	Name	Description
	MarshalException. [▶ 2797]	Initializes a new instance of the MarshalException [▶ 2795] class.
	MarshalException(I DataType) [▶ 2798]	Initializes a new instance of the MarshalException [▶ 2795] class.
	MarshalException(St ring) [▶ 2798]	Initializes a new instance of the MarshalException [▶ 2795] class.
	MarshalException(I DataType, Object) [▶ 2799]	Initializes a new instance of the MarshalException [▶ 2795] class.
	MarshalException(I DataType, Type) [▶ 2800]	Initializes a new instance of the MarshalException [▶ 2795] class.
	MarshalException(St ring, Exception) [▶ 2799]	Initializes a new Instance of the AdsException class.
	MarshalException(I DataType, Type, String) [▶ 2800]	Initializes a new instance of the MarshalException [▶ 2795] class.
	MarshalException(I nstance, Type, MemberInfo) [▶ 2801]	Initializes a new instance of the MarshalException [▶ 2795] class.

Reference

[MarshalException Class \[▶ 2795\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.106.1.1 MarshalException Constructor

Initializes a new instance of the [MarshalException \[▶ 2795\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public MarshalException()
```

Reference

[MarshalException Class \[► 2795\]](#)

[MarshalException Overload \[► 2797\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.106.1.2 MarshalException Constructor (String)

Initializes a new instance of the [MarshalException \[► 2795\]](#) class.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public MarshalException(  
    string message  
)
```

Parameters

message	Type: System.String The message.
---------	-------------------------------------

Reference

[MarshalException Class \[► 2795\]](#)

[MarshalException Overload \[► 2797\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.106.1.3 MarshalException Constructor (IDataType)

Initializes a new instance of the [MarshalException \[► 2795\]](#) class.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public MarshalException(  
    IDataType source  
)
```

Parameters

source	Type: TwinCAT.TypeSystem.IDataType [► 2475] The source dataType.
--------	---

Reference[MarshalException Class \[► 2795\]](#)[MarshalException Overload \[► 2797\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.106.1.4 MarshalException Constructor (String, Exception)**

Initializes a new Instance of the AdsException class.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public MarshalException(
    string message,
    Exception innerException
)
```

Parameters

message	Type: System.String The message.
innerException	Type: System.Exception The inner exception.

Reference[MarshalException Class \[► 2795\]](#)[MarshalException Overload \[► 2797\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.106.1.5 MarshalException Constructor (IDataType, Object)**Initializes a new instance of the [MarshalException \[► 2795\]](#) class.**Namespace:** [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public MarshalException(
    IDataType target,
    Object? value
)
```

Parameters

target	Type: TwinCAT.TypeSystem.IDataType [► 2475] The datatype information.
value	Type: System.Object The value.

Reference[MarshalException Class \[► 2795\]](#)[MarshalException Overload \[► 2797\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.106.1.6 MarshalException Constructor (IDataType, Type)**Initializes a new instance of the [MarshalException \[► 2795\]](#) class.**Namespace:** [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public MarshalException(
    IDataType source,
    Type target
)
```

Parameters

source	Type: TwinCAT.TypeSystem.IDataType [► 2475] The source.
target	Type: System.Type The target.

Reference[MarshalException Class \[► 2795\]](#)[MarshalException Overload \[► 2797\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.106.1.7 MarshalException Constructor (IDataType, Type, String)**Initializes a new instance of the [MarshalException \[► 2795\]](#) class.**Namespace:** [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public MarshalException(
    IDataType source,
    Type target,
    string message
)
```

Parameters

source	Type: TwinCAT.TypeSystem.IDataType [► 2475] The source.
target	Type: System.Type The target.

message	Type: System.String The message.
---------	-------------------------------------

Reference

[MarshalException Class \[▶ 2795\]](#)

[MarshalException Overload \[▶ 2797\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.106.1.8 MarshalException Constructor (Instance, Type, MemberInfo)

Initializes a new instance of the [MarshalException \[▶ 2795\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public MarshalException(
    Instance instance,
    Type tp,
    MemberInfo member
)
```

Parameters

instance	Type: TwinCAT.TypeSystem.Instance [▶ 2549] The instance.
tp	Type: System.Type The type.
member	Type: System.Reflection.MemberInfo The member.

Reference

[MarshalException Class \[▶ 2795\]](#)





[MarshalException Overload \[▶ 2797\]](#)





[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.106.2 MarshalException Properties

The [MarshalException \[▶ 2795\]](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)

	Name	Description
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Reference







[MarshalException Class \[► 2795\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.106.3 MarshalException Methods

The [MarshalException \[► 2795\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Reference

[MarshalException Class \[► 2795\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.107 MemberCollection Class

Collection of [IMember \[► 2561\]](#) objects.

Inheritance Hierarchy

System.Object

[TwinCAT.TypeSystem.Generic.InstanceCollection \[► 3104\].IMember \[► 2561\].](#)

[TwinCAT.TypeSystem.MemberCollection](#)

[TwinCAT.TypeSystem.AlignedMemberCollection \[► 2973\]](#)

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229



Syntax

C#











```
public class MemberCollection : InstanceCollection<IMember>,
    IMemberCollection, IInstanceCollection<IMember>, IList<IMember>,
    ICollection<IMember>, IEnumerable<IMember>, IEnumerable
```

The MemberCollection type exposes the following members.






Constructors






















	Name	Description
	MemberCollection [▶ 2805]	Initializes a new instance of the MemberCollection class.
	MemberCollection(IEnumerable.IMember) [▶ 2805]	Initializes a new instance of the MemberCollection class (copy constructor)

Properties

	Name	Description
	Count [▶ 3107]	Gets the collection count. (Inherited from InstanceCollection.T. [▶ 3104].)
	Empty [▶ 2806]	Returns an Empty Member Collection.
	InnerList [▶ 3108]	Gets the IList of instances. (Inherited from InstanceCollection.T. [▶ 3104].)
	InnerPathDict [▶ 3108]	The Path dictionary (Inherited from InstanceCollection.T. [▶ 3104].)
	Instances [▶ 2807]	Gets the Instance members (non static)
	IsReadOnly [▶ 3109]	Gets a value indicating whether this instance is read only. (Inherited from InstanceCollection.T. [▶ 3104].)
	Item.Int32. [▶ 3109]	Gets or sets the Instance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	Item.String. [▶ 3110]	Gets the Instance [▶ 2549] with the specified instance path. (Inherited from InstanceCollection.T. [▶ 3104].)
	Mode [▶ 3111]	The mode this InstanceCollection.T. [▶ 3104] is working in. (Inherited from InstanceCollection.T. [▶ 3104].)
	Statics [▶ 2807]	Gets the Static Members

Methods

	Name	Description
	Add [▶ 2811]	Adds the specified item. (Overrides InstanceCollection.T..Add(T) [▶ 3112].)
	AddRange [▶ 3113]	Adds the specified items to this collection. (Inherited from InstanceCollection.T. [▶ 3104].)
	AsReadOnly [▶ 2809]	Returns a read only copy of this collection (shallow copy)
	CalcSize [▶ 2811]	Calculates the Byte Size of the IMemberCollection [▶ 2565]
	Clear [▶ 3114]	Clears this instance. (Inherited from InstanceCollection.T. [▶ 3104].)



	Name	Description
	Clone [▶ 2809]	Clones this MemberCollection.
	Contains(T) [▶ 3115]	Determines whether this collection contains the specified Instance [▶ 2549] (Inherited from InstanceCollection.T. [▶ 3104].)
	Contains(String) [▶ 3114]	Determines whether this collection contains an Instance [▶ 2549] with the specified InstanceName / InstancePath (Inherited from InstanceCollection.T. [▶ 3104].)
	ContainsName [▶ 3116]	Determines whether the specified instance name contains name. (Inherited from InstanceCollection.T. [▶ 3104].)
	CopyTo [▶ 3116]	Copies this InstanceCollection.T. [▶ 3104] to the specified array. (Inherited from InstanceCollection.T. [▶ 3104].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 3117]	Gets the enumerator. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetInstance [▶ 3117]	Gets the Instance [▶ 2549] by instance path. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetInstanceByName [▶ 3118]	Gets the name of the instance by. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 3119]	Determines the index of the specified Instance [▶ 2549]. (Inherited from InstanceCollection.T. [▶ 3104].)
	Insert [▶ 3119]	Inserts the specified Instance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 3120]	Removes the specified item. (Inherited from InstanceCollection.T. [▶ 3104].)
	RemoveAt [▶ 3121]	Removes the Instance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInstance [▶ 3121]	Tries to get the Instance [▶ 2549] of the specified path. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetInstanceByName [▶ 3122]	Tries to get Instances by name. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetMember [▶ 2810]	Tries to get the specified member

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.107.1 MemberCollection Constructor

Overload List

	Name	Description
	MemberCollection. [▶ 2805]	Initializes a new instance of the MemberCollection [▶ 2802] class.
	MemberCollection(I Enumerable.IMemb er.) [▶ 2805]	Initializes a new instance of the MemberCollection [▶ 2802] class (copy constructor)

Reference

[MemberCollection Class](#) [[▶ 2802](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.107.1.1 MemberCollection Constructor

Initializes a new instance of the [MemberCollection](#) [[▶ 2802](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public MemberCollection()
```

Reference

[MemberCollection Class](#) [[▶ 2802](#)]

[MemberCollection Overload](#) [[▶ 2805](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.107.1.2 MemberCollection Constructor (IEnumerable.IMember.)

Initializes a new instance of the [MemberCollection](#) [[▶ 2802](#)] class (copy constructor)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public MemberCollection(
    IEnumerable<IMember> coll
)
```

Parameters

coll	Type: System.Collections.Generic.IEnumerable.IMember [▶ 2561]. The coll.
------	---

Reference

[MemberCollection Class \[▶ 2802\]](#)











[MemberCollection Overload \[▶ 2805\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.107.2 MemberCollection Properties

The [MemberCollection \[▶ 2802\]](#) type exposes the following members.

Properties

	Name	Description
	Count [▶ 3107]	Gets the collection count. (Inherited from InstanceCollection.T. [▶ 3104].)
	Empty [▶ 2806]	Returns an Empty Member Collection.
	InnerList [▶ 3108]	Gets the Llist of instances. (Inherited from InstanceCollection.T. [▶ 3104].)
	InnerPathDict [▶ 3108]	The Path dictionary (Inherited from InstanceCollection.T. [▶ 3104].)
	Instances [▶ 2807]	Gets the Instance members (non static)
	IsReadOnly [▶ 3109]	Gets a value indicating whether this instance is read only. (Inherited from InstanceCollection.T. [▶ 3104].)
	Item.Int32. [▶ 3109]	Gets or sets the IInstance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	Item.String. [▶ 3110]	Gets the IInstance [▶ 2549] with the specified instance path. (Inherited from InstanceCollection.T. [▶ 3104].)
	Mode [▶ 3111]	The mode this InstanceCollection.T. [▶ 3104] is working in. (Inherited from InstanceCollection.T. [▶ 3104].)
	Statics [▶ 2807]	Gets the Static Members

Reference

[MemberCollection Class \[▶ 2802\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.107.2.1 MemberCollection.Empty Property

Returns an Empty Member Collection.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static MemberCollection Empty { get; }
```

Return Value

Type: [MemberCollection](#) [[▶ 2802](#)]
MemberCollection.

Reference

[MemberCollection Class](#) [[▶ 2802](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.107.2.2 MemberCollection.Instances Property

Gets the Instance members (non static)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IInstanceCollection<IMember> Instances { get; }
```

Property Value

Type: [IInstanceCollection](#) [[▶ 2554](#)].[IMember](#) [[▶ 2561](#)].
The instances.

Implements

[IMemberCollection.Instances](#) [[▶ 2567](#)]

Reference

[MemberCollection Class](#) [[▶ 2802](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.107.2.3 MemberCollection.Statics Property

Gets the Static Members

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IInstanceCollection<IMember> Statics { get; }
```

Property Value

Type: [IInstanceCollection](#) [[▶ 2554](#)].[IMember](#) [[▶ 2561](#)].
The statics.

Implements

[IMemberCollection.Statics](#) [[▶ 2567](#)]

Reference

[MemberCollection Class \[▶ 2802\]](#)






[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.107.3 MemberCollection Methods

The [MemberCollection \[▶ 2802\]](#) type exposes the following members.

Methods

	Name	Description
	Add [▶ 2811]	Adds the specified item. (Overrides InstanceCollection.T..Add(T) [▶ 3112].)
	AddRange [▶ 3113]	Adds the specified items to this collection. (Inherited from InstanceCollection.T. [▶ 3104].)
	AsReadOnly [▶ 2809]	Returns a read only copy of this collection (shallow copy)
	CalcSize [▶ 2811]	Calculates the Byte Size of the IMemberCollection [▶ 2565]
	Clear [▶ 3114]	Clears this instance. (Inherited from InstanceCollection.T. [▶ 3104].)
	Clone [▶ 2809]	Clones this MemberCollection [▶ 2802] .
	Contains(T) [▶ 3115]	Determines whether this collection contains the specified Instance [▶ 2549] (Inherited from InstanceCollection.T. [▶ 3104].)
	Contains(String) [▶ 3114]	Determines whether this collection contains an Instance [▶ 2549] with the specified InstanceName / InstancePath (Inherited from InstanceCollection.T. [▶ 3104].)
	ContainsName [▶ 3116]	Determines whether the specified instance name contains name. (Inherited from InstanceCollection.T. [▶ 3104].)
	CopyTo [▶ 3116]	Copies this InstanceCollection.T. [▶ 3104] to the specified array. (Inherited from InstanceCollection.T. [▶ 3104].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 3117]	Gets the enumerator. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetInstance [▶ 3117]	Gets the Instance [▶ 2549] by instance path. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetInstanceByName [▶ 3118]	Gets the name of the instance by. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 3119]	Determines the index of the specified Instance [▶ 2549] . (Inherited from InstanceCollection.T. [▶ 3104].)
	Insert [▶ 3119]	Inserts the specified Instance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 3120]	Removes the specified item. (Inherited from InstanceCollection.T. [▶ 3104].)

	Name	Description
	RemoveAt [▶ 3121]	Removes the Instance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInstance [▶ 3121]	Tries to get the Instance [▶ 2549] . of the specified path. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetInstanceByName [▶ 3122]	Tries to get Instances by name. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetMember [▶ 2810]	Tries to get the specified member

Reference

[MemberCollection Class \[▶ 2802\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.107.3.1 MemberCollection.AsReadOnly Method

Returns a read only copy of this collection (shallow copy)

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyMemberCollection AsReadOnly()
```

Return Value

Type: [ReadOnlyMemberCollection \[▶ 2845\]](#)

The readonly copy.

Reference

[MemberCollection Class \[▶ 2802\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.107.3.2 MemberCollection.Clone Method

Clones this [MemberCollection \[▶ 2802\]](#).

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public MemberCollection Clone()
```

Return Value

Type: [MemberCollection \[▶ 2802\]](#)

A cloned [MemberCollection \[▶ 2802\]](#).

Reference[MemberCollection Class \[▸ 2802\]](#)[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)**6.11.107.3.3 MemberCollection.Empty Method**

Returns an Empty Member Collection.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax**C#**

```
public static MemberCollection Empty()
```

Return ValueType: [MemberCollection \[▸ 2802\]](#)

MemberCollection.

Reference[MemberCollection Class \[▸ 2802\]](#)[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)**6.11.107.3.4 MemberCollection.TryGetMember Method**

Tries to get the specified member

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public bool TryGetMember(
    string fieldName,
    out IMember?? symbol
)
```

Parameters

fieldName	Type: System.String Name of the member.
symbol	Type: TwinCAT.TypeSystem.IMember [▸ 2561] . The symbol.

Return Value

Type: Boolean

true if found, false otherwise.

Implements[IMemberCollection.TryGetMember\(String, IMember.\) \[▸ 2569\]](#)

Reference

[MemberCollection Class](#) [► 2802]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.107.3.5 MemberCollection.Add Method

Adds the specified item.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override void Add(  
    IMember item  
)
```

Parameters

item	Type: TwinCAT.TypeSystem.IMember [► 2561] The item.
------	--

Implements

[ICollection.T..Add\(T\)](#)

[ICollection.T..Add\(T\)](#)

Reference

[MemberCollection Class](#) [► 2802]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.107.3.6 MemberCollection.CalcSize Method

Calculates the Byte Size of the [IMemberCollection](#) [► 2565]

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int CalcSize()
```

Return Value

Type: Int32

System.Int32.

Implements

[IMemberCollection.CalcSize.](#) [► 2569]

Remarks

This takes only the instance fields/members into account.

Reference

[MemberCollection Class \[▸ 2802\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.108 MethodParamFlags Enumeration

Flag set specifying the MethodParameter context

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
[FlagsAttribute]
public enum MethodParamFlags
```

Members

	Member name	Value	Description
	In	1	Input Parameter (ADSMETHODPARAFLAG_IN)
	Out	2	Output Parameter (ADSMETHODPARAFLAG_OUT)
	ByReference	4	By reference Parameter (ADSMETHODPARAFLAG_BYREFERENCE)

Reference

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.109 PrimitiveTypeFlags Enumeration

Enum PrimitiveTypeFlags

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
[FlagsAttribute]
public enum PrimitiveTypeFlags
```

Members

	Member name	Value	Description
	None	0	None / Uninitialized
	System	1	System Type like Byte / Word / DWORD
	Unsigned	2	Primitive Type is Unsigned

	Member name	Value	Description
	Bool	4	Boolean Value (maps to true and false)
	Float	8	Floating Point
	Date	16	Type represents a Date
	Time	32	Type represents a Time
	Numeric	64	Numeric value
	Bitset	128	Bitset
	MaskNumericUnsigned	66	Numeric / Unsigned Mask
	MaskDateTime	48	Date / Time Mask
	MaskSpecialType	67	Special type Mask

Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.110 RawValueChangedEventArgs Class

Event args for the [RawValueChanged \[▶ 2774\]](#) event.

Inheritance Hierarchy

System.Object
 System.EventArgs
 [TwinCAT.TypeSystem.ValueChangedBaseEventArgs \[▶ 2968\]](#)
 TwinCAT.TypeSystem.RawValueChangedEventArgs

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229




Syntax

C#




```
public class RawValueChangedEventArgs : ValueChangedBaseEventArgs
```




The RawValueChangedEventArgs type exposes the following members.

Properties

	Name	Description
	DateTime [▶ 2970]	Notification timestamp (Inherited from ValueChangedBaseEventArgs [▶ 2968] .)
	Symbol [▶ 2970]	Gets the symbol. (Inherited from ValueChangedBaseEventArgs [▶ 2968] .)
	Value [▶ 2814]	New Value (byte[])

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)

	Name	Description
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)




Reference

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.110.1 RawValueChangedEventArgs Properties

The [RawValueChangedEventArgs \[► 2813\]](#) type exposes the following members.

Properties

	Name	Description
	DateTime [► 2970]	Notification timestamp (Inherited from ValueChangedBaseEventArgs [► 2968].)
	Symbol [► 2970]	Gets the symbol. (Inherited from ValueChangedBaseEventArgs [► 2968].)
	Value [► 2814]	New Value (byte[])

Reference

[RawValueChangedEventArgs Class \[► 2813\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.110.1.1 RawValueChangedEventArgs.Value Property

New Value (byte[])

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyMemory<byte> Value { get; }
```

Property Value

Type: `ReadOnlyMemory.Byte`.

Reference







[RawValueChangedEventArgs Class \[► 2813\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.110.2 RawValueChangedEventArgs Methods

The [RawValueChangedEventArgs \[► 2813\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[RawValueChangedEventArgs Class \[▶ 2813\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.111 ReadOnlyDataTypeCollection Class

ReadOnly Collection of [IDataType \[▶ 2475\]](#) objects.

Inheritance Hierarchy

System.Object

System.Collections.ObjectModel.ReadOnlyCollection.[IDataType \[▶ 2475\]](#).

[TwinCAT.TypeSystem.Generic.ReadOnlyDataTypeCollection \[▶ 3139\].IDataType \[▶ 2475\]](#).

TwinCAT.TypeSystem.ReadOnlyDataTypeCollection

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229


Syntax

C#



```
public class ReadOnlyDataTypeCollection : ReadOnlyDataTypeCollection<IDataType>,
    IDataValueCollection, IDataValueCollection<IDataType>, IList<IDataType>,
    ICollection<IDataType>, IEnumerable<IDataType>, IEnumerable
```




The ReadOnlyDataTypeCollection type exposes the following members.

Constructors













	Name	Description
	ReadOnlyDataTypeCollection [▶ 2816]	Initializes a new instance of the ReadOnlyDataTypeCollection class.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.IDataType [▶ 2475] ..)
	Empty [▶ 2817]	Gets the empty collection.

	Name	Description
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.IDataType [▸ 2475]..)
	Item.String. [▸ 3143]	Gets the element with the specified type name. (Inherited from ReadOnlyDataTypeCollection.T. [▸ 3139]..)
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.IDataType [▸ 2475]..)

Methods

	Name	Description
	Contains	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IDataType [▸ 2475]..)
	ContainsType [▸ 3144]	Determines whether the specified name contains type. (Inherited from ReadOnlyDataTypeCollection.T. [▸ 3139]..)
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.IDataType [▸ 2475]..)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IDataType [▸ 2475]..)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IDataType [▸ 2475]..)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetType [▸ 3145]	Tries to get the Type with the specified name out of the collection. (Inherited from ReadOnlyDataTypeCollection.T. [▸ 3139]..)

Reference

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.111.1 ReadonlyDataTypeCollection Constructor

Initializes a new instance of the [ReadOnlyDataTypeCollection \[▸ 2815\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadonlyDataTypeCollection(
    DataTypeCollection<IDataType> coll
)
```


Parameters

coll	Type: TwinCAT.TypeSystem.Generic.DataTypeCollection [▶ 3087].IDataType [▶ 2475] . Collection of types.
------	---

Reference






[ReadOnlyDataTypeCollection Class \[▶ 2815\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.111.2 ReadOnlyDataTypeCollection Properties

The [ReadOnlyDataTypeCollection \[▶ 2815\]](#) type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.IDataType [▶ 2475] ..)
	Empty [▶ 2817]	Gets the empty collection.
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.IDataType [▶ 2475] ..)
	Item.String_ [▶ 3143]	Gets the element with the specified type name. (Inherited from ReadOnlyDataTypeCollection.T. [▶ 3139] .)
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.IDataType [▶ 2475] ..)

Reference

[ReadOnlyDataTypeCollection Class \[▶ 2815\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.111.2.1 ReadOnlyDataTypeCollection.Empty Property

Gets the empty collection.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static ReadOnlyDataTypeCollection Empty { get; }
```

Property Value

Type: [ReadOnlyDataTypeCollection \[▶ 2815\]](#)

The empty.

Reference













[ReadOnlyDataTypeCollection Class \[▶ 2815\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.111.3 ReadOnlyDataTypeCollection Methods

The [ReadOnlyDataTypeCollection \[► 2815\]](#) type exposes the following members.

Methods

	Name	Description
	Contains	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IDataType [► 2475]..)
	ContainsType [► 3144]	Determines whether the specified name contains type. (Inherited from ReadOnlyDataTypeCollection.T. [► 3139]..)
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.IDataType [► 2475]..)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IDataType [► 2475]..)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IDataType [► 2475]..)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetType [► 3145]	Tries to get the Type with the specified name out of the collection. (Inherited from ReadOnlyDataTypeCollection.T. [► 3139]..)

Reference

[ReadOnlyDataTypeCollection Class \[► 2815\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.112 ReadOnlyDimensionCollection Class

ReadOnly version of the [DimensionCollection \[► 2133\]](#)

Inheritance Hierarchy

System.Object

System.Collections.ObjectModel.ReadOnlyCollection.IDimension [► 2490].

TwinCAT.TypeSystem.ReadOnlyDimensionCollection

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229









Syntax

C#










```
public class ReadOnlyDimensionCollection : ReadOnlyCollection<IDimension>,
    IDimensionCollection, IList<IDimension>, ICollection<IDimension>,
    IEnumerable<IDimension>, IEnumerable
```



The ReadOnlyDimensionCollection type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.IDimension [▶ 2490]..)
	ElementCount [▶ 2820]	Gets the Number of elements in all Dimensions
	Empty [▶ 2822]	Returns an empty ReadOnlyDimensionCollection
	IsNonZeroBased [▶ 2822]	Gets a value indicating whether this instance is non zero index based.
	Item	Gets the element at the specified index. (Inherited from ReadOnlyCollection.IDimension [▶ 2490]..)
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.IDimension [▶ 2490]..)
	LowerBounds [▶ 2821]	Gets the lower bounds.
	UpperBounds [▶ 2821]	Gets the upper bounds.

Methods

	Name	Description
	Contains	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IDimension [▶ 2490]..)
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.IDimension [▶ 2490]..)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetDimensionLengths [▶ 2824]	Gets an array the specifies the Lengths of each Array Dimension
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IDimension [▶ 2490]..)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IDimension [▶ 2490]..)

	Name	Description
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Extension Methods

	Name	Description
	AddDimension [▶ 2982]	Adds a Dimension (FluentInterface) (Defined by IDimensionCollectionExtension [▶ 2981].)









Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.112.1 ReadOnlyDimensionCollection Properties

The [ReadOnlyDimensionCollection](#) [▶ 2818] type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.IDimension [▶ 2490].)
	ElementCount [▶ 2820]	Gets the Number of elements in all Dimensions
	Empty [▶ 2822]	Returns an empty ReadOnlyDimensionCollection [▶ 2818]
	IsNonZeroBased [▶ 2822]	Gets a value indicating whether this instance is non zero index based.
	Item	Gets the element at the specified index. (Inherited from ReadOnlyCollection.IDimension [▶ 2490].)
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.IDimension [▶ 2490].)
	LowerBounds [▶ 2821]	Gets the lower bounds.
	UpperBounds [▶ 2821]	Gets the upper bounds.

Reference

[ReadOnlyDimensionCollection Class](#) [▶ 2818]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.112.1.1 ReadOnlyDimensionCollection.ElementCount Property

Gets the Number of elements in all Dimensions

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ElementCount { get; }
```

Property Value

Type: Int32

Implements

[IDimensionCollection.ElementCount](#) [► 2494]

Reference

[ReadOnlyDimensionCollection Class](#) [► 2818]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.112.1.2 **ReadOnlyDimensionCollection.LowerBounds Property**

Gets the lower bounds.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int[] LowerBounds { get; }
```

Property Value

Type: .Int32.

The lower bounds.

Implements

[IDimensionCollection.LowerBounds](#) [► 2494]

Reference

[ReadOnlyDimensionCollection Class](#) [► 2818]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.112.1.3 **ReadOnlyDimensionCollection.UpperBounds Property**

Gets the upper bounds.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int[] UpperBounds { get; }
```

Property Value

Type: `.Int32`.
The upper bounds.

Implements

[IDimensionCollection.UpperBounds](#) [[▶ 2494](#)]

Reference

[ReadOnlyDimensionCollection Class](#) [[▶ 2818](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.112.1.4 **ReadOnlyDimensionCollection.Empty Property**

Returns an empty [ReadOnlyDimensionCollection](#) [[▶ 2818](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ReadOnlyDimensionCollection Empty { get; }
```

Return Value

Type: [ReadOnlyDimensionCollection](#) [[▶ 2818](#)]
`ReadOnlyDimensionCollection`.

Reference

[ReadOnlyDimensionCollection Class](#) [[▶ 2818](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.112.1.5 **ReadOnlyDimensionCollection.IsNonZeroBased Property**

Gets a value indicating whether this instance is non zero index based.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsNonZeroBased { get; }
```

Property Value

Type: `Boolean`
true if this instance is non zero indexed based; otherwise, false.

Implements

[IDimensionCollection.IsNonZeroBased](#) [[▶ 2495](#)]

Reference












[ReadOnlyDimensionCollection Class \[▶ 2818\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.112.2 ReadOnlyDimensionCollection Methods

The [ReadOnlyDimensionCollection \[▶ 2818\]](#) type exposes the following members.

Methods

	Name	Description
	Contains	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IDimension [▶ 2490].)
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.IDimension [▶ 2490].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetDimensionLengths [▶ 2824]	Gets an array the specifies the Lengths of each Array Dimension
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IDimension [▶ 2490].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IDimension [▶ 2490].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Extension Methods

	Name	Description
	AddDimension [▶ 2982]	Adds a Dimension (FluentInterface) (Defined by IDimensionCollectionExtension [▶ 2981].)

Reference

[ReadOnlyDimensionCollection Class \[▶ 2818\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.112.2.1 ReadOnlyDimensionCollection.Empty Method

Returns an empty [ReadOnlyDimensionCollection \[▶ 2818\]](#)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public static ReadOnlyDimensionCollection Empty()
```

Return Value

Type: [ReadOnlyDimensionCollection](#) [[▶ 2818](#)]

ReadOnlyDimensionCollection.

Reference

[ReadOnlyDimensionCollection Class](#) [[▶ 2818](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.112.2 ReadOnlyDimensionCollection.GetDimensionLengths Method

Gets an array the specifies the Lengths of each Array Dimension

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int[] GetDimensionLengths()
```

Return Value

Type: [.Int32](#).
[System.Int32](#)[].

Implements

[IDimensionCollection.GetDimensionLengths](#). [[▶ 2496](#)]

Reference

[ReadOnlyDimensionCollection Class](#) [[▶ 2818](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.113 ReadOnlyEnumValueCollection Class

Read only version of the [EnumValueCollection.T](#). [[▶ 2407](#)]

Inheritance Hierarchy

System.Object

System.Collections.ObjectModel.ReadOnlyCollection.[IEnumValue](#) [[▶ 2524](#)].

TwinCAT.TypeSystem.ReadOnlyEnumValueCollection

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229


Syntax

C#





```
public class ReadOnlyEnumValueCollection : ReadOnlyCollection<IEnumValue>,
    IEnumValueCollection, IEnumValueCollection<IEnumValue, IConvertible>, ICollection<IEnumValue>,
    IEnumerable<IEnumValue>, IEnumerable
```

The ReadOnlyEnumValueCollection type exposes the following members.













Constructors





	Name	Description
	ReadOnlyEnumValueCollection [▶ 2826]	Initializes a new instance of the ReadOnlyEnumValueCollection.T. [▶ 2833] class.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.IEnumValue [▶ 2524]..)
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.IEnumValue [▶ 2524]..)
	Item.String. [▶ 2827]	Gets or sets the element at the specified index.
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.IEnumValue [▶ 2524]..)

Methods

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IEnumValue [▶ 2524]..)
	Contains(String) [▶ 2829]	Determines whether [contains] [the specified name].
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.IEnumValue [▶ 2524]..)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IEnumValue [▶ 2524]..)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetNames [▶ 2829]	Gets the Value Names.
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	GetValues [▶ 2830]	Gets the values.
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IEnumValue [▶ 2524]..)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)

	Name	Description
	Parse [▸ 2830]	Parses the specified name.
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryParse(String, IConvertible.) [▸ 2831]	Tries to pars the string value of the Enum.
	TryParse(String, EnumValue.) [▸ 2832]	Tries to pars the string value of the Enum.

Reference

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.113.1 ReadOnlyEnumValueCollection Constructor

Initializes a new instance of the [ReadOnlyEnumValueCollection.T.](#) [[▸ 2833](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyEnumValueCollection(
    EnumValueCollection coll
)
```

Parameters

coll	Type: TwinCAT.TypeSystem.EnumValueCollection [▸ 2391] The coll.
------	--

Reference





[ReadOnlyEnumValueCollection Class](#) [[▸ 2824](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.113.2 ReadOnlyEnumValueCollection Properties

The [ReadOnlyEnumValueCollection](#) [[▸ 2824](#)] type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.IEnumValue [▸ 2524]..)
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.IEnumValue [▸ 2524]..)
	Item.String. [▸ 2827]	Gets or sets the element at the specified index.
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.IEnumValue [▸ 2524]..)



Reference

[ReadOnlyEnumValueCollection Class \[▸ 2824\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.113.2.1 ReadOnlyEnumValueCollection.Item Property

Overload List

	Name	Description
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.IEnumValue [▸ 2524]..)
	Item.String. [▸ 2827]	Gets or sets the element at the specified index.

Reference

[ReadOnlyEnumValueCollection Class \[▸ 2824\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.113.2.1.1 ReadOnlyEnumValueCollection.Item Property (String)

Gets or sets the element at the specified index.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IConvertible this[
    string name
] { get; }
```

Parameters

name	Type: System.String The name of the value
------	--

Return Value

Type: IConvertible
EnumValue<T>.

Implements

[IEnumValueCollection.TEnumValue, TValue..Item.String. \[▸ 2530\]](#)

Exceptions

Exception	Condition
NotImplementedException	
NotImplementedException	

Reference

[ReadOnlyEnumValueCollection Class \[▸ 2824\]](#)

















[Item Overload \[▶ 2827\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.113.3 ReadOnlyEnumValueCollection Methods

The [ReadOnlyEnumValueCollection \[▶ 2824\]](#) type exposes the following members.

Methods

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IEnumValue [▶ 2524].)
	Contains(String) [▶ 2829]	Determines whether [contains] [the specified name].
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.IEnumValue [▶ 2524].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IEnumValue [▶ 2524].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetNames [▶ 2829]	Gets the Value Names.
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	GetValues [▶ 2830]	Gets the values.
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IEnumValue [▶ 2524].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Parse [▶ 2830]	Parses the specified name.
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryParse(String, IConvertible.) [▶ 2831]	Tries to pars the string value of the Enum.
	TryParse(String, IEnumValue.) [▶ 2832]	Tries to pars the string value of the Enum.



Reference

[ReadOnlyEnumValueCollection Class \[▶ 2824\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.113.3.1 ReadOnlyEnumValueCollection.Contains Method

Overload List

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IEnumValue [▶ 2524].)
	Contains(String) [▶ 2829]	Determines whether [contains] [the specified name].

Reference

[ReadOnlyEnumValueCollection Class \[▶ 2824\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.113.3.1.1 ReadOnlyEnumValueCollection.Contains Method (String)

Determines whether [contains] [the specified name].

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(
    string value
)
```

Parameters

value	Type: System.String Value
-------	------------------------------

Return Value

Type: Boolean

true if [contains] [the specified name]; otherwise, false.

Implements

[IEnumValueCollection.TEnumValue, TValue..Contains\(String\) \[▶ 2532\]](#)

Reference

[ReadOnlyEnumValueCollection Class \[▶ 2824\]](#)

[Contains Overload \[▶ 2829\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.113.3.2 ReadOnlyEnumValueCollection.GetNames Method

Gets the Value Names.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string[] GetNames()
```

Return Value

Type: `.String`.
`System.String[]`.

Implements

[IEnumerable.TEnumValue, TValue..GetNames.](#) [[▶ 2533](#)]

Reference

[ReadOnlyEnumValueCollection Class](#) [[▶ 2824](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.113.3.3 **ReadOnlyEnumValueCollection.GetValues Method**

Gets the values.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IConvertible[] GetValues()
```

Return Value

Type: `.IConvertible`.
`T[]`.

Implements

[IEnumerable.TEnumValue, TValue..GetValues.](#) [[▶ 2533](#)]

Reference

[ReadOnlyEnumValueCollection Class](#) [[▶ 2824](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.113.3.4 **ReadOnlyEnumValueCollection.Parse Method**

Parses the specified name.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IConvertible Parse(
    string name
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: IConvertible
T.

Implements

[IEnumValueCollection.TEnumValue, TValue..Parse\(String\) \[▸ 2533\]](#)



Reference

[ReadOnlyEnumValueCollection Class \[▸ 2824\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.113.3.5 ReadOnlyEnumValueCollection.TryParse Method

Overload List

	Name	Description
	TryParse(String, IConvertible.) [▸ 2831]	Tries to pars the string value of the Enum.
	TryParse(String, IEnumValue.) [▸ 2832]	Tries to pars the string value of the Enum.

Reference

[ReadOnlyEnumValueCollection Class \[▸ 2824\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.113.3.5.1 ReadOnlyEnumValueCollection.TryParse Method (String, IConvertible.)

Tries to pars the string value of the Enum.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryParse(
    string strValue,
    out IConvertible value
)
```

Parameters

strValue	Type: System.String The Value in string representation.
value	Type: System.IConvertible. The value.

Return Value

Type: Boolean
true if the value can be parsed, false otherwise.

Implements

[IEnumerable{TEnumValue, TValue}.TryParse\(String, TValue.\)](#) [[▶ 2535](#)]

Reference

[ReadOnlyEnumValueCollection Class](#) [[▶ 2824](#)]

[TryParse Overload](#) [[▶ 2831](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.113.3.5.2 ReadOnlyEnumValueCollection.TryParse Method (String, IEnumValue.)

Tries to pars the string value of the Enum.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryParse(
    string strValue,
    out IEnumValue?? value
)
```

Parameters

strValue	Type: System.String The Value in string representation.
value	Type: TwinCAT.TypeSystem.IEnumValue [▶ 2524]. The value.

Return Value

Type: Boolean
true if the value can be parsed, false otherwise.

Implements

[IEnumerable.TEnumerable, TValue..TryParse\(String, TValue.\) \[▶ 2535\]](#)

Reference

[ReadOnlyEnumValueCollection Class \[▶ 2824\]](#)

[TryParse Overload \[▶ 2831\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.114 ReadOnlyEnumValueCollection.T. Class

Read only version of the [EnumValueCollection.T. \[▶ 2407\]](#)

Inheritance Hierarchy

System.Object

System.Collections.ObjectModel.ReadOnlyCollection.EnumValue [▶ 2384].T..

TwinCAT.TypeSystem.ReadOnlyEnumValueCollection.T.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#


```
public class ReadOnlyEnumValueCollection<T> : ReadOnlyCollection<EnumValue<T>>,
    IEnumerable<EnumValue<T>, T>, ICollection<EnumValue<T>>, IEnumerable<EnumValue<T>>,
    IEnumerable
where T : struct, new(), IConvertible
```

Type Parameters





T

The ReadOnlyEnumValueCollection.T. type exposes the following members.

















Constructors

	Name	Description
	ReadOnlyEnumValueCollection.T. [▶ 2834]	Initializes a new instance of the ReadOnlyEnumValueCollection.T. class.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.EnumValue [▶ 2384].T...)
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.EnumValue [▶ 2384].T...)
	Item.String. [▶ 2835]	Gets the enumeration value T from its string representation.
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.EnumValue [▶ 2384].T...)

Methods

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.EnumValue [▶ 2384].T...)
	Contains(String) [▶ 2837]	Determines whether [contains] [the specified name].
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.EnumValue [▶ 2384].T...)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.EnumValue [▶ 2384].T...)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetNames [▶ 2838]	Gets the Value Names.
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	GetValues [▶ 2838]	Gets the values.
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.EnumValue [▶ 2384].T...)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Parse [▶ 2839]	Parses the specified name.
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryParse(String, T.) [▶ 2840]	Tries to parse the string value of the Enum.
	TryParse(String, EnumValue.T.) [▶ 2840]	Tries to parse the string value of the Enum.

Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.114.1 ReadOnlyEnumValueCollection.T. Constructor

Initializes a new instance of the [ReadOnlyEnumValueCollection.T. \[▶ 2833\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyEnumValueCollection(
    EnumValueCollection<T> coll
)
```

Parameters

coll	Type: TwinCAT.TypeSystem.EnumValueCollection [▶ 2407].T [▶ 2833]. The coll.
------	--

Reference





[ReadOnlyEnumValueCollection.T. Class](#) [[▶ 2833](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.114.2 ReadOnlyEnumValueCollection.T. Properties

The [ReadOnlyEnumValueCollection.T.](#) [[▶ 2833](#)] generic type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.EnumValue [▶ 2384].T [▶ 2833]...)
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.EnumValue [▶ 2384].T [▶ 2833]...)
	Item.String. [▶ 2835]	Gets the enumeration value T from its string representation.
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.EnumValue [▶ 2384].T [▶ 2833]...)



Reference

[ReadOnlyEnumValueCollection.T. Class](#) [[▶ 2833](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.114.2.1 ReadOnlyEnumValueCollection.T..Item Property

Overload List

	Name	Description
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.EnumValue [▶ 2384].T [▶ 2833]...)
	Item.String. [▶ 2835]	Gets the enumeration value T from its string representation.

Reference

[ReadOnlyEnumValueCollection.T. Class](#) [[▶ 2833](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.114.2.1.1 ReadOnlyEnumValueCollection.T..Item Property (String)

Gets the enumeration value T from its string representation.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T this[
    string name
] { get; }
```

Parameters

name	Type: System.String The name of the enum value.
------	--

Return Value

Type: [T](#) [[▶ 2833](#)]

T.

Implements

[IEnumerable.TEnumValue, TValue..Item.String.](#) [[▶ 2530](#)]

Reference

[ReadOnlyEnumValueCollection.T. Class](#) [[▶ 2833](#)]











[Item Overload](#) [[▶ 2835](#)]







[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.114.3 ReadOnlyEnumValueCollection.T. Methods

The [ReadOnlyEnumValueCollection.T.](#) [[▶ 2833](#)] generic type exposes the following members.

Methods

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.EnumValue [▶ 2384].T [▶ 2833]...)
	Contains(String) [▶ 2837]	Determines whether [contains] [the specified name].
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.EnumValue [▶ 2384].T [▶ 2833]...)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.EnumValue [▶ 2384].T [▶ 2833]...)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetNames [▶ 2838]	Gets the Value Names.
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	GetValues [▶ 2838]	Gets the values.

	Name	Description
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.EnumValue [▶ 2384].T [▶ 2833]...)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Parse [▶ 2839]	Parses the specified name.
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryParse(String, T.) [▶ 2840]	Tries to parse the string value of the Enum.
	TryParse(String, EnumValue.T.) [▶ 2840]	Tries to parse the string value of the Enum.



Reference

[ReadOnlyEnumValueCollection.T. Class \[▶ 2833\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.114.3.1 ReadOnlyEnumValueCollection.T..Contains Method

Overload List

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.EnumValue [▶ 2384].T [▶ 2833]...)
	Contains(String) [▶ 2837]	Determines whether [contains] [the specified name].

Reference

[ReadOnlyEnumValueCollection.T. Class \[▶ 2833\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.114.3.1.1 ReadOnlyEnumValueCollection.T..Contains Method (String)

Determines whether [contains] [the specified name].

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(
    string value
)
```

Parameters

value	Type: System.String Value
-------	------------------------------

Return Value

Type: Boolean
true if [contains] [the specified name]; otherwise, false.

Implements

[IEnumerable.TEnumValue, TValue..Contains\(String\) \[► 2532\]](#)

Reference

[ReadOnlyEnumValueCollection.T. Class \[► 2833\]](#)

[Contains Overload \[► 2837\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.114.3.2 ReadOnlyEnumValueCollection.T..GetNames Method

Gets the Value Names.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string[] GetNames()
```

Return Value

Type: .String.
System.String[].

Implements

[IEnumerable.TEnumValue, TValue..GetNames. \[► 2533\]](#)

Reference

[ReadOnlyEnumValueCollection.T. Class \[► 2833\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.114.3.3 ReadOnlyEnumValueCollection.T..GetValues Method

Gets the values.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T[] GetValues()
```

Return Value

Type: [T](#) [[▶ 2833](#)].
[T\[\]](#).

Implements

[IEnumerable.TEnumValue, TValue..GetValues.](#) [[▶ 2533](#)]

Reference

[ReadOnlyEnumValueCollection.T. Class](#) [[▶ 2833](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.114.3.4 ReadOnlyEnumValueCollection.T..Parse Method

Parses the specified name.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public T Parse(
    string name
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: [T](#) [[▶ 2833](#)]
[T](#).

Implements


[IEnumerable.TEnumValue, TValue..Parse\(String\)](#) [[▶ 2533](#)]


Reference

[ReadOnlyEnumValueCollection.T. Class](#) [[▶ 2833](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.114.3.5 ReadOnlyEnumValueCollection.T..TryParse Method**Overload List**

	Name	Description
	TryParse(String, T.) [▶ 2840]	Tries to parse the string value of the Enum.

	Name	Description
	TryParse(String, EnumValue.T.) [▶ 2840]	Tries to parse the string value of the Enum.

Reference

[ReadOnlyEnumValueCollection.T. Class](#) [▶ 2833]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.114.3.5.1 ReadOnlyEnumValueCollection.T..TryParse Method (String, EnumValue.T..)

Tries to parse the string value of the Enum.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryParse(
    string strValue,
    out EnumValue<T>? value
)
```

Parameters

strValue	Type: System.String The Value in string representation.
value	Type: TwinCAT.TypeSystem.EnumValue [▶ 2384].T [▶ 2833].. The value.

Return Value

Type: Boolean
true if the enum value can be parsed, false otherwise.

Implements

[IEnumValueCollection.TEnumValue, TValue..TryParse\(String, TValue.\)](#) [▶ 2535]

Reference

[ReadOnlyEnumValueCollection.T. Class](#) [▶ 2833]

[TryParse Overload](#) [▶ 2839]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.114.3.5.2 ReadOnlyEnumValueCollection.T..TryParse Method (String, T.)

Tries to parse the string value of the Enum.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryParse(
    string strValue,
    out T value
)
```

Parameters

strValue	Type: System.String The Value in string representation.
value	Type: T [▶ 2833]. The value.

Return Value

Type: Boolean
true if the enum value can be parsed, false otherwise.

Implements

[IEnumValueCollection.TEnumValue, TValue..TryParse\(String, TValue.\)](#) [[▶ 2535](#)]

Reference

[ReadOnlyEnumValueCollection.T. Class](#) [[▶ 2833](#)]

[TryParse Overload](#) [[▶ 2839](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.115 ReadOnlyFieldCollection Class

Read only collection of [IField](#) [[▶ 2535](#)] objects

Inheritance Hierarchy

System.Object

System.Collections.ObjectModel.ReadOnlyCollection.[IField](#) [[▶ 2535](#)].

[TwinCAT.TypeSystem.Generic.ReadOnlyInstanceCollection](#) [[▶ 3145](#)].[IField](#) [[▶ 2535](#)].

[TwinCAT.TypeSystem.ReadOnlyFieldCollection](#)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229


Syntax

C#






```
public class ReadOnlyFieldCollection : ReadOnlyInstanceCollection<IField>,
    IFieldCollection, IInstanceCollection<IField>, IList<IField>,
    ICollection<IField>, IEnumerable<IField>, IEnumerable
```

The [ReadOnlyFieldCollection](#) type exposes the following members.














Constructors





	Name	Description
	ReadOnlyFieldCollection [▶ 2843]	Initializes a new instance of the ReadOnlyMemberCollection [▶ 2845] class.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.IField [▶ 2535]..)
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.IField [▶ 2535]..)
	Item.String. [▶ 3148]	Gets the element with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145]..)
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.IField [▶ 2535]..)
	Mode [▶ 3149]	Gets the InstanceCollectionMode [▶ 2573]. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145]..)

Methods

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T. (Inherited from ReadOnlyCollection.IField [▶ 2535]..)
	Contains(String) [▶ 3150]	Determines whether the ReadOnlyInstanceCollection.T. [▶ 3145] contains an instance with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145]..)
	ContainsName [▶ 3151]	Determines whether the specified instance is contained. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145]..)
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.IField [▶ 2535]..)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T. (Inherited from ReadOnlyCollection.IField [▶ 2535]..)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetInstance [▶ 3152]	Gets the Instance [▶ 2549] by instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145]..)
	GetInstanceByName [▶ 3152]	Gets the Instance [▶ 2549] by instance name. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145]..)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T. (Inherited from ReadOnlyCollection.IField [▶ 2535]..)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)

	Name	Description
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInstance [▶ 3153]	Tries to get the instance with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145] .)
	TryGetInstanceByName [▶ 3153]	Tries to get the instance by name. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145] .)
	TryGetMember [▶ 2845]	Tries to get the specified member

Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.115.1 ReadOnlyFieldCollection Constructor

Initializes a new instance of the [ReadOnlyMemberCollection \[▶ 2845\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyFieldCollection(
    FieldCollection members
)
```

Parameters

members	Type: TwinCAT.TypeSystem.FieldCollection [▶ 2427] The members.
---------	---

Reference





[ReadOnlyFieldCollection Class \[▶ 2841\]](#)


[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.115.2 ReadOnlyFieldCollection Properties

The [ReadOnlyFieldCollection \[▶ 2841\]](#) type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.IField [▶ 2535] ..)
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.IField [▶ 2535] ..)
	Item.String. [▶ 3148]	Gets the element with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145] .)
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.IField [▶ 2535] ..)

	Name	Description
	Mode [▶ 3149]	Gets the InstanceCollectionMode [▶ 2573]. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)

Reference

















[ReadOnlyFieldCollection Class](#) [[▶ 2841](#)]


[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.115.3 ReadOnlyFieldCollection Methods

The [ReadOnlyFieldCollection](#) [[▶ 2841](#)] type exposes the following members.

Methods

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IField [▶ 2535]..)
	Contains(String) [▶ 3150]	Determines whether the ReadOnlyInstanceCollection.T. [▶ 3145] contains an instance with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	ContainsName [▶ 3151]	Determines whether the specified instance is contained. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.IField [▶ 2535]..)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IField [▶ 2535]..)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetInstance [▶ 3152]	Gets the Instance [▶ 2549] by instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	GetInstanceByName [▶ 3152]	Gets the Instance [▶ 2549] by instance name. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IField [▶ 2535]..)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInstance [▶ 3153]	Tries to get the instance with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	TryGetInstanceByName [▶ 3153]	Tries to get the instance by name. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)

	Name	Description
	TryGetMember [▶ 2845]	Tries to get the specified member

Reference

[ReadOnlyFieldCollection Class](#) [▶ 2841]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.115.3.1 ReadOnlyFieldCollection.TryGetMember Method

Tries to get the specified member

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetMember(
    string fieldName,
    out IField?? symbol
)
```

Parameters

fieldName	Type: System.String Name of the member.
symbol	Type: TwinCAT.TypeSystem.IField [▶ 2535]. The symbol.

Return Value

Type: Boolean
true if found, false otherwise.

Implements

[IFieldCollection.TryGetMember\(String, IField.\)](#) [▶ 2541]

Reference

[ReadOnlyFieldCollection Class](#) [▶ 2841]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.116 ReadOnlyMemberCollection Class

Read only collection of [IMember](#) [▶ 2561] objects

Inheritance Hierarchy

System.Object

System.Collections.ObjectModel.ReadOnlyCollection.[IMember](#) [▶ 2561].

[TwinCAT.TypeSystem.Generic.ReadOnlyInstanceCollection](#) [▶ 3145].[IMember](#) [▶ 2561].

TwinCAT.TypeSystem.ReadOnlyMemberCollection

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229


Syntax

C#









```
public class ReadOnlyMemberCollection : ReadOnlyInstanceCollection<IMember>,
    IMemberCollection, IInstanceCollection<IMember>, IList<IMember>,
    ICollection<IMember>, IEnumerable<IMember>, IEnumerable
```

The `ReadOnlyMemberCollection` type exposes the following members.






Constructors














	Name	Description
	ReadOnlyMemberCollection [▶ 2847]	Initializes a new instance of the <code>ReadOnlyMemberCollection</code> class.

Properties

	Name	Description
	Count	Gets the number of elements contained in the <code>ReadOnlyCollection.T.</code> instance. (Inherited from ReadOnlyCollection.IMember [▶ 2561]..)
	Empty [▶ 2848]	Returns an Empty <code>ReadOnlyMemberCollection</code>
	Instances [▶ 2849]	Gets the Instance members (non static)
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.IMember [▶ 2561]..)
	Item.String. [▶ 3148]	Gets the element with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	Items	Returns the <code>IList.T.</code> that the <code>ReadOnlyCollection.T.</code> wraps. (Inherited from ReadOnlyCollection.IMember [▶ 2561]..)
	Mode [▶ 3149]	Gets the InstanceCollectionMode [▶ 2573]. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	Statics [▶ 2849]	Gets the Static Members

Methods

	Name	Description
	CalcSize [▶ 2852]	Calculates the Byte Size of the IMemberCollection [▶ 2565]
	Contains(T)	Determines whether an element is in the <code>ReadOnlyCollection.T.</code> (Inherited from ReadOnlyCollection.IMember [▶ 2561]..)
	Contains(String) [▶ 3150]	Determines whether the ReadOnlyInstanceCollection.T. [▶ 3145] contains an instance with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	ContainsName [▶ 3151]	Determines whether the specified instance is contained. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	CopyTo	Copies the entire <code>ReadOnlyCollection.T.</code> to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.IMember [▶ 2561]..)

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IMember [▶ 2561].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetInstance [▶ 3152]	Gets the Instance [▶ 2549] by instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145] .)
	GetInstanceByName [▶ 3152]	Gets the Instance [▶ 2549] by instance name. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145] .)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IMember [▶ 2561] ..)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInstance [▶ 3153]	Tries to get the instance with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145] .)
	TryGetInstanceByName [▶ 3153]	Tries to get the instance by name. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145] .)
	TryGetMember [▶ 2851]	Tries to get the specified member

Reference

[TwinCAT.TypeSystem Namespace \[\[▶ 2083\]\(#\)\]](#)

6.11.116.1 ReadOnlyMemberCollection Constructor

Initializes a new instance of the [ReadOnlyMemberCollection \[\[▶ 2845\]\(#\)\]](#) class.

Namespace: [TwinCAT.TypeSystem \[\[▶ 2083\]\(#\)\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyMemberCollection(
    MemberCollection members
)
```

Parameters

members	Type: TwinCAT.TypeSystem.MemberCollection [▶ 2802] The members.
---------	--

Reference









[ReadOnlyMemberCollection Class \[\[▶ 2845\]\(#\)\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.116.2 ReadOnlyMemberCollection Properties

The [ReadOnlyMemberCollection \[▶ 2845\]](#) type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.IMember [▶ 2561]..)
	Empty [▶ 2848]	Returns an Empty ReadOnlyMemberCollection [▶ 2845]
	Instances [▶ 2849]	Gets the Instance members (non static)
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.IMember [▶ 2561]..)
	Item.String. [▶ 3148]	Gets the element with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145]..)
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.IMember [▶ 2561]..)
	Mode [▶ 3149]	Gets the InstanceCollectionMode [▶ 2573] . (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145]..)
	Statics [▶ 2849]	Gets the Static Members

Reference

[ReadOnlyMemberCollection Class \[▶ 2845\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.116.2.1 ReadOnlyMemberCollection.Empty Property

Returns an Empty [ReadOnlyMemberCollection \[▶ 2845\]](#)

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ReadOnlyMemberCollection Empty { get; }
```

Return Value

Type: [ReadOnlyMemberCollection \[▶ 2845\]](#)

ReadOnlyMemberCollection.

Reference

[ReadOnlyMemberCollection Class \[▶ 2845\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.116.2.2 **ReadOnlyMemberCollection.Instances Property**

Gets the Instance members (non static)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IInstanceCollection<IMember> Instances { get; }
```

Property Value

Type: [IInstanceCollection](#) [[▶ 2554](#)].[IMember](#) [[▶ 2561](#)].

The instances.

Implements

[IMemberCollection.Instances](#) [[▶ 2567](#)]

Reference

[ReadOnlyMemberCollection Class](#) [[▶ 2845](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.116.2.3 **ReadOnlyMemberCollection.Statics Property**

Gets the Static Members

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IInstanceCollection<IMember> Statics { get; }
```

Property Value

Type: [IInstanceCollection](#) [[▶ 2554](#)].[IMember](#) [[▶ 2561](#)].

The statics.

Implements

[IMemberCollection.Statics](#) [[▶ 2567](#)]

Reference



















[ReadOnlyMemberCollection Class](#) [[▶ 2845](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.116.3 **ReadOnlyMemberCollection Methods**

The [ReadOnlyMemberCollection](#) [[▶ 2845](#)] type exposes the following members.

Methods

	Name	Description
	CalcSize [▶ 2852]	Calculates the Byte Size of the IMemberCollection [▶ 2565]
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IMember [▶ 2561]..)
	Contains(String) [▶ 3150]	Determines whether the ReadOnlyInstanceCollection.T. [▶ 3145] contains an instance with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	ContainsName [▶ 3151]	Determines whether the specified instance is contained. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.IMember [▶ 2561]..)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IMember [▶ 2561]..)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetInstance [▶ 3152]	Gets the Instance [▶ 2549] by instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	GetInstanceByName [▶ 3152]	Gets the Instance [▶ 2549] by instance name. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IMember [▶ 2561]..)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInstance [▶ 3153]	Tries to get the instance with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	TryGetInstanceByName [▶ 3153]	Tries to get the instance by name. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	TryGetMember [▶ 2851]	Tries to get the specified member

Reference

[ReadOnlyMemberCollection Class \[▶ 2845\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.116.3.1 **ReadOnlyMemberCollection.Empty Method**

Returns an Empty [ReadOnlyMemberCollection \[▶ 2845\]](#)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public static ReadOnlyMemberCollection Empty()
```

Return Value

Type: [ReadOnlyMemberCollection](#) [[▶ 2845](#)]

ReadOnlyMemberCollection.

Reference

[ReadOnlyMemberCollection Class](#) [[▶ 2845](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.116.3.2 ReadOnlyMemberCollection.TryGetMember Method

Tries to get the specified member

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetMember(  
    string memberName,  
    out IMember?? symbol  
)
```

Parameters

memberName	Type: System.String Name of the member.
symbol	Type: TwinCAT.TypeSystem.IMember [▶ 2561]. The symbol.

Return Value

Type: Boolean

true if found, false otherwise.

Implements

[IMemberCollection.TryGetMember\(String, IMember.\)](#) [[▶ 2569](#)]

Reference

[ReadOnlyMemberCollection Class](#) [[▶ 2845](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.116.3.3 ReadOnlyMemberCollection.CalcSize Method

Calculates the Byte Size of the [IMemberCollection](#) [[▶ 2565](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int CalcSize()
```

Return Value

Type: Int32

System.Int32.

Implements

[IMemberCollection.CalcSize.](#) [[▶ 2569](#)]

Remarks

This takes only the instance fields/members into account.

Reference

[ReadOnlyMemberCollection Class](#) [[▶ 2845](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.117 ReadOnlyMethodParameterCollection Class

Read only [RpcMethodParameterCollection](#) [[▶ 2915](#)].

Inheritance Hierarchy

System.Object

System.Collections.ObjectModel.ReadOnlyCollection.[IRpcMethodParameter](#) [[▶ 2635](#)].

TwinCAT.TypeSystem.ReadOnlyMethodParameterCollection

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229


Syntax



C#

```
public class ReadOnlyMethodParameterCollection : ReadOnlyCollection<IRpcMethodParameter>,
    IRpcMethodParameterCollection, IList<IRpcMethodParameter>, ICollection<IRpcMethodParameter>,
    IEnumerable<IRpcMethodParameter>, IEnumerable
```












The [ReadOnlyMethodParameterCollection](#) type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.IRpcMethodParameter [▶ 2635].)

	Name	Description
	Item	Gets the element at the specified index. (Inherited from ReadOnlyCollection.IRpcMethodParameter [▸ 2635]..)
	Items	Returns the <code>IList.T.</code> that the <code>ReadOnlyCollection.T.</code> wraps. (Inherited from ReadOnlyCollection.IRpcMethodParameter [▸ 2635]..)

Methods

	Name	Description
	Contains	Determines whether an element is in the <code>ReadOnlyCollection.T..</code> (Inherited from ReadOnlyCollection.IRpcMethodParameter [▸ 2635]..)
	CopyTo	Copies the entire <code>ReadOnlyCollection.T.</code> to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.IRpcMethodParameter [▸ 2635]..)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from <code>Object.</code>)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from <code>Object.</code>)
	GetEnumerator	Returns an enumerator that iterates through the <code>ReadOnlyCollection.T..</code> (Inherited from ReadOnlyCollection.IRpcMethodParameter [▸ 2635]..)
	GetHashCode	Serves as the default hash function. (Inherited from <code>Object.</code>)
	GetLengthsParameter [▸ 2854]	Gets the corresponding <code>Lengths</code> parameter.
	GetType	Gets the <code>Type</code> of the current instance. (Inherited from <code>Object.</code>)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire <code>ReadOnlyCollection.T..</code> (Inherited from ReadOnlyCollection.IRpcMethodParameter [▸ 2635]..)
	MemberwiseClone	Creates a shallow copy of the current <code>Object.</code> (Inherited from <code>Object.</code>)
	ToString	Returns a string that represents the current object. (Inherited from <code>Object.</code>)




Reference

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.117.1 ReadOnlyMethodParameterCollection Properties

The [ReadOnlyMethodParameterCollection \[▸ 2852\]](#) type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the <code>ReadOnlyCollection.T.</code> instance. (Inherited from ReadOnlyCollection.IRpcMethodParameter [▸ 2635]..)
	Item	Gets the element at the specified index. (Inherited from ReadOnlyCollection.IRpcMethodParameter [▸ 2635]..)
	Items	Returns the <code>IList.T.</code> that the <code>ReadOnlyCollection.T.</code> wraps. (Inherited from ReadOnlyCollection.IRpcMethodParameter [▸ 2635]..)

Reference












[ReadOnlyMethodParameterCollection Class \[► 2852\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.117.2 ReadOnlyMethodParameterCollection Methods

The [ReadOnlyMethodParameterCollection \[► 2852\]](#) type exposes the following members.

Methods

	Name	Description
	Contains	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IRpcMethodParameter [► 2635].)
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.IRpcMethodParameter [► 2635].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IRpcMethodParameter [► 2635].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetLengthsParameter [► 2854]	Gets the corresponding Lengths parameter.
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IRpcMethodParameter [► 2635].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[ReadOnlyMethodParameterCollection Class \[► 2852\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.117.2.1 ReadOnlyMethodParameterCollection.GetLengthsParameter Method

Gets the corresponding Lengths parameter.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IRpcMethodParameter? GetLengthIsParameter(
    IRpcMethodParameter parameter
)
```

Parameters

parameter	Type: TwinCAT.TypeSystem.IRpcMethodParameter [▶ 2635] The value parameter
-----------	--

Return Value

Type: [IRpcMethodParameter](#) [▶ 2635]
The LengthIs Parameter

Implements

[IRpcMethodParameterCollection.GetLengthIsParameter\(IRpcMethodParameter\)](#) [▶ 2641]

Reference

[ReadOnlyMethodParameterCollection Class](#) [▶ 2852]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

[IRpcMethodParameter.LengthIsParameterIndex](#) [▶ 2637]

[IRpcMethodParameter.HasLengthIsParameter](#) [▶ 2636]

6.11.118 ReadOnlyRpcMethodCollection Class

Read only [RpcMethodCollection](#) [▶ 2894]

Inheritance Hierarchy

System.Object

System.Collections.ObjectModel.ReadOnlyCollection.IRpcMethod [▶ 2625].

TwinCAT.TypeSystem.ReadOnlyRpcMethodCollection

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229



Syntax



C#

```
public class ReadOnlyRpcMethodCollection : ReadOnlyCollection<IRpcMethod>,
    IRpcMethodCollection, IList<IRpcMethod>, ICollection<IRpcMethod>,
    IEnumerable<IRpcMethod>, IEnumerable
```














The [ReadOnlyRpcMethodCollection](#) type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T instance. (Inherited from ReadOnlyCollection.IRpcMethod [▶ 2625]..)
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.IRpcMethod [▶ 2625]..)

	Name	Description
	Item.String [▶ 2857]	Gets the IRpcMethod [▶ 2625] with the specified method name.
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.IRpcMethod [▶ 2625]..)

Methods

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T. (Inherited from ReadOnlyCollection.IRpcMethod [▶ 2625]..)
	Contains(String) [▶ 2859]	Determines whether this collection contains the specified method name.
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.IRpcMethod [▶ 2625]..)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T. (Inherited from ReadOnlyCollection.IRpcMethod [▶ 2625]..)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T. (Inherited from ReadOnlyCollection.IRpcMethod [▶ 2625]..)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryGetMethod(Int32, IRpcMethod.) [▶ 2860]	Tries to get the specified method.
	TryGetMethod(String, IRpcMethod.) [▶ 2861]	Tries to get the specified method.


Reference




[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.118.1 ReadOnlyRpcMethodCollection Properties

The [ReadOnlyRpcMethodCollection](#) [[▶ 2855](#)] type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.IRpcMethod [▶ 2625]..)

	Name	Description
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.IRpcMethod [▸ 2625]..)
	Item.String. [▸ 2857]	Gets the IRpcMethod [▸ 2625] with the specified method name.
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.IRpcMethod [▸ 2625]..)



Reference

[ReadOnlyRpcMethodCollection Class \[▸ 2855\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.118.1.1 ReadOnlyRpcMethodCollection.Item Property

Overload List

	Name	Description
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.IRpcMethod [▸ 2625]..)
	Item.String. [▸ 2857]	Gets the IRpcMethod [▸ 2625] with the specified method name.

Reference

[ReadOnlyRpcMethodCollection Class \[▸ 2855\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.118.1.1.1 ReadOnlyRpcMethodCollection.Item Property (String)

Gets the [IRpcMethod \[▸ 2625\]](#) with the specified method name.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IRpcMethod this[
    string methodName
] { get; }
```

Parameters

methodName	Type: System.String Name of the method.
------------	--

Return Value

Type: [IRpcMethod \[▸ 2625\]](#)
RpcMethod.

Implements

[IRpcMethodCollection.Item.String. \[▸ 2631\]](#)

Exceptions

Exception	Condition
KeyNotFoundException	

Reference

[ReadOnlyRpcMethodCollection Class \[► 2855\]](#)














[Item Overload \[► 2857\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.118.2 ReadOnlyRpcMethodCollection Methods

The [ReadOnlyRpcMethodCollection \[► 2855\]](#) type exposes the following members.

Methods

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IRpcMethod [► 2625].)
	Contains(String) [► 2859]	Determines whether this collection contains the specified method name.
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.IRpcMethod [► 2625].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IRpcMethod [► 2625].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IRpcMethod [► 2625].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetMethod(Int32, IRpcMethod.) [► 2860]	Tries to get the specified method.
	TryGetMethod(String, IRpcMethod.) [► 2861]	Tries to get the specified method.



Reference

[ReadOnlyRpcMethodCollection Class \[► 2855\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.118.2.1 ReadOnlyRpcMethodCollection.Contains Method

Overload List

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.IRpcMethod [▸ 2625]..)
	Contains(String) [▸ 2859]	Determines whether this collection contains the specified method name.

Reference

[ReadOnlyRpcMethodCollection Class \[▸ 2855\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.118.2.1.1 ReadOnlyRpcMethodCollection.Contains Method (String)

Determines whether this collection contains the specified method name.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(
    string methodName
)
```

Parameters

methodName	Type: System.String Name of the method.
------------	--

Return Value

Type: Boolean
true if contained.; otherwise, false.

Implements

[IRpcMethodCollection.Contains\(String\) \[▸ 2633\]](#)

Reference



[ReadOnlyRpcMethodCollection Class \[▸ 2855\]](#)

[Contains Overload \[▸ 2859\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.118.2.2 ReadOnlyRpcMethodCollection.TryGetMethod Method

Overload List

	Name	Description
	TryGetMethod(Int32, IRpcMethod.) [▶ 2860]	Tries to get the specified method.
	TryGetMethod(String, IRpcMethod.) [▶ 2861]	Tries to get the specified method.

Reference

[ReadOnlyRpcMethodCollection Class](#) [▶ 2855]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.118.2.2.1 ReadOnlyRpcMethodCollection.TryGetMethod Method (Int32, IRpcMethod.)

Tries to get the specified method.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetMethod(
    int vTableIndex,
    out IRpcMethod?? method
)
```

Parameters

vTableIndex	Type: System.Int32 vTableIndex.
method	Type: TwinCAT.TypeSystem.IRpcMethod [▶ 2625]. The method if found, NULL otherwise.

Return Value

Type: Boolean
true if found, false otherwise.

Implements

[IRpcMethodCollection.TryGetMethod\(Int32, IRpcMethod.\)](#) [▶ 2634]

Reference

[ReadOnlyRpcMethodCollection Class](#) [▶ 2855]

[TryGetMethod Overload](#) [▶ 2860]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.118.2.2 ReadOnlyRpcMethodCollection.TryGetMethod Method (String, IRpcMethod.)

Tries to get the specified method.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetMethod(  
    string methodName,  
    out IRpcMethod?? method  
)
```

Parameters

methodName	Type: System.String Name of the method.
method	Type: TwinCAT.TypeSystem.IRpcMethod [▶ 2625]. The method if found, NULL otherwise.

Return Value

Type: Boolean
true if found, false otherwise.

Implements

[IRpcMethodCollection.TryGetMethod\(String, IRpcMethod.\)](#) [[▶ 2634](#)]

Reference

[ReadOnlyRpcMethodCollection Class](#) [[▶ 2855](#)]

[TryGetMethod Overload](#) [[▶ 2860](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.119 ReadOnlySymbolCollection Class

ReadOnly collection containing [ISymbol](#) [[▶ 2691](#)] objects.

Inheritance Hierarchy

System.Object

System.Collections.ObjectModel.ReadOnlyCollection.[ISymbol](#) [[▶ 2691](#)].

[TwinCAT.TypeSystem.Generic.ReadOnlyInstanceCollection](#) [[▶ 3145](#)].[ISymbol](#) [[▶ 2691](#)].

[TwinCAT.TypeSystem.Generic.ReadOnlySymbolCollection](#) [[▶ 3161](#)].[ISymbol](#) [[▶ 2691](#)].

[TwinCAT.TypeSystem.ReadOnlySymbolCollection](#)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229


Syntax

C#







```
public class ReadOnlySymbolCollection : ReadOnlySymbolCollection<ISymbol>,
    ISymbolCollection, ISymbolCollection<ISymbol>, IInstanceCollection<ISymbol>,
    IList<ISymbol>, ICollection<ISymbol>, IEnumerable<ISymbol>,
    IEnumerable
```

The ReadOnlySymbolCollection type exposes the following members.









Constructors









	Name	Description
	ReadOnlySymbolCollection [▶ 2863]	Initializes a new instance of the ReadOnlySymbolCollection class.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.ISymbol [▶ 2691].)
	Empty [▶ 2864]	Returns an Empty collection.
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.ISymbol [▶ 2691].)
	Item.String. [▶ 3148]	Gets the element with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.ISymbol [▶ 2691].)
	Mode [▶ 3149]	Gets the InstanceCollectionMode [▶ 2573]. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)

Methods

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.ISymbol [▶ 2691].)
	Contains(String) [▶ 3150]	Determines whether the ReadOnlyInstanceCollection.T. [▶ 3145] contains an instance with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	ContainsName [▶ 3151]	Determines whether the specified instance is contained. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.ISymbol [▶ 2691].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.ISymbol [▶ 2691].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)

	Name	Description
	GetInstance [▶ 3152]	Gets the IInstance [▶ 2549] by instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	GetInstanceByName [▶ 3152]	Gets the IInstance [▶ 2549] by instance name. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.ISymbol [▶ 2691].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInstance [▶ 3153]	Tries to get the instance with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	TryGetInstanceByName [▶ 3153]	Tries to get the instance by name. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)

Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.119.1 ReadOnlySymbolCollection Constructor

Initializes a new instance of the [ReadOnlySymbolCollection \[▶ 2861\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlySymbolCollection(
    IInstanceCollection<ISymbol> symbols
)
```

Parameters

symbols	Type: TwinCAT.TypeSystem.IInstanceCollection [▶ 2554].ISymbol [▶ 2691]. The symbols.
---------	---

Reference







[ReadOnlySymbolCollection Class \[▶ 2861\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.119.2 ReadOnlySymbolCollection Properties

The [ReadOnlySymbolCollection \[▶ 2861\]](#) type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.ISymbol [▶ 2691]..)
	Empty [▶ 2864]	Returns an Empty collection.
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.ISymbol [▶ 2691]..)
	Item.String. [▶ 3148]	Gets the element with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.ISymbol [▶ 2691]..)
	Mode [▶ 3149]	Gets the InstanceCollectionMode [▶ 2573]. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)

Reference

[ReadOnlySymbolCollection Class \[▶ 2861\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.119.2.1 ReadOnlySymbolCollection.Empty Property

Returns an Empty collection.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static ReadOnlySymbolCollection Empty { get; }
```

Return Value

Type: [ReadOnlySymbolCollection \[▶ 2861\]](#)

ReadOnlySymbolCollection.

Reference


[ReadOnlySymbolCollection Class \[▶ 2861\]](#)
















[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.119.3 ReadOnlySymbolCollection Methods

The [ReadOnlySymbolCollection \[▶ 2861\]](#) type exposes the following members.

Methods

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.ISymbol [▶ 2691]..)

	Name	Description
	Contains(String) [▶ 3150]	Determines whether the ReadOnlyInstanceCollection.T. [▶ 3145] contains an instance with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145] .)
	ContainsName [▶ 3151]	Determines whether the specified instance is contained. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145] .)
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.ISymbol [▶ 2691] ..)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T. . (Inherited from ReadOnlyCollection.ISymbol [▶ 2691] ..)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetInstance [▶ 3152]	Gets the IInstance [▶ 2549] by instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145] .)
	GetInstanceByName [▶ 3152]	Gets the IInstance [▶ 2549] by instance name. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145] .)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T. . (Inherited from ReadOnlyCollection.ISymbol [▶ 2691] ..)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInstance [▶ 3153]	Tries to get the instance with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145] .)
	TryGetInstanceByName [▶ 3153]	Tries to get the instance by name. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145] .)

Reference

[ReadOnlySymbolCollection Class](#) [\[▶ 2861\]](#)

[TwinCAT.TypeSystem Namespace](#) [\[▶ 2083\]](#)

6.11.119.3.1 ReadOnlySymbolCollection.Empty Method

Returns an Empty collection.

Namespace: [TwinCAT.TypeSystem](#) [\[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public static ReadOnlySymbolCollection Empty()
```

Return Value

Type: [ReadOnlySymbolCollection](#) [▶ 2861]
 ReadOnlySymbolCollection.

Reference

[ReadOnlySymbolCollection Class](#) [▶ 2861]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.120 ReadOnlyTypeAttributeCollection Class

Read only version of the [TypeAttributeCollection](#) [▶ 2952]

Inheritance Hierarchy

System.Object

System.Collections.ObjectModel.ReadOnlyCollection.ITypeAttribute [▶ 2725].

TwinCAT.TypeSystem.ReadOnlyTypeAttributeCollection

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229






Syntax

C#




```
public class ReadOnlyTypeAttributeCollection : ReadOnlyCollection<ITypeAttribute>,
    ITypeAttributeCollection, IList<ITypeAttribute>, ICollection<ITypeAttribute>,
    IEnumerable<ITypeAttribute>, IEnumerable
```











The ReadOnlyTypeAttributeCollection type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.ITypeAttribute [▶ 2725]..)
	Empty [▶ 2868]	Returns an empty ReadOnlyTypeAttributeCollection
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.ITypeAttribute [▶ 2725]..)
	Item.String. [▶ 2868]	Gets the String with the specified name.
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.ITypeAttribute [▶ 2725]..)

Methods

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.ITypeAttribute [▶ 2725]..)
	Contains(String) [▶ 2870]	Determines whether this ReadOnlyTypeAttributeCollection contains the specified attribute.
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.ITypeAttribute [▶ 2725]..)

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.ITypeAttribute [▶ 2725].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.ITypeAttribute [▶ 2725].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetAttribute [▶ 2871]	Tries to get the specified ITypeAttribute [▶ 2725]
	TryGetValue [▶ 2871]	Tries to get the specified Attribute value.






Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.120.1 ReadOnlyTypeAttributeCollection Properties

The [ReadOnlyTypeAttributeCollection \[▶ 2866\]](#) type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.ITypeAttribute [▶ 2725].)
	Empty [▶ 2868]	Returns an empty ReadOnlyTypeAttributeCollection [▶ 2866]
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.ITypeAttribute [▶ 2725].)
	Item.String. [▶ 2868]	Gets the String with the specified name.
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.ITypeAttribute [▶ 2725].)



Reference

[ReadOnlyTypeAttributeCollection Class \[▶ 2866\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.120.1.1 ReadOnlyTypeAttributeCollection.Item Property

Overload List

	Name	Description
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.ITypeAttribute [▸ 2725].)
	Item.String. [▸ 2868]	Gets the String with the specified name.

Reference

[ReadOnlyTypeAttributeCollection Class](#) [▸ 2866]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.120.1.1.1 ReadOnlyTypeAttributeCollection.Item Property (String)

Gets the String with the specified name.

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string this[
    string name
] { get; }
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: String
System.String.

Implements

[ITypeAttributeCollection.Item.String.](#) [▸ 2728]

Reference

[ReadOnlyTypeAttributeCollection Class](#) [▸ 2866]

[Item Overload](#) [▸ 2868]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.120.1.2 ReadOnlyTypeAttributeCollection.Empty Property

Returns an empty [ReadOnlyTypeAttributeCollection](#). [▸ 2866]

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ReadOnlyTypeAttributeCollection Empty { get; }
```

Return Value

Type: [ReadOnlyTypeAttributeCollection \[▶ 2866\]](#)
 ReadOnlyTypeAttributeCollection.

Reference

[ReadOnlyTypeAttributeCollection Class \[▶ 2866\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.120.2 ReadOnlyTypeAttributeCollection Methods

The [ReadOnlyTypeAttributeCollection \[▶ 2866\]](#) type exposes the following members.

Methods

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.ITypeAttribute [▶ 2725].)
	Contains(String) [▶ 2870]	Determines whether this ReadOnlyTypeAttributeCollection [▶ 2866] contains the specified attribute.
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.ITypeAttribute [▶ 2725].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.ITypeAttribute [▶ 2725].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.ITypeAttribute [▶ 2725].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetAttribute [▶ 2871]	Tries to get the specified ITypeAttribute [▶ 2725]
	TryGetValue [▶ 2871]	Tries to get the specified Attribute value.



Reference

[ReadOnlyTypeAttributeCollection Class \[▶ 2866\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.120.2.1 ReadOnlyTypeAttributeCollection.Contains Method

Overload List

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.ITypeAttribute [▶ 2725]..)
	Contains(String) [▶ 2870]	Determines whether this ReadOnlyTypeAttributeCollection [▶ 2866] contains the specified attribute.

Reference

[ReadOnlyTypeAttributeCollection Class](#) [[▶ 2866](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.120.2.1.1 ReadOnlyTypeAttributeCollection.Contains Method (String)

Determines whether this [ReadOnlyTypeAttributeCollection](#) [[▶ 2866](#)] contains the specified attribute.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(
    string name
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: Boolean
true if [contains] [the specified name]; otherwise, false.

Implements

[ITypeAttributeCollection.Contains\(String\)](#) [[▶ 2730](#)]

Reference

[ReadOnlyTypeAttributeCollection Class](#) [[▶ 2866](#)]

[Contains Overload](#) [[▶ 2870](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.120.2.2 ReadOnlyTypeAttributeCollection.Empty Method

Returns an empty [ReadOnlyTypeAttributeCollection](#) [[▶ 2866](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:
5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public static ReadOnlyTypeAttributeCollection Empty()
```

Return Value

Type: [ReadOnlyTypeAttributeCollection](#) [[▸ 2866](#)]

ReadOnlyTypeAttributeCollection.

Reference

[ReadOnlyTypeAttributeCollection Class](#) [[▸ 2866](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.120.2.3 ReadOnlyTypeAttributeCollection.TryGetAttribute Method

Tries to get the specified [ITypeAttribute](#) [[▸ 2725](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetAttribute(
    string name,
    out ITypeAttribute?? attribute
)
```

Parameters

name	Type: System.String The name of the ITypeAttribute [▸ 2725].
attribute	Type: TwinCAT.TypeSystem.ITypeAttribute [▸ 2725]. The attribute.

Return Value

Type: Boolean

true if found, false otherwise.

Implements

[ITypeAttributeCollection.TryGetAttribute\(String, ITypeAttribute.\)](#) [[▸ 2730](#)]

Reference

[ReadOnlyTypeAttributeCollection Class](#) [[▸ 2866](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.120.2.4 ReadOnlyTypeAttributeCollection.TryGetValue Method

Tries to get the specified Attribute value.

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetValue(
    string name,
    out string?? value
)
```

Parameters

name	Type: System.String The name.
value	Type: System.String. The value.

Return Value

Type: Boolean
true if the value could be found, false otherwise.

Implements

[ITypeAttributeCollection.TryGetValue\(String, String.\)](#) [[▶ 2731](#)]

Reference

[ReadOnlyTypeAttributeCollection Class](#) [[▶ 2866](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.121 ResultDataTypes Class

Class representing the asynchronous result of reading a [IDataTypeCollection](#) [[▶ 2484](#)] via ADS. Implements the [ResultValue.TValue.](#) [[▶ 1181](#)]

Inheritance Hierarchy

System.Object

[TwinCAT.Ads.ResultAds](#) [[▶ 1116](#)]

[TwinCAT.Ads.ResultValue](#) [[▶ 1181](#)].[IDataTypeCollection](#) [[▶ 2486](#)].[IDataType](#) [[▶ 2475](#)].

[TwinCAT.TypeSystem.ResultDataTypes](#)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#







```
public class ResultDataTypes : ResultValue<IDataTypeCollection<IDataType>>
```

The ResultDataTypes type exposes the following members.










Constructors

	Name	Description
	ResultDataTypes [▶ 2873]	Initializes a new instance of the ResultDataTypes class.

Properties

	Name	Description
	DataTypes [▸ 2874]	Gets the data types.
	ErrorCode [▸ 1120]	Gets the ADS Error code bound to this Result [▸ 1116] object. (Inherited from ResultAds [▸ 1116] .)
	Failed [▸ 1121]	Gets a value indicating whether the ResultAds [▸ 1116] state is failed. (Inherited from ResultAds [▸ 1116] .)
	Invokeld [▸ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▸ 1116] .)
	Succeeded [▸ 1121]	Gets a value indicating whether the ResultAds [▸ 1116] state is succeeded. (Inherited from ResultAds [▸ 1116] .)
	Value [▸ 1185]	The value object. (Inherited from ResultValue.TValue. [▸ 1181] .)

Methods

	Name	Description
	CreateError [▸ 2875]	Creates an error result.
	CreateSuccess [▸ 2876]	Creates a success result
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [▸ 1128]	Sets the error state of this ResultAds [▸ 1116] (Inherited from ResultAds [▸ 1116] .)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Remarks

The Result value of this operation can be accessed by the [DataTypes \[▸ 2874\]](#) property.

Reference

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

[TwinCAT.Ads.ResultValue.TValue. \[▸ 1181\]](#)

6.11.121.1 ResultDataTypes Constructor

Initializes a new instance of the [ResultDataTypes \[▸ 2872\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ResultDataTypes (
    AdsErrorCode errorCode,
    IDataValueCollection<IDataType>? dataTypes
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▸ 664] The error code.
dataTypes	Type: TwinCAT.TypeSystem.IDataValueCollection [▸ 2486]. IDataType [▸ 2475]. The datatype collection..

Reference







[ResultDataTypes Class](#) [▸ 2872]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.121.2 ResultDataTypes Properties

The [ResultDataTypes](#) [▸ 2872] type exposes the following members.

Properties

	Name	Description
	DataTypes [▸ 2874]	Gets the data types.
	ErrorCode [▸ 1120]	Gets the ADS Error code bound to this Result [▸ 1116] object. (Inherited from ResultAds [▸ 1116].)
	Failed [▸ 1121]	Gets a value indicating whether the ResultAds [▸ 1116] state is failed. (Inherited from ResultAds [▸ 1116].)
	Invokeld [▸ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▸ 1116].)
	Succeeded [▸ 1121]	Gets a value indicating whether the ResultAds [▸ 1116] state is succeeded. (Inherited from ResultAds [▸ 1116].)
	Value [▸ 1185]	The value object. (Inherited from ResultValue.TValue . [▸ 1181].)

Reference

[ResultDataTypes Class](#) [▸ 2872]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.121.2.1 ResultDataTypes.DataTypes Property

Gets the data types.

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataValueCollection<IDataType>? DataTypes { get; }
```

Property Value

Type: [IDataTypeCollection](#) [[▶ 2486](#)].[IDataType](#) [[▶ 2475](#)].
The data types.

Reference










[ResultDataTypes Class](#) [[▶ 2872](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.121.3 ResultDataTypes Methods

The [ResultDataTypes](#) [[▶ 2872](#)] type exposes the following members.

Methods

	Name	Description
	CreateError [▶ 2875]	Creates an error result.
	CreateSuccess [▶ 2876]	Creates a success result
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[ResultDataTypes Class](#) [[▶ 2872](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.121.3.1 ResultDataTypes.CreateError Method

Creates an error result.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultDataTypes CreateError(
    AdsErrorCode error
)
```

Parameters

error	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error.
-------	--

Return Value

Type: [ResultDataTypes](#) [[▶ 2872](#)]
ResultRpcMethod.

Reference

[ResultDataTypes Class](#) [[▶ 2872](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.121.3.2 ResultDataTypes.CreateSuccess Method

Creates a success result

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static ResultDataTypes CreateSuccess(
    IDataValueCollection<IDataType>? dataTypes
)
```

Parameters

dataTypes	Type: TwinCAT.TypeSystem.IDataValueCollection [▶ 2486]. IDataType [▶ 2475]. The data types.
-----------	--

Return Value

Type: [ResultDataTypes](#) [[▶ 2872](#)]
ResultRpcMethod.

Reference

[ResultDataTypes Class](#) [[▶ 2872](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.122 ResultDynamicSymbols Class

Class representing the asynchronous result of reading a dynamic symbol collection via ADS. Implements the [ResultValue.TValue](#). [[▶ 1181](#)]

Inheritance Hierarchy

System.Object

[TwinCAT.Ads.ResultAds](#) [[▶ 1116](#)]

[TwinCAT.Ads.ResultValue](#) [[▶ 1181](#)]. [IDynamicSymbolsCollection](#) [[▶ 2503](#)].

[TwinCAT.TypeSystem.ResultSymbols](#) [[▶ 2884](#)]. [IDynamicSymbolsCollection](#) [[▶ 2503](#)].

[TwinCAT.TypeSystem.ResultDynamicSymbols](#)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#







```
public class ResultDynamicSymbols : ResultSymbols<IDynamicSymbolsCollection>
```

The ResultDynamicSymbols type exposes the following members.










Constructors

	Name	Description
	ResultDynamicSymbols [▶ 2878]	Initializes a new instance of the ResultDynamicSymbols class.

Properties

	Name	Description
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)
	Symbols [▶ 2886]	Get the Symbols enumerable (T) as result of an asynchronous operation. (Inherited from ResultSymbols.T . [▶ 2884].)
	Value [▶ 1185]	The value object. (Inherited from ResultValue.TValue . [▶ 1181].)

Methods

	Name	Description
	CreateError [▶ 2879]	Creates an Error Result.
	CreateSuccess [▶ 2880]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Remarks

This result class is used to return Symbol instances of dynamic nature in an [IDynamicSymbolsCollection](#) [[▸ 2503](#)]. The value/result of the completed operation can be returned by the [Symbols](#) [[▸ 2886](#)] property.

Reference

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

[TwinCAT.Ads.ResultValue.TValue.](#) [[▸ 1181](#)]

6.11.122.1 ResultDynamicSymbols Constructor

Initializes a new instance of the [ResultDynamicSymbols](#) [[▸ 2876](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ResultDynamicSymbols(
    AdsErrorCode errorCode,
    IDynamicSymbolsCollection? symbols
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▸ 664] The error code.
symbols	Type: TwinCAT.TypeSystem.IDynamicSymbolsCollection [▸ 2503] The result symbols.

Reference







[ResultDynamicSymbols Class](#) [[▸ 2876](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.122.2 ResultDynamicSymbols Properties

The [ResultDynamicSymbols](#) [[▸ 2876](#)] type exposes the following members.

Properties

	Name	Description
	ErrorCode [▸ 1120]	Gets the ADS Error code bound to this Result [▸ 1116] object. (Inherited from ResultAds [▸ 1116].)
	Failed [▸ 1121]	Gets a value indicating whether the ResultAds [▸ 1116] state is failed. (Inherited from ResultAds [▸ 1116].)
	Invokeld [▸ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▸ 1116].)
	Succeeded [▸ 1121]	Gets a value indicating whether the ResultAds [▸ 1116] state is succeeded. (Inherited from ResultAds [▸ 1116].)
	Symbols [▸ 2886]	Get the Symbols enumerable (T) as result of an asynchronous operation. (Inherited from ResultSymbols.T. [▸ 2884].)
	Value [▸ 1185]	The value object. (Inherited from ResultValue.TValue. [▸ 1181].)

Reference










[ResultDynamicSymbols Class \[▶ 2876\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.122.3 ResultDynamicSymbols Methods

The [ResultDynamicSymbols \[▶ 2876\]](#) type exposes the following members.

Methods

	Name	Description
	CreateError [▶ 2879]	Creates an Error Result.
	CreateSuccess [▶ 2880]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	setError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116] .)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[ResultDynamicSymbols Class \[▶ 2876\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.122.3.1 ResultDynamicSymbols.CreateError Method

Creates an Error Result.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultDynamicSymbols CreateError(
    AdsErrorCode errorCode
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▶ 664] The error code.
-----------	---

Return Value

Type: [ResultDynamicSymbols](#) [[▸ 2876](#)]
 ResultValue<T>.

Reference

[ResultDynamicSymbols Class](#) [[▸ 2876](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.122.3.2 ResultDynamicSymbols.CreateSuccess Method

Creates a success result.

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static ResultDynamicSymbols CreateSuccess(
    IDynamicSymbolsCollection symbols
)
```

Parameters

symbols	Type: TwinCAT.TypeSystem.IDynamicSymbolsCollection [▸ 2503]
---------	---

Return Value

Type: [ResultDynamicSymbols](#) [[▸ 2876](#)]
 ResultValue<T>.

Reference

[ResultDynamicSymbols Class](#) [[▸ 2876](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.123 ResultSymbols Class

Class representing the asynchronous result of reading an symbol collection of type [ISymbolCollection.T](#). [[▸ 2700](#)] via ADS. Implements the [ResultValue.TValue](#). [[▸ 1181](#)]

Inheritance Hierarchy

System.Object

[TwinCAT.Ads.ResultAds](#) [[▸ 1116](#)]

[TwinCAT.Ads.ResultValue](#) [[▸ 1181](#)].[ISymbolCollection](#) [[▸ 2700](#)].[ISymbol](#) [[▸ 2691](#)].

[TwinCAT.TypeSystem.ResultSymbols](#) [[▸ 2884](#)].[ISymbolCollection](#) [[▸ 2700](#)].[ISymbol](#) [[▸ 2691](#)].

[TwinCAT.TypeSystem.ResultSymbols](#)

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#







```
public class ResultSymbols : ResultSymbols<ISymbolCollection<ISymbol>>
```

The ResultSymbols type exposes the following members.










Constructors

	Name	Description
	ResultSymbols [▶ 2882]	Initializes a new instance of the ResultSymbols class.

Properties

	Name	Description
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116] .)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116] .)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116] .)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116] .)
	Symbols [▶ 2886]	Get the Symbols enumerable (T) as result of an asynchronous operation. (Inherited from ResultSymbols.T . [▶ 2884] .)
	Value [▶ 1185]	The value object. (Inherited from ResultValue.TValue . [▶ 1181] .)

Methods

	Name	Description
	CreateError [▶ 2883]	Creates an Error Result.
	CreateSuccess [▶ 2884]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116] .)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Remarks

This result class is used to return generic Symbol instances. in an enumerable class. The value/result of the completed operation can be returned by the [Symbols](#) [\[▶ 2886\]](#) property.

Reference

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

[TwinCAT.Ads.ResultValue.TValue. \[► 1181\]](#)

6.11.123.1 ResultSymbols Constructor

Initializes a new instance of the [ResultSymbols \[► 2880\]](#) class.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ResultSymbols(
    AdsErrorCode errorCode,
    ISymbolCollection<ISymbol>? symbols
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [► 664] The error code.
symbols	Type: TwinCAT.TypeSystem.ISymbolCollection [► 2700] . ISymbol [► 2691] . The result symbols.

Reference







[ResultSymbols Class \[► 2880\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.123.2 ResultSymbols Properties

The [ResultSymbols \[► 2880\]](#) type exposes the following members.

Properties

	Name	Description
	ErrorCode [► 1120]	Gets the ADS Error code bound to this Result [► 1116] object. (Inherited from ResultAds [► 1116] .)
	Failed [► 1121]	Gets a value indicating whether the ResultAds [► 1116] state is failed. (Inherited from ResultAds [► 1116] .)
	Invokeld [► 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [► 1116] .)
	Succeeded [► 1121]	Gets a value indicating whether the ResultAds [► 1116] state is succeeded. (Inherited from ResultAds [► 1116] .)
	Symbols [► 2886]	Get the Symbols enumerable (T) as result of an asynchronous operation. (Inherited from ResultSymbols.T. [► 2884] .)
	Value [► 1185]	The value object. (Inherited from ResultValue.TValue. [► 1181] .)

Reference










[ResultSymbols Class \[► 2880\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.123.3 ResultSymbols Methods

The [ResultSymbols \[▸ 2880\]](#) type exposes the following members.

Methods

	Name	Description
	CreateError [▸ 2883]	Creates an Error Result.
	CreateSuccess [▸ 2884]	Creates a success result.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [▸ 1128]	Sets the error state of this ResultAds [▸ 1116] (Inherited from ResultAds [▸ 1116] .)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[ResultSymbols Class \[▸ 2880\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.123.3.1 ResultSymbols.CreateError Method

Creates an Error Result.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultSymbols CreateError(
    AdsErrorCode errorCode
)
```

Parameters

errorCode	Type: TwinCAT.Ads.AdsErrorCode [▸ 664] The error code.
-----------	---

Return Value

Type: [ResultSymbols \[▸ 2880\]](#)

ResultValue<T>.

Reference[ResultSymbols Class \[▸ 2880\]](#)[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)**6.11.123.3.2 ResultSymbols.CreateSuccess Method**

Creates a success result.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static ResultSymbols CreateSuccess(
    ISymbolCollection<ISymbol> symbols
)
```

Parameters

symbols	Type: TwinCAT.TypeSystem.ISymbolCollection [▸ 2700] . ISymbol [▸ 2691] .
---------	--

Return ValueType: [ResultSymbols \[▸ 2880\]](#)

ResultValue<T>.

Reference[ResultSymbols Class \[▸ 2880\]](#)[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)**6.11.124 ResultSymbols.T. Class**Class representing the asynchronous result of reading a symbol enumeration of type IEnumerable.T. via ADS. Implements the [ResultValue.TValue. \[▸ 1181\]](#)**Inheritance Hierarchy**

System.Object

[TwinCAT.Ads.ResultAds \[▸ 1116\]](#)[TwinCAT.Ads.ResultValue \[▸ 1181\].T.](#)[TwinCAT.TypeSystem.ResultSymbols.T.](#)[TwinCAT.TypeSystem.ResultDynamicSymbols \[▸ 2876\]](#)[TwinCAT.TypeSystem.ResultSymbols \[▸ 2880\]](#)**Namespace:** [TwinCAT.TypeSystem \[▸ 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**







```
public class ResultSymbols<T> : ResultValue<T>
where T : class, Object, IEnumerable<ISymbol>
```

Type Parameters








T	The type parameter is an IEnumerable.T..
---	--

The ResultSymbols.T. type exposes the following members.

Properties

	Name	Description
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)
	Symbols [▶ 2886]	Get the Symbols enumerable (T) as result of an asynchronous operation.
	Value [▶ 1185]	The value object. (Inherited from ResultValue.TValue . [▶ 1181].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Remarks

This result class is used to return generic Symbol instances. in an enumerable class. The value/result of the completed operation can be returned by the [Symbols](#) [[▶ 2886](#)] property.

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

[TwinCAT.Ads.ResultValue.TValue](#). [[▶ 1181](#)]

6.11.124.1 ResultSymbols.T. Constructor

Initializes a new instance of the [ResultSymbols.T](#). [[▶ 2884](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public ResultSymbols(
    AdsErrorCode errorCode,
    T symbols
)
```

Parameters

errorCode Type: [TwinCAT.Ads.AdsErrorCode](#) [[▶ 664](#)]
The error code.

symbols Type: [T](#) [[▶ 2884](#)]
The symbols.

Reference







[ResultSymbols.T. Class](#) [[▶ 2884](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.124.2 **ResultSymbols.T. Properties**

The [ResultSymbols.T](#). [[▶ 2884](#)] generic type exposes the following members.

Properties

	Name	Description
	ErrorCode [▶ 1120]	Gets the ADS Error code bound to this Result [▶ 1116] object. (Inherited from ResultAds [▶ 1116].)
	Failed [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is failed. (Inherited from ResultAds [▶ 1116].)
	Invokeld [▶ 1121]	Gets the ADS requests invoke identifier (or 0 if not supported) (Inherited from ResultAds [▶ 1116].)
	Succeeded [▶ 1121]	Gets a value indicating whether the ResultAds [▶ 1116] state is succeeded. (Inherited from ResultAds [▶ 1116].)
	Symbols [▶ 2886]	Get the Symbols enumerable (T) as result of an asynchronous operation.
	Value [▶ 1185]	The value object. (Inherited from ResultValue.TValue . [▶ 1181].)

Reference

[ResultSymbols.T. Class](#) [[▶ 2884](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.124.2.1 **ResultSymbols.T..Symbols Property**

Get the Symbols enumerable (T) as result of an asynchronous operation.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public T Symbols { get; }
```

Property Value

Type: [T](#) [[▶ 2884](#)]
The symbols.

Reference








[ResultSymbols.T. Class](#) [[▶ 2884](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.124.3 ResultSymbols.T. Methods

The [ResultSymbols.T.](#) [[▶ 2884](#)] generic type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [▶ 1128]	Sets the error state of this ResultAds [▶ 1116] (Inherited from ResultAds [▶ 1116].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[ResultSymbols.T. Class](#) [[▶ 2884](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.125 RpcInvokeException Class

Class [RpcInvokeException](#). Implements the [SymbolException](#) [[▶ 2933](#)]

Inheritance Hierarchy

System.Object
 System.Exception
[TwinCAT.AdsException](#) [[▶ 61](#)]
[TwinCAT.TypeSystem.SymbolException](#) [[▶ 2933](#)]
[TwinCAT.TypeSystem.RpcInvokeException](#)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]




Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax**C#**











```
[SerializableAttribute]
public class RpcInvokeException : SymbolException
```

The RpcInvokeException type exposes the following members.





Constructors





	Name	Description
	RpcInvokeException (SerializationInfo, StreamingContext) [▶ 2889]	Initializes a new instance of the RpcInvokeException class.
	RpcInvokeException (IInterfaceInstance, String, Exception) [▶ 2891]	Initializes a new instance of the RpcInvokeException class.
	RpcInvokeException (IInterfaceInstance, String, Int32) [▶ 2892]	Initializes a new instance of the RpcInvokeException class.

Properties


	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	InstancePath [▶ 2944]	Gets the instance path. (Inherited from SymbolException [▶ 2933].)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	Symbol [▶ 2944]	Gets the symbol. (Inherited from SymbolException [▶ 2933].)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)

	Name	Description
	GetObjectData [▶ 2893]	When overridden in a derived class, sets the <code>SerializationInfo</code> with information about the exception. (Overrides SymbolException.GetObjectData(SerializationInfo, StreamingContext) [▶ 2945].)
	<code>GetType</code>	Gets the runtime type of the current instance. (Inherited from <code>Exception</code> .)
	<code>MemberwiseClone</code>	Creates a shallow copy of the current <code>Object</code> . (Inherited from <code>Object</code> .)
	<code>ToString</code>	Creates and returns a string representation of the current exception. (Inherited from <code>Exception</code> .)

Events

	Name	Description
	<code>SerializeObjectState</code>	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from <code>Exception</code> .)




Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

[TwinCAT.TypeSystem.SymbolException](#) [▶ 2933]

6.11.125.1 RpcInvokeException Constructor

Overload List

	Name	Description
	RpcInvokeException (SerializationInfo , StreamingContext) [▶ 2889]	Initializes a new instance of the RpcInvokeException [▶ 2887] class.
	RpcInvokeException (IInterfaceInstance , String , Exception) [▶ 2891]	Initializes a new instance of the RpcInvokeException [▶ 2887] class.
	RpcInvokeException (IInterfaceInstance , String , Int32) [▶ 2892]	Initializes a new instance of the RpcInvokeException [▶ 2887] class.

Reference

[RpcInvokeException Class](#) [▶ 2887]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.125.1.1 RpcInvokeException Constructor (SerializationInfo, StreamingContext)

Initializes a new instance of the [RpcInvokeException](#) [▶ 2887] class.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected RpcInvokeException(
    SerializationInfo serializationInfo,
    StreamingContext streamingContext
)
```

Parameters

serializationInfo	Type: System.Runtime.Serialization.SerializationInfo The serialization information.
streamingContext	Type: System.Runtime.Serialization.StreamingContext The streaming context.

Reference

[RpcInvokeException Class](#) [► 2887]

[RpcInvokeException Overload](#) [► 2889]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.125.1.2 RpcInvokeException Constructor (IRpcStructInstance, String, Int32)

Initializes a new instance of the [RpcInvokeException](#) [► 2887] class.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public RpcInvokeException(
    IRpcStructInstance rpcInstance,
    string methodName,
    int errorCode
)
```

Parameters

rpcInstance	Type: TwinCAT.TypeSystem.IRpcStructInstance [► 2642] The RPC instance.
methodName	Type: System.String Name of the method.
errorCode	Type: System.Int32 The error code.

Reference

[RpcInvokeException Class](#) [► 2887]

[RpcInvokeException Overload](#) [► 2889]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.125.1.3 RpcInvokeException Constructor (IRpcStructInstance, String, Exception)

Initializes a new instance of the [RpcInvokeException](#) [▶ 2887] class.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
public RpcInvokeException(
    IRpcStructInstance rpcInstance,
    string methodName,
    Exception innerException
)
```

Parameters

rpcInstance	Type: TwinCAT.TypeSystem.IRpcStructInstance [▶ 2642] The RPC instance.
methodName	Type: System.String Name of the method.
innerException	Type: System.Exception The inner exception.

Reference

[RpcInvokeException Class](#) [▶ 2887]

[RpcInvokeException Overload](#) [▶ 2889]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.125.1.4 RpcInvokeException Constructor (IInterfaceInstance, String, Exception)

Initializes a new instance of the [RpcInvokeException](#) [▶ 2887] class.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RpcInvokeException(
    IInterfaceInstance rpcInstance,
    string methodName,
    Exception innerException
)
```

Parameters

rpcInstance	Type: TwinCAT.TypeSystem.IInterfaceInstance [▶ 3014] The RPC instance.
methodName	Type: System.String Name of the method.
innerException	Type: System.Exception The inner exception.







Reference[RpcInvokeException Class \[► 2887\]](#)[RpcInvokeException Overload \[► 2889\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.125.1.5 RpcInvokeException Constructor (IInterfaceInstance, String, Int32)**Initializes a new instance of the [RpcInvokeException \[► 2887\]](#) class.**Namespace:** [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**





```
public RpcInvokeException(
    IInterfaceInstance rpcInstance,
    string methodName,
    int errorCode
)
```

Parameters

rpcInstance	Type: TwinCAT.TypeSystem.IInterfaceInstance [► 3014] The RPC instance.
methodName	Type: System.String Name of the method.
errorCode	Type: System.Int32 The error code.

Reference[RpcInvokeException Class \[► 2887\]](#)[RpcInvokeException Overload \[► 2889\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.125.2 RpcInvokeException Properties**The [RpcInvokeException \[► 2887\]](#) type exposes the following members.**Properties**

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	InstancePath [► 2944]	Gets the instance path. (Inherited from SymbolException [► 2933] .)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)

	Name	Description
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	Symbol [▶ 2944]	Gets the symbol. (Inherited from SymbolException [▶ 2933] .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Reference









[RpcInvokeException Class \[▶ 2887\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.125.3 RpcInvokeException Methods

The [RpcInvokeException \[▶ 2887\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData [▶ 2893]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Overrides SymbolException.GetObjectData(SerializationInfo, StreamingContext) [▶ 2945] .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Reference

[RpcInvokeException Class \[▶ 2887\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.125.3.1 RpcInvokeException.GetObjectData Method

When overridden in a derived class, sets the SerializationInfo with information about the exception.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override void GetObjectData(
    SerializationInfo info,
    StreamingContext context
)
```

Parameters

info	Type: System.Runtime.Serialization.SerializationInfo The SerializationInfo that holds the serialized object data about the exception being thrown.
context	Type: System.Runtime.Serialization.StreamingContext The StreamingContext that contains contextual information about the source or destination.

Implements

ISerializable.GetObjectData(SerializationInfo, StreamingContext)

Exceptions

Exception	Condition
ArgumentNullException	info

Reference


[RpcInvokeException Class \[► 2887\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.125.4 RpcInvokeException Events

The [RpcInvokeException](#) [► 2887] type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)

Reference

[RpcInvokeException Class \[► 2887\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.126 RpcMethodCollection Class

Collection of [RpcMethods](#). [► 2625]

Inheritance Hierarchy

System.Object
 TwinCAT.TypeSystem.RpcMethodCollection

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#






```
public class RpcMethodCollection : IRpcMethodCollection,
    IList<IRpcMethod>, ICollection<IRpcMethod>, IEnumerable<IRpcMethod>,
    IEnumerable
```

The RpcMethodCollection type exposes the following members.








Constructors














	Name	Description
	RpcMethodCollection. [▸ 2909]	Initializes a new instance of the RpcMethodCollection class.
	RpcMethodCollection(IEnumerable.IRpcMethod.) [▸ 2910]	Initializes a new instance of the RpcMethodCollection class.

Properties

	Name	Description
	Count [▸ 2897]	Gets the number of elements contained in the ICollection.T..
	Empty [▸ 2899]	Gets the empty collection
	IsReadOnly [▸ 2897]	Gets a value indicating whether the ICollection.T. is read-only.
	Item.Int32. [▸ 2898]	Gets or sets the element at the specified index.
	Item.String. [▸ 2899]	Gets the IRpcMethod [▸ 2625] with the specified method name.

Methods

	Name	Description
	Add [▸ 2901]	Adds an item to the ICollection.T..
	AddRange [▸ 2908]	Adds the range of Methods.
	AsReadOnly [▸ 2901]	Gets a read only collection of this RpcMethodCollection
	Clear [▸ 2902]	Removes all items from the ICollection.T..
	Contains(IRpcMethod) [▸ 2903]	Determines whether the ICollection.T. contains a specific value.
	Contains(String) [▸ 2902]	Determines whether this collection contains the specified method name.
	CopyTo [▸ 2904]	Copies to.

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 2904]	Returns an enumerator that iterates through the collection.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 2905]	Determines the index of a specific item in the IList.T..
	Insert [▶ 2905]	Inserts an item to the IList.T. at the specified index.
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 2906]	Removes the first occurrence of a specific object from the ICollection.T..
	RemoveAt [▶ 2906]	Removes the IList.T. item at the specified index.
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetMethod(Int32, IRpcMethod.) [▶ 2907]	Tries to get the specified method.
	TryGetMethod(String, IRpcMethod.) [▶ 2908]	Tries to get the specified method.






Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.126.1 RpcMethodCollection Properties

The [RpcMethodCollection](#) [▶ 2894] type exposes the following members.

Properties

	Name	Description
	Count [▶ 2897]	Gets the number of elements contained in the ICollection.T..
	Empty [▶ 2899]	Gets the empty collection
	IsReadOnly [▶ 2897]	Gets a value indicating whether the ICollection.T. is read-only.
	Item.Int32. [▶ 2898]	Gets or sets the element at the specified index.
	Item.String. [▶ 2899]	Gets the IRpcMethod [▶ 2625] with the specified method name.

Reference

[RpcMethodCollection Class](#) [▶ 2894]

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.126.1.1 RpcMethodCollection.Count Property

Gets the number of elements contained in the ICollection.T..

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Count { get; }
```

Property Value

Type: Int32

The count.

Implements

ICollection.T..Count

Reference

[RpcMethodCollection Class \[► 2894\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.126.1.2 RpcMethodCollection.IsReadOnly Property

Gets a value indicating whether the ICollection.T. is read-only.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsReadOnly { get; }
```

Property Value

Type: Boolean

true if this instance is read only; otherwise, false.

Implements

ICollection.T..IsReadOnly



Reference

[RpcMethodCollection Class \[► 2894\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.126.1.3 RpcMethodCollection.Item Property

Overload List

	Name	Description
	Item.Int32 [▶ 2898]	Gets or sets the element at the specified index.
	Item.String [▶ 2899]	Gets the IRpcMethod [▶ 2625] with the specified method name.

Reference

[RpcMethodCollection Class](#) [[▶ 2894](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.126.1.3.1 RpcMethodCollection.Item Property (Int32)

Gets or sets the element at the specified index.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IRpcMethod this[
    int index
] { get; set; }
```

Parameters

index	Type: System.Int32 The index.
-------	----------------------------------

Return Value

Type: [IRpcMethod](#) [[▶ 2625](#)]
RpcMethod.

Implements

[IList.T..Item.Int32](#).

Exceptions

Exception	Condition
NotImplementedException	

Reference

[RpcMethodCollection Class](#) [[▶ 2894](#)]

[Item Overload](#) [[▶ 2898](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.126.1.3.2 RpcMethodCollection.Item Property (String)

Gets the [IRpcMethod](#) [▸ 2625] with the specified method name.

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IRpcMethod this[
    string methodName
] { get; }
```

Parameters

methodName	Type: System.String Name of the method.
------------	--

Return Value

Type: [IRpcMethod](#) [▸ 2625]

[RpcMethod](#).

Implements

[IRpcMethodCollection.Item.String](#). [▸ 2631]

Exceptions

Exception	Condition
KeyNotFoundException	

Reference

[RpcMethodCollection Class](#) [▸ 2894]

[Item Overload](#) [▸ 2898]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.126.1.4 RpcMethodCollection.Empty Property

Gets the empty collection

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static RpcMethodCollection Empty { get; }
```

Property Value

Type: [RpcMethodCollection](#) [▸ 2894]

The empty collection

Reference





















[RpcMethodCollection Class \[▶ 2894\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.126.2 RpcMethodCollection Methods

The [RpcMethodCollection \[▶ 2894\]](#) type exposes the following members.

Methods

	Name	Description
	Add [▶ 2901]	Adds an item to the ICollection.T..
	AddRange [▶ 2908]	Adds the range of Methods.
	AsReadOnly [▶ 2901]	Gets a read only collection of this RpcMethodCollection [▶ 2894]
	Clear [▶ 2902]	Removes all items from the ICollection.T..
	Contains(IRpcMethod) [▶ 2903]	Determines whether the ICollection.T. contains a specific value.
	Contains(String) [▶ 2902]	Determines whether this collection contains the specified method name.
	CopyTo [▶ 2904]	Copies to.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 2904]	Returns an enumerator that iterates through the collection.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 2905]	Determines the index of a specific item in the IList.T..
	Insert [▶ 2905]	Inserts an item to the IList.T. at the specified index.
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 2906]	Removes the first occurrence of a specific object from the ICollection.T..
	RemoveAt [▶ 2906]	Removes the IList.T. item at the specified index.
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetMethod(Int32, IRpcMethod.) [▶ 2907]	Tries to get the specified method.
	TryGetMethod(String, IRpcMethod.) [▶ 2908]	Tries to get the specified method.

Reference

[RpcMethodCollection Class](#) [[▶ 2894](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.126.2.1 RpcMethodCollection.Add Method

Adds an item to the ICollection.T..

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Add(  
    IRpcMethod item  
)
```

Parameters

item	Type: TwinCAT.TypeSystem.IRpcMethod [▶ 2625] The object to add to the ICollection.T..
------	--

Implements

ICollection.T..Add(T)

Reference

[RpcMethodCollection Class](#) [[▶ 2894](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.126.2.2 RpcMethodCollection.AsReadOnly Method

Gets a read only collection of this [RpcMethodCollection](#) [[▶ 2894](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyRpcMethodCollection AsReadOnly()
```

Field Value

Type: [ReadOnlyRpcMethodCollection](#) [[▶ 2855](#)]

Returns a read only version of this [RpcMethodCollection](#) [[▶ 2894](#)]

Return Value

Type: [ReadOnlyRpcMethodCollection](#) [[▶ 2855](#)]

ReadOnlyRpcMethodCollection.

Reference[RpcMethodCollection Class \[► 2894\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.126.2.3 RpcMethodCollection.Clear Method**

Removes all items from the ICollection.T..



Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public void Clear()
```

Implements

ICollection.T..Clear.

Reference[RpcMethodCollection Class \[► 2894\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.126.2.4 RpcMethodCollection.Contains Method****Overload List**

	Name	Description
	Contains(IRpcMethod) [► 2903]	Determines whether the ICollection.T. contains a specific value.
	Contains(String) [► 2902]	Determines whether this collection contains the specified method name.

Reference[RpcMethodCollection Class \[► 2894\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.126.2.4.1 RpcMethodCollection.Contains Method (String)**

Determines whether this collection contains the specified method name.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public bool Contains(
    string methodName
)
```

Parameters

methodName	Type: System.String Name of the method.
------------	--

Return Value

Type: Boolean
true if contained.; otherwise, false.

Implements

[IRpcMethodCollection.Contains\(String\) \[► 2633\]](#)

Reference

[RpcMethodCollection Class \[► 2894\]](#)

[Contains Overload \[► 2902\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.126.2.4.2 RpcMethodCollection.Contains Method (IRpcMethod)

Determines whether the ICollection.T. contains a specific value.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(  
    IRpcMethod item  
)
```

Parameters

item	Type: TwinCAT.TypeSystem.IRpcMethod [► 2625] The object to locate in the ICollection.T..
------	---

Return Value

Type: Boolean
true if item is found in the ICollection.T.; otherwise, false.

Implements

[ICollection.T..Contains\(T\)](#)

Reference

[RpcMethodCollection Class \[► 2894\]](#)

[Contains Overload \[► 2902\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.126.2.5 RpcMethodCollection.CopyTo Method

Copies to.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void CopyTo(  
    IRpcMethod[] array,  
    int arrayIndex  
)
```

Parameters

array	Type: .TwinCAT.TypeSystem.IRpcMethod [▸ 2625] . The array.
arrayIndex	Type: System.Int32 Index of the array.

Implements

ICollection.T..CopyTo(.T., Int32)

Reference

[RpcMethodCollection Class \[▸ 2894\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.126.2.6 RpcMethodCollection.GetEnumerator Method

Returns an enumerator that iterates through the collection.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IEnumerator<IRpcMethod> GetEnumerator()
```

Return Value

Type: [IEnumerator.IRpcMethod \[▸ 2625\]](#).

A [IEnumerator.T.](#) that can be used to iterate through the collection.

Implements

IEnumerable.T..GetEnumerator.

Reference

[RpcMethodCollection Class \[▸ 2894\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.126.2.7 RpcMethodCollection.IndexOf Method

Determines the index of a specific item in the IList.T..

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int IndexOf(  
    IRpcMethod item  
)
```

Parameters

item	Type: TwinCAT.TypeSystem.IRpcMethod [▶ 2625] The object to locate in the IList.T..
------	---

Return Value

Type: Int32

The index of item if found in the list; otherwise, -1.

Implements

IList.T..IndexOf(T)

Reference

[RpcMethodCollection Class](#) [[▶ 2894](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.126.2.8 RpcMethodCollection.Insert Method

Inserts an item to the IList.T. at the specified index.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Insert(  
    int index,  
    IRpcMethod item  
)
```

Parameters

index	Type: System.Int32 The zero-based index at which item should be inserted.
item	Type: TwinCAT.TypeSystem.IRpcMethod [▶ 2625] The object to insert into the IList.T..

Implements

IList.T..Insert(Int32, T)

Reference[RpcMethodCollection Class \[► 2894\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.126.2.9 RpcMethodCollection.Remove Method**

Removes the first occurrence of a specific object from the ICollection.T..

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public bool Remove(
    IRpcMethod item
)
```

Parameters

item	Type: TwinCAT.TypeSystem.IRpcMethod [► 2625] The object to remove from the ICollection.T..
------	---

Return Value

Type: Boolean

true if item was successfully removed from the ICollection.T.; otherwise, false. This method also returns false if item is not found in the original ICollection.T..

Implements

ICollection.T..Remove(T)

Reference[RpcMethodCollection Class \[► 2894\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.126.2.10 RpcMethodCollection.RemoveAt Method**

Removes the IList.T. item at the specified index.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public void RemoveAt(
    int index
)
```

Parameters

index	Type: System.Int32 The zero-based index of the item to remove.
-------	---

Implements

IList.T..RemoveAt(Int32)



Reference

[RpcMethodCollection Class \[► 2894\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.126.2.11 RpcMethodCollection.TryGetMethod Method

Overload List

	Name	Description
	TryGetMethod(Int32, IRpcMethod.) [► 2907]	Tries to get the specified method.
	TryGetMethod(String, IRpcMethod.) [► 2908]	Tries to get the specified method.

Reference

[RpcMethodCollection Class \[► 2894\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

**6.11.126.2.11. RpcMethodCollection.TryGetMethod Method (Int32, IRpcMethod.)
1**

Tries to get the specified method.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetMethod(
    int vTableIndex,
    out IRpcMethod?? method
)
```

Parameters

vTableIndex	Type: System.Int32 VTable index.
method	Type: TwinCAT.TypeSystem.IRpcMethod [► 2625] . The method if found, NULL otherwise.

Return Value

Type: Boolean
true if found, false otherwise.

Implements

[IRpcMethodCollection.TryGetMethod\(Int32, IRpcMethod.\)](#) [[▶ 2634](#)]

Reference

[RpcMethodCollection Class](#) [[▶ 2894](#)]

[TryGetMethod Overload](#) [[▶ 2907](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

**6.11.126.2.11. RpcMethodCollection.TryGetMethod Method (String, IRpcMethod.)
2**

Tries to get the specified method.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public bool TryGetMethod(
    string methodName,
    out IRpcMethod?? method
)
```

Parameters

methodName	Type: System.String Name of the method.
method	Type: TwinCAT.TypeSystem.IRpcMethod [▶ 2625]. The method if fund, NULL otherwise.

Return Value

Type: Boolean
true if found, false otherwise.

Implements

[IRpcMethodCollection.TryGetMethod\(String, IRpcMethod.\)](#) [[▶ 2634](#)]

Reference

[RpcMethodCollection Class](#) [[▶ 2894](#)]

[TryGetMethod Overload](#) [[▶ 2907](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.126.2.12 RpcMethodCollection.AddRange Method

Adds the range of Methods.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void AddRange(
    IEnumerable<IRpcMethod> methods
)
```

Parameters

methods	Type: System.Collections.Generic.IEnumerable. IRpcMethod [2625]. The methods.
---------	--

Reference

[RpcMethodCollection Class](#) [[2894](#)]

[TwinCAT.TypeSystem Namespace](#) [[2083](#)]

6.11.126.3 RpcMethodCollection Constructor

Overload List

	Name	Description
	RpcMethodCollection . [2909]	Initializes a new instance of the RpcMethodCollection [2894] class.
	RpcMethodCollection(IEnumerable.IRpcMethod.) [2910]	Initializes a new instance of the RpcMethodCollection [2894] class.

Reference

[RpcMethodCollection Class](#) [[2894](#)]

[TwinCAT.TypeSystem Namespace](#) [[2083](#)]

6.11.126.3.1 RpcMethodCollection Constructor

Initializes a new instance of the [RpcMethodCollection](#) [[2894](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RpcMethodCollection()
```

Reference

[RpcMethodCollection Class](#) [[2894](#)]

[RpcMethodCollection Overload](#) [[2909](#)]

[TwinCAT.TypeSystem Namespace](#) [[2083](#)]

6.11.126.3.2 RpcMethodCollection Constructor (IEnumerable.IRpcMethod.)

Initializes a new instance of the [RpcMethodCollection](#) [▶ 2894] class.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RpcMethodCollection(
    IEnumerable<IRpcMethod> methods
)
```

Parameters

methods	Type: System.Collections.Generic.IEnumerable.IRpcMethod [▶ 2625]. The methods.
---------	---

Reference

[RpcMethodCollection Class](#) [▶ 2894]

[RpcMethodCollection Overload](#) [▶ 2909]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.127 RpcMethodNotSupportedException Class

Symbol Exception

Inheritance Hierarchy

System.Object
 System.Exception
[TwinCAT.AdsException](#) [▶ 61]
[TwinCAT.TypeSystem.SymbolException](#) [▶ 2933]
 TwinCAT.TypeSystem.RpcMethodNotSupportedException

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax


C#

```
[SerializableAttribute]
public class RpcMethodNotSupportedException : SymbolException
```











The [RpcMethodNotSupportedException](#) type exposes the following members.

Constructors









	Name	Description
	RpcMethodNotSupportedException(Int32, ISymbol) [▶ 2912]	Initializes a new instance of the RpcMethodNotSupportedException class.
	RpcMethodNotSupportedException(Ser	Initializes a new instance of the RpcMethodNotSupportedException class.

	Name	Description
	ializationInfo, StreamingContext) [▶ 2913]	
	RpcMethodNotSupportedException(String, ISymbol) [▶ 2913]	Initializes a new instance of the <code>RpcMethodNotSupportedException</code> class.


Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from <code>Exception</code> .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from <code>Exception</code> .)
	HResult	Gets or sets <code>HRESULT</code> , a coded numerical value that is assigned to a specific exception. (Inherited from <code>Exception</code> .)
	InnerException	Gets the <code>Exception</code> instance that caused the current exception. (Inherited from <code>Exception</code> .)
	InstancePath [▶ 2944]	Gets the instance path. (Inherited from SymbolException [▶ 2933].)
	Message	Gets a message that describes the current exception. (Inherited from <code>Exception</code> .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from <code>Exception</code> .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from <code>Exception</code> .)
	Symbol [▶ 2944]	Gets the symbol. (Inherited from SymbolException [▶ 2933].)
	TargetSite	Gets the method that throws the current exception. (Inherited from <code>Exception</code> .)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from <code>Object</code> .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from <code>Object</code> .)
	GetBaseException	When overridden in a derived class, returns the <code>Exception</code> that is the root cause of one or more subsequent exceptions. (Inherited from <code>Exception</code> .)
	GetHashCode	Serves as the default hash function. (Inherited from <code>Object</code> .)
	GetObjectData [▶ 2945]	When overridden in a derived class, sets the <code>SerializationInfo</code> with information about the exception. (Inherited from SymbolException [▶ 2933].)
	GetType	Gets the runtime type of the current instance. (Inherited from <code>Exception</code> .)
	MemberwiseClone	Creates a shallow copy of the current <code>Object</code> . (Inherited from <code>Object</code> .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from <code>Exception</code> .)




Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)

Reference

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.127.1 RpcMethodNotSupportedException Constructor**Overload List**

	Name	Description
	RpcMethodNotSupportedException(Int32, ISymbol) [► 2912]	Initializes a new instance of the RpcMethodNotSupportedException [► 2910] class.
	RpcMethodNotSupportedException(SerializationInfo, StreamingContext) [► 2913]	Initializes a new instance of the RpcMethodNotSupportedException [► 2910] class.
	RpcMethodNotSupportedException(String, ISymbol) [► 2913]	Initializes a new instance of the RpcMethodNotSupportedException [► 2910] class.

Reference

[RpcMethodNotSupportedException Class \[► 2910\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.127.1.1 RpcMethodNotSupportedException Constructor (Int32, ISymbol)

Initializes a new instance of the [RpcMethodNotSupportedException \[► 2910\]](#) class.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public RpcMethodNotSupportedException(
    int vTableIndex,
    ISymbol symbol
)
```

Parameters

vTableIndex	Type: System.Int32 Index of the v table.
-------------	---

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol.
--------	--

Reference

[RpcMethodNotSupportedException Class](#) [[▶ 2910](#)]

[RpcMethodNotSupportedException Overload](#) [[▶ 2912](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.127.1.2 RpcMethodNotSupportedException Constructor (SerializationInfo, StreamingContext)

Initializes a new instance of the [RpcMethodNotSupportedException](#) [[▶ 2910](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected RpcMethodNotSupportedException(
    SerializationInfo serializationInfo,
    StreamingContext streamingContext
)
```

Parameters

serializationInfo	Type: System.Runtime.Serialization.SerializationInfo The serialization information.
streamingContext	Type: System.Runtime.Serialization.StreamingContext The streaming context.

Exceptions

Exception	Condition
NotImplementedException	

Reference

[RpcMethodNotSupportedException Class](#) [[▶ 2910](#)]

[RpcMethodNotSupportedException Overload](#) [[▶ 2912](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.127.1.3 RpcMethodNotSupportedException Constructor (String, ISymbol)

Initializes a new instance of the [RpcMethodNotSupportedException](#) [[▶ 2910](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RpcMethodNotSupportedException(
    string methodName,
    ISymbol symbol
)
```

Parameters

methodName	Type: System.String Name of the method.
symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol.

Reference

[RpcMethodNotSupportedException Class \[▶ 2910\]](#)











[RpcMethodNotSupportedException Overload \[▶ 2912\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.127.2 RpcMethodNotSupportedException Properties

The [RpcMethodNotSupportedException \[▶ 2910\]](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	InstancePath [▶ 2944]	Gets the instance path. (Inherited from SymbolException [▶ 2933] .)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	Symbol [▶ 2944]	Gets the symbol. (Inherited from SymbolException [▶ 2933] .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Reference









[RpcMethodNotSupportedException Class \[▶ 2910\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.127.3 RpcMethodNotSupportedException Methods

The [RpcMethodNotSupportedException](#) [▶ 2910] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData [▶ 2945]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from SymbolException [▶ 2933].)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Reference


[RpcMethodNotSupportedException Class](#) [▶ 2910]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.127.4 RpcMethodNotSupportedException Events

The [RpcMethodNotSupportedException](#) [▶ 2910] type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)

Reference

[RpcMethodNotSupportedException Class](#) [▶ 2910]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.128 RpcMethodParameterCollection Class

Collection of RPC method parameters

Inheritance Hierarchy

System.Object
 TwinCAT.TypeSystem.RpcMethodParameterCollection

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#





```
public class RpcMethodParameterCollection : IRpcMethodParameterCollection,
    IList<IRpcMethodParameter>, ICollection<IRpcMethodParameter>, IEnumerable<IRpcMethodParameter>,
    IEnumerable
```

The RpcMethodParameterCollection type exposes the following members.











Constructors








	Name	Description
	RpcMethodParameterCollection . [▶ 2926]	Initializes a new instance of the RpcMethodParameterCollection class.
	RpcMethodParameterCollection(IEnumerable<IRpcMethodParameter>) [▶ 2927]	Initializes a new instance of the RpcMethodParameterCollection class.

Properties

	Name	Description
	Count [▶ 2917]	Gets the number of elements contained in the ICollection.T..
	Empty [▶ 2919]	Gets an Empty RpcMethodParameterCollection
	IsReadOnly [▶ 2918]	Gets a value indicating whether the ICollection.T. is read-only.
	Item [▶ 2918]	Gets or sets the element at the specified index.

Methods

	Name	Description
	Add [▶ 2920]	Adds an item to the ICollection.T..
	AsReadOnly [▶ 2921]	Returns a read only version of this RpcMethodParameterCollection
	Clear [▶ 2921]	Removes all items from the ICollection.T..
	Contains [▶ 2921]	Determines whether the ICollection.T. contains a specific value.
	CopyTo [▶ 2922]	Copies to.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 2923]	Returns an enumerator that iterates through the collection.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetLengthIsParameter [▶ 2923]	Gets the length is parameter.

	Name	Description
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 2924]	Determines the index of a specific item in the IList.T..
	Insert [▶ 2924]	Inserts an item to the IList.T. at the specified index.
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 2925]	Removes the first occurrence of a specific object from the ICollection.T..
	RemoveAt [▶ 2925]	Removes the IList.T. item at the specified index.
	ToString	Returns a string that represents the current object. (Inherited from Object.)





Reference

[TwinCAT.TypeSystem Namespace ▶ 2083](#)

6.11.128.1 RpcMethodParameterCollection Properties

The [RpcMethodParameterCollection ▶ 2915](#) type exposes the following members.

Properties

	Name	Description
	Count ▶ 2917	Gets the number of elements contained in the ICollection.T..
	Empty ▶ 2919	Gets an Empty RpcMethodParameterCollection ▶ 2915
	IsReadOnly ▶ 2918	Gets a value indicating whether the ICollection.T. is read-only.
	Item ▶ 2918	Gets or sets the element at the specified index.

Reference

[RpcMethodParameterCollection Class ▶ 2915](#)

[TwinCAT.TypeSystem Namespace ▶ 2083](#)

6.11.128.1.1 RpcMethodParameterCollection.Count Property

Gets the number of elements contained in the ICollection.T..

Namespace: [TwinCAT.TypeSystem ▶ 2083](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Count { get; }
```

Property Value

Type: Int32
The count.

Implements

ICollection.T..Count

Reference

[RpcMethodParameterCollection Class \[▸ 2915\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.128.1.2 RpcMethodParameterCollection.IsReadOnly Property

Gets a value indicating whether the ICollection.T. is read-only.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsReadOnly { get; }
```

Property Value

Type: Boolean

true if this instance is read only; otherwise, false.

Implements

ICollection.T..IsReadOnly

Reference

[RpcMethodParameterCollection Class \[▸ 2915\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.128.1.3 RpcMethodParameterCollection.Item Property

Gets or sets the element at the specified index.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IRpcMethodParameter this[  
    int index  
] { get; set; }
```

Parameters

index	Type: System.Int32 The index.
-------	----------------------------------

Return Value

Type: [IRpcMethodParameter](#) [[▸ 2635](#)]
[RpcMethodParameter](#).

Implements

[IList.T..Item.Int32](#).

Exceptions

Exception	Condition
NotImplementedException	

Reference

[RpcMethodParameterCollection Class](#) [[▸ 2915](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.128.1.4 RpcMethodParameterCollection.Empty Property

Gets an Empty [RpcMethodParameterCollection](#) [[▸ 2915](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: [TwinCAT.Ads](#) (in [TwinCAT.Ads.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public static RpcMethodParameterCollection Empty { get; }
```

Property Value

Type: [RpcMethodParameterCollection](#) [[▸ 2915](#)]
 The empty.

Reference





[RpcMethodParameterCollection Class](#) [[▸ 2915](#)]














[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.128.2 RpcMethodParameterCollection Methods

The [RpcMethodParameterCollection](#) [[▸ 2915](#)] type exposes the following members.

Methods

	Name	Description
	Add [▸ 2920]	Adds an item to the ICollection.T..
	AsReadOnly [▸ 2921]	Returns a read only version of this RpcMethodParameterCollection [▸ 2915]
	Clear [▸ 2921]	Removes all items from the ICollection.T..
	Contains [▸ 2921]	Determines whether the ICollection.T. contains a specific value.

	Name	Description
	CopyTo [▶ 2922]	Copies to.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 2923]	Returns an enumerator that iterates through the collection.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetLengthIsParameter [▶ 2923]	Gets the length is parameter.
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 2924]	Determines the index of a specific item in the IList.T..
	Insert [▶ 2924]	Inserts an item to the IList.T. at the specified index.
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 2925]	Removes the first occurrence of a specific object from the ICollection.T..
	RemoveAt [▶ 2925]	Removes the IList.T. item at the specified index.
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[RpcMethodParameterCollection Class](#) [[▶ 2915](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.128.2.1 RpcMethodParameterCollection.Add Method

Adds an item to the ICollection.T..

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Add(
    IRpcMethodParameter item
)
```

Parameters

item	Type: TwinCAT.TypeSystem.IRpcMethodParameter [▶ 2635] The object to add to the ICollection.T..
------	---

Implements

ICollection.T..Add(T)

Reference

[RpcMethodParameterCollection Class \[▸ 2915\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.128.2.2 **RpcMethodParameterCollection.AsReadOnly Method**

Returns a read only version of this [RpcMethodParameterCollection \[▸ 2915\]](#)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyMethodParameterCollection AsReadOnly()
```

Field Value

Type: [ReadOnlyMethodParameterCollection \[▸ 2852\]](#)

Collection as read only version.

Return Value

Type: [ReadOnlyMethodParameterCollection \[▸ 2852\]](#)

ReadOnlyMethodParameterCollection.

Reference

[RpcMethodParameterCollection Class \[▸ 2915\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.128.2.3 **RpcMethodParameterCollection.Clear Method**

Removes all items from the ICollection.T..

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Clear()
```

Implements

ICollection.T..Clear.

Reference

[RpcMethodParameterCollection Class \[▸ 2915\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.128.2.4 **RpcMethodParameterCollection.Contains Method**

Determines whether the ICollection.T. contains a specific value.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(
    IRpcMethodParameter item
)
```

Parameters

item	Type: TwinCAT.TypeSystem.IRpcMethodParameter [▶ 2635] The object to locate in the ICollection.T..
------	--

Return Value

Type: Boolean

true if item is found in the ICollection.T.; otherwise, false.

Implements

ICollection.T..Contains(T)

Reference

[RpcMethodParameterCollection Class](#) [[▶ 2915](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.128.2.5 RpcMethodParameterCollection.CopyTo Method

Copies to.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void CopyTo(
    IRpcMethodParameter[] array,
    int arrayIndex
)
```

Parameters

array	Type: .TwinCAT.TypeSystem.IRpcMethodParameter [▶ 2635]. The array.
arrayIndex	Type: System.Int32 Index of the array.

Implements

ICollection.T..CopyTo(.T., Int32)

Reference

[RpcMethodParameterCollection Class](#) [[▶ 2915](#)]

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.128.2.6 RpcMethodParameterCollection.GetEnumerator Method

Returns an enumerator that iterates through the collection.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IEnumerator<IRpcMethodParameter> GetEnumerator()
```

Return Value

Type: [IEnumerator.IRpcMethodParameter \[▸ 2635\]](#).

A [IEnumerator.T.](#) that can be used to iterate through the collection.

Implements

[IEnumerable.T..GetEnumerator.](#)

Reference

[RpcMethodParameterCollection Class \[▸ 2915\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.128.2.7 RpcMethodParameterCollection.GetLengthIsParameter Method

Gets the length is parameter.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IRpcMethodParameter? GetLengthIsParameter(  
    IRpcMethodParameter parameter  
)
```

Parameters

parameter	Type: TwinCAT.TypeSystem.IRpcMethodParameter [▸ 2635] The parameter.
-----------	---

Return Value

Type: [IRpcMethodParameter \[▸ 2635\]](#)

[IRpcMethodParameter.](#)

Implements

[IRpcMethodParameterCollection.GetLengthIsParameter\(IRpcMethodParameter\) \[▸ 2641\]](#)

Exceptions

Exception	Condition
ArgumentNullException	parameter
ArgumentException	Parameter is not contained in ParameterList - parameter

Reference

[RpcMethodParameterCollection Class \[► 2915\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.128.2.8 RpcMethodParameterCollection.IndexOf Method

Determines the index of a specific item in the IList.T..

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public int IndexOf(
    IRpcMethodParameter item
)
```

Parameters

item	Type: TwinCAT.TypeSystem.IRpcMethodParameter [► 2635] The object to locate in the IList.T..
------	--

Return Value

Type: Int32

The index of item if found in the list; otherwise, -1.

Implements

IList.T..IndexOf(T)

Reference

[RpcMethodParameterCollection Class \[► 2915\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.128.2.9 RpcMethodParameterCollection.Insert Method

Inserts an item to the IList.T. at the specified index.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public void Insert(
    int index,
    IRpcMethodParameter item
)
```

Parameters

index	Type: System.Int32 The zero-based index at which item should be inserted.
item	Type: TwinCAT.TypeSystem.IRpcMethodParameter [▶ 2635] The object to insert into the IList.T..

Implements

IList.T..Insert(Int32, T)

Reference

[RpcMethodParameterCollection Class](#) [▶ 2915]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.128.2.10 RpcMethodParameterCollection.Remove Method

Removes the first occurrence of a specific object from the ICollection.T..

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Remove(  
    IRpcMethodParameter item  
)
```

Parameters

item	Type: TwinCAT.TypeSystem.IRpcMethodParameter [▶ 2635] The object to remove from the ICollection.T..
------	--

Return Value

Type: Boolean

true if item was successfully removed from the ICollection.T.; otherwise, false. This method also returns false if item is not found in the original ICollection.T..

Implements

ICollection.T..Remove(T)

Reference

[RpcMethodParameterCollection Class](#) [▶ 2915]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.128.2.11 RpcMethodParameterCollection.RemoveAt Method

Removes the IList.T. item at the specified index.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void RemoveAt(
    int index
)
```

Parameters

index	Type: System.Int32 The zero-based index of the item to remove.
-------	---

Implements

IList.T..RemoveAt(Int32)

Reference

[RpcMethodParameterCollection Class](#) [► 2915]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.128.3 RpcMethodParameterCollection Constructor

Overload List

	Name	Description
	RpcMethodParameterCollection. [► 2926]	Initializes a new instance of the RpcMethodParameterCollection [► 2915] class.
	RpcMethodParameterCollection(IEnumerable.IRpcMethodParameter.) [► 2927]	Initializes a new instance of the RpcMethodParameterCollection [► 2915] class.

Reference

[RpcMethodParameterCollection Class](#) [► 2915]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.128.3.1 RpcMethodParameterCollection Constructor

Initializes a new instance of the [RpcMethodParameterCollection](#) [► 2915] class.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RpcMethodParameterCollection()
```

Reference

[RpcMethodParameterCollection Class](#) [► 2915]

[RpcMethodParameterCollection Overload \[▸ 2926\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.128.3.2 RpcMethodParameterCollection Constructor (IEnumerable.IRpcMethodParameter.)

Initializes a new instance of the [RpcMethodParameterCollection \[▸ 2915\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RpcMethodParameterCollection(
    IEnumerable<IRpcMethodParameter> coll
)
```

Parameters

coll	Type: System.Collections.Generic.IEnumerable. IRpcMethodParameter [▸ 2635] . The coll.
------	---

Reference

[RpcMethodParameterCollection Class \[▸ 2915\]](#)

[RpcMethodParameterCollection Overload \[▸ 2926\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.129 StringConvertMode Enumeration

Enum StringConvertMode

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum StringConvertMode
```

Members

	Member name	Value	Description
	FixedLength	0	Fixed Length String
	FixedLengthZeroTerminated	1	Fixed Length String that can be terminated with '\0'
	ZeroTerminated	2	'\0' terminated dynamic length string
	LengthPrefix	3	Length Prefix (number of following bytes as uint 4-Byte)

Remarks

The StringConvertMode is used to specify, how Strings will be marshalled / demarshalled.

Reference

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

StringMarshaler

6.11.130 SymbolAccessRights Enumeration

Enum specifying Access Rights to symbols

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
[FlagsAttribute]
public enum SymbolAccessRights
```

Members

	Member name	Value	Description
	None	0	None / Uninitialized
	Read	1	Read-Access
	Write	2	Write-Access
	MethodInvoke	4	Right to Invoke Methods / RPC Invoke
	ReadWrite	3	Read / Write Access
	All	7	Full Access

Reference

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.131 SymbolCollection Class

Interface represents a collection of [ISymbol \[► 2691\]](#) objects.

Inheritance Hierarchy

System.Object

[TwinCAT.TypeSystem.Generic.InstanceCollection \[► 3104\].ISymbol \[► 2691\].](#)

[TwinCAT.TypeSystem.Generic.SymbolCollection \[► 3165\].ISymbol \[► 2691\].](#)

[TwinCAT.TypeSystem.SymbolCollection](#)

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)









Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**















```
public class SymbolCollection : SymbolCollection<ISymbol>,
    ISymbolCollection, ISymbolCollection<ISymbol>, IInstanceCollection<ISymbol>,
    IList<ISymbol>, ICollection<ISymbol>, IEnumerable<ISymbol>,
    IEnumerable
```












The SymbolCollection type exposes the following members.

Properties

	Name	Description
	Count [▶ 3107]	Gets the collection count. (Inherited from InstanceCollection.T. [▶ 3104].)
	Empty [▶ 2931]	Returns an Empty Collection.
	InnerList [▶ 3108]	Gets the List of instances. (Inherited from InstanceCollection.T. [▶ 3104].)
	InnerPathDict [▶ 3108]	The Path dictionary (Inherited from InstanceCollection.T. [▶ 3104].)
	IsReadOnly [▶ 3109]	Gets a value indicating whether this instance is read only. (Inherited from InstanceCollection.T. [▶ 3104].)
	Item.Int32. [▶ 3109]	Gets or sets the Instance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	Item.String. [▶ 3110]	Gets the Instance [▶ 2549] with the specified instance path. (Inherited from InstanceCollection.T. [▶ 3104].)
	Mode [▶ 3111]	The mode this InstanceCollection.T. [▶ 3104] is working in. (Inherited from InstanceCollection.T. [▶ 3104].)

Methods

	Name	Description
	Add [▶ 3112]	Adds the specified item. (Inherited from InstanceCollection.T. [▶ 3104].)
	AddRange [▶ 3113]	Adds the specified items to this collection. (Inherited from InstanceCollection.T. [▶ 3104].)
	AsReadOnly [▶ 2932]	Returns a Read only version of this collection (shallow copy).
	Clear [▶ 3114]	Clears this instance. (Inherited from InstanceCollection.T. [▶ 3104].)
	Clone [▶ 2933]	Clones this instance.
	Contains(T) [▶ 3115]	Determines whether this collection contains the specified Instance [▶ 2549] (Inherited from InstanceCollection.T. [▶ 3104].)
	Contains(String) [▶ 3114]	Determines whether this collection contains an Instance [▶ 2549] with the specified InstanceName / InstancePath (Inherited from InstanceCollection.T. [▶ 3104].)
	ContainsName [▶ 3116]	Determines whether the specified instance name contains name. (Inherited from InstanceCollection.T. [▶ 3104].)
	CopyTo [▶ 3116]	Copies this InstanceCollection.T. [▶ 3104] to the specified array. (Inherited from InstanceCollection.T. [▶ 3104].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 3117]	Gets the enumerator. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetInstance [▶ 3117]	Gets the Instance [▶ 2549]by instance path. (Inherited from InstanceCollection.T. [▶ 3104].)

	Name	Description
	GetInstanceByName [▶ 3118]	Gets the name of the instance by. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 3119]	Determines the index of the specified IInstance [▶ 2549]. (Inherited from InstanceCollection.T. [▶ 3104].)
	Insert [▶ 3119]	Inserts the specified IInstance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 3120]	Removes the specified item. (Inherited from InstanceCollection.T. [▶ 3104].)
	RemoveAt [▶ 3121]	Removes the IInstance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInstance [▶ 3121]	Tries to get the IInstance [▶ 2549]. of the specified path. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetInstanceByName [▶ 3122]	Tries to get Instances by name. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetInstances [▶ 3170]	Try to get instances with predicate function (Inherited from SymbolCollection.T. [▶ 3165].)









Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.131.1 SymbolCollection Properties

The [SymbolCollection](#) [[▶ 2928](#)] type exposes the following members.

Properties

	Name	Description
	Count [▶ 3107]	Gets the collection count. (Inherited from InstanceCollection.T. [▶ 3104].)
	Empty [▶ 2931]	Returns an Empty Collection.
	InnerList [▶ 3108]	Gets the List of instances. (Inherited from InstanceCollection.T. [▶ 3104].)
	InnerPathDict [▶ 3108]	The Path dictionary (Inherited from InstanceCollection.T. [▶ 3104].)
	IsReadOnly [▶ 3109]	Gets a value indicating whether this instance is read only. (Inherited from InstanceCollection.T. [▶ 3104].)
	Item.Int32. [▶ 3109]	Gets or sets the IInstance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	Item.String. [▶ 3110]	Gets the IInstance [▶ 2549] with the specified instance path. (Inherited from InstanceCollection.T. [▶ 3104].)
	Mode [▶ 3111]	The mode this InstanceCollection.T. [▶ 3104] is working in. (Inherited from InstanceCollection.T. [▶ 3104].)

Reference

[SymbolCollection Class \[▶ 2928\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.131.1 SymbolCollection.Empty Property

Returns an Empty Collection.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static SymbolCollection Empty { get; }
```

Return Value

Type: [SymbolCollection \[▶ 2928\]](#)

SymbolCollection.

Reference











[SymbolCollection Class \[▶ 2928\]](#)
















[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.131.2 SymbolCollection Methods

The [SymbolCollection \[▶ 2928\]](#) type exposes the following members.

Methods

	Name	Description
	Add [▶ 3112]	Adds the specified item. (Inherited from InstanceCollection.T. [▶ 3104].)
	AddRange [▶ 3113]	Adds the specified items to this collection. (Inherited from InstanceCollection.T. [▶ 3104].)
	AsReadOnly [▶ 2932]	Returns a Read only version of this collection (shallow copy).
	Clear [▶ 3114]	Clears this instance. (Inherited from InstanceCollection.T. [▶ 3104].)
	Clone [▶ 2933]	Clones this instance.
	Contains(T) [▶ 3115]	Determines whether this collection contains the specified Instance [▶ 2549] (Inherited from InstanceCollection.T. [▶ 3104].)
	Contains(String) [▶ 3114]	Determines whether this collection contains an Instance [▶ 2549] with the specified InstanceName / InstancePath (Inherited from InstanceCollection.T. [▶ 3104].)
	ContainsName [▶ 3116]	Determines whether the specified instance name contains name. (Inherited from InstanceCollection.T. [▶ 3104].)
	CopyTo [▶ 3116]	Copies this InstanceCollection.T. [▶ 3104] to the specified array. (Inherited from InstanceCollection.T. [▶ 3104].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)

	Name	Description
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 3117]	Gets the enumerator. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetInstance [▶ 3117]	Gets the Instance [▶ 2549] by instance path. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetInstanceByName [▶ 3118]	Gets the name of the instance by. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 3119]	Determines the index of the specified Instance [▶ 2549]. (Inherited from InstanceCollection.T. [▶ 3104].)
	Insert [▶ 3119]	Inserts the specified Instance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 3120]	Removes the specified item. (Inherited from InstanceCollection.T. [▶ 3104].)
	RemoveAt [▶ 3121]	Removes the Instance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInstance [▶ 3121]	Tries to get the Instance [▶ 2549]. of the specified path. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetInstanceByName [▶ 3122]	Tries to get Instances by name. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetInstances [▶ 3170]	Try to get instances with predicate function (Inherited from SymbolCollection.T. [▶ 3165].)

Reference

[SymbolCollection Class](#) [▶ 2928]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.131.2.1 SymbolCollection.AsReadOnly Method

Returns a Read only version of this collection (shallow copy).

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlySymbolCollection AsReadOnly()
```

Return Value

Type: [ReadOnlySymbolCollection](#) [▶ 2861]

Read only collection.

Reference

[SymbolCollection Class](#) [[▶ 2928](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.131.2.2 SymbolCollection.Clone Method

Clones this instance.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolCollection Clone()
```

Return Value

Type: [SymbolCollection](#) [[▶ 2928](#)]

Cloned [SymbolCollection](#) [[▶ 2928](#)].

Reference

[SymbolCollection Class](#) [[▶ 2928](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.131.2.3 SymbolCollection.Empty Method

Returns an Empty Collection.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public static SymbolCollection Empty()
```

Return Value

Type: [SymbolCollection](#) [[▶ 2928](#)]

[SymbolCollection](#).

Reference

[SymbolCollection Class](#) [[▶ 2928](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.132 SymbolException Class

Symbol bound exceptions

Inheritance Hierarchy

System.Object

System.Exception

TwinCAT.AdsException [▶ 61]

TwinCAT.TypeSystem.SymbolException

TwinCAT.TypeSystem.CannotAccessVirtualSymbolException [▶ 2103]

TwinCAT.TypeSystem.InsufficientAccessRightsException [▶ 2573]

TwinCAT.TypeSystem.RpcInvokeException [▶ 2887]

TwinCAT.TypeSystem.RpcMethodNotSupportedException [▶ 2910]

TwinCAT.ValueAccess.CannotAccessValueException [▶ 3181]

Namespace: TwinCAT.TypeSystem [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229








Syntax






C#

```
[SerializableAttribute]
public class SymbolException : AdsException
```











The SymbolException type exposes the following members.

Constructors



	Name	Description
	SymbolException(ISymbol) [▶ 2937]	Initializes a new instance of the SymbolException class.
	SymbolException(ISymbol, Exception) [▶ 2939]	Initializes a new instance of the SymbolException class.
	SymbolException(ISymbol, Int32) [▶ 2939]	Initializes a new instance of the SymbolException class.
	SymbolException(SerializationInfo, StreamingContext) [▶ 2938]	Initializes a new instance of the SymbolException class.
	SymbolException(String, ISymbol) [▶ 2938]	Initializes a new instance of the SymbolException class.
	SymbolException(ISymbol, Int32, Exception) [▶ 2941]	Initializes a new instance of the SymbolException class.
	SymbolException(String, String, Exception) [▶ 2942]	Initializes a new instance of the SymbolException class.







	Name	Description
	SymbolException(String, String, Int32) [▶ 2942]	Initializes a new instance of the SymbolException class.
	SymbolException(String, ISymbolException) [▶ 2940]	Initializes a new instance of the SymbolException class.
	SymbolException(String, ISymbol, Int32) [▶ 2940]	Initializes a new instance of the SymbolException class.
	SymbolException(String, String, Int32, Exception) [▶ 2943]	Initializes a new instance of the SymbolException class.
	SymbolException(String, ISymbol, Int32, Exception) [▶ 2941]	Initializes a new instance of the SymbolException class.

Properties


	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	InstancePath [▶ 2944]	Gets the instance path.
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	Symbol [▶ 2944]	Gets the symbol.
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)

	Name	Description
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData [▶ 2945]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Overrides Exception.GetObjectData(SerializationInfo, StreamingContext).)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Events







	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)







Reference

TwinCAT.TypeSystem Namespace [▶ 2083]

6.11.132.1 SymbolException Constructor

Overload List

	Name	Description
	SymbolException(ISymbol) [▶ 2937]	Initializes a new instance of the SymbolException [▶ 2933] class.
	SymbolException(ISymbol, Exception) [▶ 2939]	Initializes a new instance of the SymbolException [▶ 2933] class.
	SymbolException(ISymbol, Int32) [▶ 2939]	Initializes a new instance of the SymbolException [▶ 2933] class.
	SymbolException(SerializationInfo, StreamingContext) [▶ 2938]	Initializes a new instance of the SymbolException [▶ 2933] class.
	SymbolException(String, ISymbol) [▶ 2938]	Initializes a new instance of the SymbolException [▶ 2933] class.
	SymbolException(ISymbol, Int32, Exception) [▶ 2941]	Initializes a new instance of the SymbolException [▶ 2933] class.

	Name	Description
	SymbolException(String, String, Exception) [▸ 2942]	Initializes a new instance of the SymbolException [▸ 2933] class.
	SymbolException(String, String, Int32) [▸ 2942]	Initializes a new instance of the SymbolException [▸ 2933] class.
	SymbolException(String, ISymbol, Exception) [▸ 2940]	Initializes a new instance of the SymbolException [▸ 2933] class.
	SymbolException(String, ISymbol, Int32) [▸ 2940]	Initializes a new instance of the SymbolException [▸ 2933] class.
	SymbolException(String, String, Int32, Exception) [▸ 2943]	Initializes a new instance of the SymbolException [▸ 2933] class.
	SymbolException(String, ISymbol, Int32, Exception) [▸ 2941]	Initializes a new instance of the SymbolException [▸ 2933] class.

Reference

[SymbolException Class](#) [[▸ 2933](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.132.1.1 SymbolException Constructor (ISymbol)

Initializes a new instance of the [SymbolException](#) [[▸ 2933](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolException(
    ISymbol symbol
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▸ 2691] The symbol.
--------	--

Reference

[SymbolException Class](#) [[▸ 2933](#)]

[SymbolException Overload](#) [[▸ 2936](#)]

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.132.1.2 SymbolException Constructor (SerializationInfo, StreamingContext)

Initializes a new instance of the [SymbolException \[▸ 2933\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected SymbolException(
    SerializationInfo serializationInfo,
    StreamingContext streamingContext
)
```

Parameters

serializationInfo	Type: System.Runtime.Serialization.SerializationInfo The serialization information.
streamingContext	Type: System.Runtime.Serialization.StreamingContext The streaming context.

Reference

[SymbolException Class \[▸ 2933\]](#)

[SymbolException Overload \[▸ 2936\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.132.1.3 SymbolException Constructor (String, ISymbol)

Initializes a new instance of the [SymbolException \[▸ 2933\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolException(
    string message,
    ISymbol? symbol
)
```

Parameters

message	Type: System.String The message.
symbol	Type: TwinCAT.TypeSystem.ISymbol [▸ 2691] The symbol.

Reference

[SymbolException Class \[▸ 2933\]](#)

[SymbolException Overload \[▸ 2936\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.132.1.4 SymbolException Constructor (ISymbol, Exception)

Initializes a new instance of the [SymbolException](#) [[▶ 2933](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolException(  
    ISymbol symbol,  
    Exception innerException  
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol.
innerException	Type: System.Exception The inner exception.

Reference

[SymbolException Class](#) [[▶ 2933](#)]

[SymbolException Overload](#) [[▶ 2936](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.132.1.5 SymbolException Constructor (ISymbol, Int32)

Initializes a new instance of the [SymbolException](#) [[▶ 2933](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolException(  
    ISymbol symbol,  
    int errorCode  
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol.
errorCode	Type: System.Int32 The error code.

Reference

[SymbolException Class](#) [[▶ 2933](#)]

[SymbolException Overload](#) [[▶ 2936](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.132.1.6 SymbolException Constructor (String, ISymbol, Exception)

Initializes a new instance of the [SymbolException](#) [▸ 2933] class.

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolException(
    string message,
    ISymbol? symbol,
    Exception? innerException
)
```

Parameters

message	Type: System.String The message.
symbol	Type: TwinCAT.TypeSystem.ISymbol [▸ 2691] The symbol.
innerException	Type: System.Exception The inner exception.

Reference

[SymbolException Class](#) [▸ 2933]

[SymbolException Overload](#) [▸ 2936]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

6.11.132.1.7 SymbolException Constructor (String, ISymbol, Int32)

Initializes a new instance of the [SymbolException](#) [▸ 2933] class.

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolException(
    string message,
    ISymbol? symbol,
    int errorCode
)
```

Parameters

message	Type: System.String The message.
symbol	Type: TwinCAT.TypeSystem.ISymbol [▸ 2691] The symbol.
errorCode	Type: System.Int32 The error code.

Reference

[SymbolException Class](#) [▸ 2933]

[SymbolException Overload \[▸ 2936\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.132.1.8 SymbolException Constructor (ISymbol, Int32, Exception)

Initializes a new instance of the [SymbolException \[▸ 2933\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolException(
    ISymbol symbol,
    int errorCode,
    Exception innerException
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▸ 2691] The symbol.
errorCode	Type: System.Int32 The error code.
innerException	Type: System.Exception The inner exception.

Reference

[SymbolException Class \[▸ 2933\]](#)

[SymbolException Overload \[▸ 2936\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.132.1.9 SymbolException Constructor (String, ISymbol, Int32, Exception)

Initializes a new instance of the [SymbolException \[▸ 2933\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolException(
    string message,
    ISymbol? symbol,
    int errorCode,
    Exception? innerException
)
```

Parameters

message	Type: System.String The message.
symbol	Type: TwinCAT.TypeSystem.ISymbol [▸ 2691] The symbol.

errorCode	Type: System.Int32 The error code.
innerException	Type: System.Exception The inner exception.

Reference

[SymbolException Class \[▸ 2933\]](#)

[SymbolException Overload \[▸ 2936\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.132.1.10 SymbolException Constructor (String, String, Exception)

Initializes a new instance of the [SymbolException \[▸ 2933\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolException(
    string message,
    string instancePath,
    Exception? innerException
)
```

Parameters

message	Type: System.String The message.
instancePath	Type: System.String The symbol/instance path.
innerException	Type: System.Exception The inner exception.

Reference

[SymbolException Class \[▸ 2933\]](#)

[SymbolException Overload \[▸ 2936\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.132.1.11 SymbolException Constructor (String, String, Int32)

Initializes a new instance of the [SymbolException \[▸ 2933\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolException(
    string message,
    string instancePath,
    int errorCode
)
```

Parameters

message	Type: System.String The message.
instancePath	Type: System.String The symbol/instance path.
errorCode	Type: System.Int32 The error code.

Reference

[SymbolException Class \[▸ 2933\]](#)

[SymbolException Overload \[▸ 2936\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.132.1.12 SymbolException Constructor (String, String, Int32, Exception)

Initializes a new instance of the [SymbolException \[▸ 2933\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public SymbolException(
    string message,
    string instancePath,
    int errorCode,
    Exception? innerException
)
```

Parameters

message	Type: System.String The message.
instancePath	Type: System.String The instance path.
errorCode	Type: System.Int32 The error code.
innerException	Type: System.Exception The inner exception.

Reference

[SymbolException Class \[▸ 2933\]](#)











[SymbolException Overload \[▸ 2936\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.132.2 SymbolException Properties

The [SymbolException \[▸ 2933\]](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	InstancePath [▶ 2944]	Gets the instance path.
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	Symbol [▶ 2944]	Gets the symbol.
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Reference

[SymbolException Class \[▶ 2933\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.132.2.1 SymbolException.InstancePath Property

Gets the instance path.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string InstancePath { get; }
```

Property Value

Type: String

The instance path.

Reference

[SymbolException Class \[▶ 2933\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.132.2.2 SymbolException.Symbol Property

Gets the symbol.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISymbol? Symbol { get; }
```

Property Value

Type: [ISymbol](#) [[▸ 2691](#)]

The symbol.

Reference









[SymbolException Class](#) [[▸ 2933](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.132.3 SymbolException Methods

The [SymbolException](#) [[▸ 2933](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData [▸ 2945]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Overrides Exception.GetObjectData(SerializationInfo, StreamingContext).)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Reference

[SymbolException Class](#) [[▸ 2933](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.132.3.1 SymbolException.GetObjectData Method

When overridden in a derived class, sets the SerializationInfo with information about the exception.

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override void GetObjectData(
    SerializationInfo info,
    StreamingContext context
)
```

Parameters

info	Type: System.Runtime.Serialization.SerializationInfo The SerializationInfo that holds the serialized object data about the exception being thrown.
context	Type: System.Runtime.Serialization.StreamingContext The StreamingContext that contains contextual information about the source or destination.

Implements

ISerializable.GetObjectData(SerializationInfo, StreamingContext)

Exceptions

Exception	Condition
ArgumentNullException	info

Reference


[SymbolException Class \[► 2933\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.132.4 SymbolException Events

The [SymbolException \[► 2933\]](#) type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)

Reference

[SymbolException Class \[► 2933\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.133 TypeAttribute Class

ADS Attribute

Inheritance Hierarchy

System.Object
 TwinCAT.TypeSystem.TypeAttribute

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax

C#







```
public class TypeAttribute : ITypeAttribute
```

The TypeAttribute type exposes the following members.





Properties

	Name	Description
	Name [▸ 2948]	Name of the Attribute
	Value [▸ 2948]	Gets the value of the attribute

Methods

	Name	Description
	Equals [▸ 2949]	Equals (Overrides Object.Equals(Object).)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode [▸ 2950]	Gets the GetHashCode of the Address (Overrides Object.GetHashCode..)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Operators

	Name	Description
 	Equality [▸ 2950]	Operator==
 	Inequality [▸ 2951]	Implements the != operator.



Reference

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.133.1 TypeAttribute Properties

The [TypeAttribute \[▸ 2946\]](#) type exposes the following members.

Properties

	Name	Description
	Name [▶ 2948]	Name of the Attribute
	Value [▶ 2948]	Gets the value of the attribute

Reference

[TypeAttribute Class](#) [[▶ 2946](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.133.1.1 TypeAttribute.Name Property

Name of the Attribute

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Name { get; }
```

Property Value

Type: String

The name.

Implements

[ITypeAttribute.Name](#) [[▶ 2725](#)]

Reference

[TypeAttribute Class](#) [[▶ 2946](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.133.1.2 TypeAttribute.Value Property

Gets the value of the attribute

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Value { get; }
```

Property Value







Type: String

The value.

Implements[ITypeAttribute.Value](#) [[▶ 2726](#)]**Reference**[TypeAttribute Class](#) [[▶ 2946](#)][TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]**6.11.133.2 TypeAttribute Methods**

The [TypeAttribute](#) [[▶ 2946](#)] type exposes the following members.

Methods

	Name	Description
	Equals [▶ 2949]	Equals (Overrides Object.Equals(Object).)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode [▶ 2950]	Gets the GetHashCode of the Address (Overrides Object.GetHashCode..)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference[TypeAttribute Class](#) [[▶ 2946](#)][TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]**6.11.133.2.1 TypeAttribute.Equals Method**

Equals

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public override bool Equals(
    Object? obj
)
```

Parameters

obj	Type: System.Object The object to compare with the current object.
-----	---

Return Value

Type: Boolean
true if the specified Object is equal to this instance; otherwise, false.

Reference[TypeAttribute Class](#) [► 2946][TwinCAT.TypeSystem Namespace](#) [► 2083]**6.11.133.2.2 TypeAttribute.GetHashCode Method**

Gets the HashCode of the Address

Namespace: [TwinCAT.TypeSystem](#) [► 2083]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**





```
public override int GetHashCode()
```

Return Value

Type: Int32

A hash code for this instance, suitable for use in hashing algorithms and data structures like a hash table.

Reference[TypeAttribute Class](#) [► 2946][TwinCAT.TypeSystem Namespace](#) [► 2083]**6.11.133.3 TypeAttribute Operators**The [TypeAttribute](#) [► 2946] type exposes the following members.**Operators**

	Name	Description
 	Equality [► 2950]	Operator==
 	Inequality [► 2951]	Implements the != operator.

Reference[TypeAttribute Class](#) [► 2946][TwinCAT.TypeSystem Namespace](#) [► 2083]**6.11.133.3.1 TypeAttribute.Equality Operator**

Operator==

Namespace: [TwinCAT.TypeSystem](#) [► 2083]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool operator ==(
    TypeAttribute? o1,
    TypeAttribute? o2
)
```

Parameters

o1	Type: TwinCAT.TypeSystem.TypeAttribute [▶ 2946] The o1.
o2	Type: TwinCAT.TypeSystem.TypeAttribute [▶ 2946] The o2.

Return Value

Type: Boolean
The result of the operator.

Reference

[TypeAttribute Class](#) [[▶ 2946](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.133.3.2 TypeAttribute.Inequality Operator

Implements the != operator.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool operator !=(
    TypeAttribute? o1,
    TypeAttribute? o2
)
```

Parameters

o1	Type: TwinCAT.TypeSystem.TypeAttribute [▶ 2946] The o1.
o2	Type: TwinCAT.TypeSystem.TypeAttribute [▶ 2946] The o2.

Return Value

Type: Boolean
The result of the operator.

Reference

[TypeAttribute Class](#) [[▶ 2946](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.134 TypeAttributeCollection Class

Collection of [AdsAttributes](#) [[▶ 2725](#)]

Inheritance Hierarchy

System.Object

 TwinCAT.TypeSystem.TypeAttributeCollection

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229



Syntax

C#






```
public class TypeAttributeCollection : ITypeAttributeCollection,
    IList<ITypeAttribute>, ICollection<ITypeAttribute>, IEnumerable<ITypeAttribute>,
    IEnumerable
```

The TypeAttributeCollection type exposes the following members.





Constructors


















	Name	Description
	TypeAttributeCollection. [▶ 2954]	Initializes a new instance of the TypeAttributeCollection class.
	TypeAttributeCollection(IEnumerable.ITypeAttribute.) [▶ 2954]	Initializes a new instance of the TypeAttributeCollection class.

Properties

	Name	Description
	Count [▶ 2955]	Gets the number of elements contained in the ICollection.T..
	Empty [▶ 2957]	Returns an Empty TypeAttributeCollection.
	IsReadOnly [▶ 2955]	Gets a value indicating whether the ICollection.T. is read-only.
	Item.Int32. [▶ 2956]	Gets or sets the element at the specified index.
	Item.String. [▶ 2957]	Gets the String with the specified name.

Methods

	Name	Description
	Add [▶ 2959]	Adds an item to the ICollection.T..
	AddRange [▶ 2959]	Adds the range.
	AsReadOnly [▶ 2960]	Gets a read only version of this TypeAttributeCollection
	Clear [▶ 2960]	Removes all items from the ICollection.T..



	Name	Description
	Contains(ITypeAttribute) [▶ 2961]	Determines whether the ICollection.T. contains a specific value.
	Contains(String) [▶ 2961]	Determines whether this TypeAttributeCollection contains the ITypeAttribute [▶ 2725] with the specified name.
	CopyTo [▶ 2962]	Copies to.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 2963]	Returns an enumerator that iterates through the collection.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 2964]	Determines the index of a specific item in the IList.T..
	Insert [▶ 2964]	Inserts an item to the IList.T. at the specified index.
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove(ITypeAttribute) [▶ 2966]	Removes the first occurrence of a specific object from the ICollection.T..
	Remove(String) [▶ 2965]	Removes the specified ITypeAttribute [▶ 2725] from the TypeAttributeCollection
	RemoveAt [▶ 2966]	Removes the IList.T. item at the specified index.
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetAttribute [▶ 2967]	Tries to get the specified ITypeAttribute [▶ 2725]
	TryGetValue [▶ 2967]	Tries to get the specified Attribute value.

Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.134.1 TypeAttributeCollection Constructor

Overload List

	Name	Description
	TypeAttributeCollection. [▶ 2954]	Initializes a new instance of the TypeAttributeCollection [▶ 2952] class.
	TypeAttributeCollection(IEnumerable.ITypeAttribute.) [▶ 2954]	Initializes a new instance of the TypeAttributeCollection [▶ 2952] class.

Reference[TypeAttributeCollection Class \[► 2952\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.134.1.1 TypeAttributeCollection Constructor**

Initializes a new instance of the [TypeAttributeCollection \[► 2952\]](#) class.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public TypeAttributeCollection()
```

Reference[TypeAttributeCollection Class \[► 2952\]](#)[TypeAttributeCollection Overload \[► 2953\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.134.1.2 TypeAttributeCollection Constructor (IEnumerable.ITypeAttribute.)**

Initializes a new instance of the [TypeAttributeCollection \[► 2952\]](#) class.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public TypeAttributeCollection(
    IEnumerable<ITypeAttribute> coll
)
```






Parameters

coll	Type: System.Collections.Generic.IEnumerable.ITypeAttribute [► 2725]. The coll.
------	--

Reference[TypeAttributeCollection Class \[► 2952\]](#)[TypeAttributeCollection Overload \[► 2953\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.134.2 TypeAttributeCollection Properties**

The [TypeAttributeCollection \[► 2952\]](#) type exposes the following members.

Properties

	Name	Description
	Count [▶ 2955]	Gets the number of elements contained in the ICollection.T..
	Empty [▶ 2957]	Returns an Empty TypeAttributeCollection [▶ 2952].
	IsReadOnly [▶ 2955]	Gets a value indicating whether the ICollection.T. is read-only.
	Item.Int32. [▶ 2956]	Gets or sets the element at the specified index.
	Item.String. [▶ 2957]	Gets the String with the specified name.

Reference

[TypeAttributeCollection Class](#) [[▶ 2952](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.134.2.1 TypeAttributeCollection.Count Property

Gets the number of elements contained in the ICollection.T..

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Count { get; }
```

Property Value

Type: Int32

The count.

Implements

ICollection.T..Count

Reference

[TypeAttributeCollection Class](#) [[▶ 2952](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.134.2.2 TypeAttributeCollection.IsReadOnly Property

Gets a value indicating whether the ICollection.T. is read-only.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsReadOnly { get; }
```

Property Value



Type: Boolean

true if this instance is read only; otherwise, false.

Implements

ICollection.T.IsReadOnly

Reference[TypeAttributeCollection Class \[▸ 2952\]](#)[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)**6.11.134.2.3 TypeAttributeCollection.Item Property****Overload List**

	Name	Description
	Item.Int32. [▸ 2956]	Gets or sets the element at the specified index.
	Item.String. [▸ 2957]	Gets the String with the specified name.

Reference[TypeAttributeCollection Class \[▸ 2952\]](#)[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)**6.11.134.2.3.1 TypeAttributeCollection.Item Property (Int32)**

Gets or sets the element at the specified index.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public ITypeAttribute this[
    int index
] { get; set; }
```

Parameters

index	Type: System.Int32 The index.
-------	----------------------------------

Return ValueType: [ITypeAttribute \[▸ 2725\]](#)

AdsAttribute.

Implements

IList.T.Item.Int32.

Reference[TypeAttributeCollection Class \[► 2952\]](#)[Item Overload \[► 2956\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.134.2.3.2 TypeAttributeCollection.Item Property (String)**

Gets the String with the specified name.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public string this[
    string name
] { get; }
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: String
System.String.

Implements[ITypeAttributeCollection.Item.String. \[► 2728\]](#)**Exceptions**

Exception	Condition
KeyNotFoundException	

Reference[TypeAttributeCollection Class \[► 2952\]](#)[Item Overload \[► 2956\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.134.2.4 TypeAttributeCollection.Empty Property**

Returns an Empty [TypeAttributeCollection \[► 2952\]](#).

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static TypeAttributeCollection Empty { get; }
```

Return Value

Type: [TypeAttributeCollection](#) [[▶ 2952](#)]
 TypeAttributeCollection.

Reference




















[TypeAttributeCollection Class](#) [[▶ 2952](#)]



[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.134.3 TypeAttributeCollection Methods

The [TypeAttributeCollection](#) [[▶ 2952](#)] type exposes the following members.

Methods

	Name	Description
	Add [▶ 2959]	Adds an item to the ICollection.T..
	AddRange [▶ 2959]	Adds the range.
	AsReadOnly [▶ 2960]	Gets a read only version of this TypeAttributeCollection [▶ 2952]
	Clear [▶ 2960]	Removes all items from the ICollection.T..
	Contains(ITypeAttribute) [▶ 2961]	Determines whether the ICollection.T. contains a specific value.
	Contains(String) [▶ 2961]	Determines whether this TypeAttributeCollection [▶ 2952] contains the ITypeAttribute [▶ 2725] with the specified name.
	CopyTo [▶ 2962]	Copies to.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 2963]	Returns an enumerator that iterates through the collection.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 2964]	Determines the index of a specific item in the IList.T..
	Insert [▶ 2964]	Inserts an item to the IList.T. at the specified index.
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove(ITypeAttribute) [▶ 2966]	Removes the first occurrence of a specific object from the ICollection.T..
	Remove(String) [▶ 2965]	Removes the specified ITypeAttribute [▶ 2725] from the TypeAttributeCollection [▶ 2952]
	RemoveAt [▶ 2966]	Removes the IList.T. item at the specified index.
	ToString	Returns a string that represents the current object. (Inherited from Object.)

	Name	Description
	TryGetAttribute [▶ 2967]	Tries to get the specified ITypeAttribute [▶ 2725]
	TryGetValue [▶ 2967]	Tries to get the specified Attribute value.

Reference

[TypeAttributeCollection Class](#) [\[▶ 2952\]](#)

[TwinCAT.TypeSystem Namespace](#) [\[▶ 2083\]](#)

6.11.134.3.1 TypeAttributeCollection.Add Method

Adds an item to the ICollection.T..

Namespace: [TwinCAT.TypeSystem](#) [\[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Add(
    ITypeAttribute item
)
```

Parameters

item	Type: TwinCAT.TypeSystem.ITypeAttribute [▶ 2725] The object to add to the ICollection.T..
------	--

Implements

ICollection.T..Add(T)

Reference

[TypeAttributeCollection Class](#) [\[▶ 2952\]](#)

[TwinCAT.TypeSystem Namespace](#) [\[▶ 2083\]](#)

6.11.134.3.2 TypeAttributeCollection.AddRange Method

Adds the range.

Namespace: [TwinCAT.TypeSystem](#) [\[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void AddRange(
    IEnumerable<ITypeAttribute> items
)
```

Parameters

items	Type: System.Collections.Generic.IEnumerable. ITypeAttribute [▶ 2725] . The items.
-------	---

Reference

[TypeAttributeCollection Class](#) [► 2952]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.134.3.3 TypeAttributeCollection.AsReadOnly Method

Gets a read only version of this [TypeAttributeCollection](#) [► 2952]

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyTypeAttributeCollection AsReadOnly()
```

Field Value

Type: [ReadOnlyTypeAttributeCollection](#) [► 2866]

As read only.

Return Value

Type: [ReadOnlyTypeAttributeCollection](#) [► 2866]

ReadOnlyAttributeCollection.

Reference

[TypeAttributeCollection Class](#) [► 2952]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.134.3.4 TypeAttributeCollection.Clear Method

Removes all items from the ICollection.T..

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Clear()
```

Implements

ICollection.T..Clear.



Reference

[TypeAttributeCollection Class](#) [► 2952]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.134.3.5 TypeAttributeCollection.Contains Method

Overload List

	Name	Description
	Contains(ITypeAttribute) [▸ 2961]	Determines whether the ICollection.T. contains a specific value.
	Contains(String) [▸ 2961]	Determines whether this TypeAttributeCollection [▸ 2952] contains the ITypeAttribute [▸ 2725] with the specified name.

Reference

[TypeAttributeCollection Class \[▸ 2952\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.134.3.5.1 TypeAttributeCollection.Contains Method (String)

Determines whether this [TypeAttributeCollection \[▸ 2952\]](#) contains the [ITypeAttribute \[▸ 2725\]](#) with the specified name.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(
    string name
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: Boolean
true if [contains] [the specified name]; otherwise, false.

Implements

[ITypeAttributeCollection.Contains\(String\) \[▸ 2730\]](#)

Reference

[TypeAttributeCollection Class \[▸ 2952\]](#)

[Contains Overload \[▸ 2961\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.134.3.5.2 TypeAttributeCollection.Contains Method (ITypeAttribute)

Determines whether the ICollection.T. contains a specific value.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(
    ITypeAttribute item
)
```

Parameters

item	Type: TwinCAT.TypeSystem.ITypeAttribute [▶ 2725] The object to locate in the ICollection.T..
------	---

Return Value

Type: Boolean

true if item is found in the ICollection.T.; otherwise, false.

Implements

ICollection.T..Contains(T)

Reference

[TypeAttributeCollection Class](#) [[▶ 2952](#)]

[Contains Overload](#) [[▶ 2961](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.134.3.6 TypeAttributeCollection.CopyTo Method

Copies to.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void CopyTo(
    ITypeAttribute[] array,
    int arrayIndex
)
```

Parameters

array	Type: .TwinCAT.TypeSystem.ITypeAttribute [▶ 2725]. The array.
arrayIndex	Type: System.Int32 Index of the array.

Implements

ICollection.T..CopyTo(.T., Int32)

Reference

[TypeAttributeCollection Class \[► 2952\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.134.3.7 TypeAttributeCollection.Empty Method

Returns an Empty [TypeAttributeCollection \[► 2952\]](#).

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public static TypeAttributeCollection Empty()
```

Return Value

Type: [TypeAttributeCollection \[► 2952\]](#)

TypeAttributeCollection.

Reference

[TypeAttributeCollection Class \[► 2952\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.134.3.8 TypeAttributeCollection.GetEnumerator Method

Returns an enumerator that iterates through the collection.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IEnumerator<ITypeAttribute> GetEnumerator()
```

Return Value

Type: [IEnumerator.ITypeAttribute \[► 2725\]](#).

A [IEnumerator.T.](#) that can be used to iterate through the collection.

Implements

[IEnumerable.T..GetEnumerator.](#)

Reference

[TypeAttributeCollection Class \[► 2952\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.134.3.9 TypeAttributeCollection.IndexOf Method

Determines the index of a specific item in the IList.T..

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int IndexOf(
    ITypeAttribute item
)
```

Parameters

item	Type: TwinCAT.TypeSystem.ITypeAttribute [▸ 2725] The object to locate in the IList.T..
------	---

Return Value

Type: Int32

The index of item if found in the list; otherwise, -1.

Implements

IList.T..IndexOf(T)

Reference

[TypeAttributeCollection Class \[▸ 2952\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.134.3.10 TypeAttributeCollection.Insert Method

Inserts an item to the IList.T. at the specified index.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Insert(
    int index,
    ITypeAttribute item
)
```



Parameters

index	Type: System.Int32 The zero-based index at which item should be inserted.
item	Type: TwinCAT.TypeSystem.ITypeAttribute [▸ 2725] The object to insert into the IList.T..

Implements

IList.T..Insert(Int32, T)

Reference[TypeAttributeCollection Class \[► 2952\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.134.3.11 TypeAttributeCollection.Remove Method****Overload List**

	Name	Description
	Remove(ITypeAttribute) [► 2966]	Removes the first occurrence of a specific object from the ICollection.T..
	Remove(String) [► 2965]	Removes the specified ITypeAttribute [► 2725] from the TypeAttributeCollection [► 2952]

Reference[TypeAttributeCollection Class \[► 2952\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.134.3.11. TypeAttributeCollection.Remove Method (String)****1**Removes the specified [ITypeAttribute \[► 2725\]](#) from the [TypeAttributeCollection \[► 2952\]](#)**Namespace:** [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public bool Remove(
    string name
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: Boolean

true if the attribute is removed, false otherwise.

Reference[TypeAttributeCollection Class \[► 2952\]](#)[Remove Overload \[► 2965\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.134.3.11. TypeAttributeCollection.Remove Method (ITypeAttribute)

2

Removes the first occurrence of a specific object from the ICollection.T..

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Remove(  
    ITypeAttribute item  
)
```

Parameters

item	Type: TwinCAT.TypeSystem.ITypeAttribute [▶ 2725] The object to remove from the ICollection.T..
------	---

Return Value

Type: Boolean

true if item was successfully removed from the ICollection.T.; otherwise, false. This method also returns false if item is not found in the original ICollection.T..

Implements

ICollection.T..Remove(T)

Reference

[TypeAttributeCollection Class](#) [[▶ 2952](#)]

[Remove Overload](#) [[▶ 2965](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.134.3.12 TypeAttributeCollection.RemoveAt Method

Removes the IList.T. item at the specified index.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void RemoveAt(  
    int index  
)
```

Parameters

index	Type: System.Int32 The zero-based index of the item to remove.
-------	---

Implements

IList.T..RemoveAt(Int32)

Reference

[TypeAttributeCollection Class](#) [► 2952]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.134.3.13 TypeAttributeCollection.TryGetAttribute Method

Tries to get the specified [ITypeAttribute](#) [► 2725]

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetAttribute(  
    string name,  
    out ITypeAttribute?? att  
)
```

Parameters

name	Type: System.String The name of the ITypeAttribute [► 2725].
att	Type: TwinCAT.TypeSystem.ITypeAttribute [► 2725]. The att.

Return Value

Type: Boolean

true if found, false otherwise.

Implements

[ITypeAttributeCollection.TryGetAttribute\(String, ITypeAttribute.\)](#) [► 2730]

Reference

[TypeAttributeCollection Class](#) [► 2952]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.134.3.14 TypeAttributeCollection.TryGetValue Method

Tries to get the specified Attribute value.

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetValue(  
    string name,  
    out string?? value  
)
```

Parameters

name	Type: System.String The name.
value	Type: System.String. The value.

Return Value

Type: Boolean
true if value could be found, false otherwise.

Implements

[ITypeAttributeCollection.TryGetValue\(String, String.\)](#) [[▶ 2731](#)]

Reference

[TypeAttributeCollection Class](#) [[▶ 2952](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.135 ValueChangedBaseEventArgs Class

Event args for the [RawValueChanged](#) [[▶ 2774](#)] event.

Inheritance Hierarchy

System.Object
 System.EventArgs
 TwinCAT.TypeSystem.ValueChangedBaseEventArgs
 TwinCAT.TypeSystem.RawValueChangedEventArgs [[▶ 2813](#)]
 TwinCAT.TypeSystem.ValueChangedEventArgs [[▶ 2971](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]


Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**



```
public class ValueChangedBaseEventArgs : EventArgs
```

The ValueChangedBaseEventArgs type exposes the following members.

Constructors

	Name	Description
	ValueChangedBaseEventArgs [▶ 2969]	Initializes a new instance of the RawValueChangedEventArgs [▶ 2813] class.

Properties

	Name	Description
	DateTime [▶ 2970]	Notification timestamp
	Symbol [▶ 2970]	Gets the symbol.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.135.1 ValueChangedBaseEventArgs Constructor

Initializes a new instance of the [RawValueChangedEventArgs \[▶ 2813\]](#) class.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected ValueChangedBaseEventArgs (
    ISymbol symbol,
    DateTimeOffset timeStamp
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol.
timeStamp	Type: System.DateTimeOffset The TwinCAT Real time time stamp (UTC)

Reference

[ValueChangedBaseEventArgs Class \[▶ 2968\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.135.2 ValueChangedBaseEventArgs Properties

The [ValueChangedBaseEventArgs \[▶ 2968\]](#) type exposes the following members.

Properties

	Name	Description
	DateTime [▶ 2970]	Notification timestamp
	Symbol [▶ 2970]	Gets the symbol.

Reference[ValueChangedBaseEventArgs Class \[▸ 2968\]](#)[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)**6.11.135.2.1 ValueChangedBaseEventArgs.DateTime Property**

Notification timestamp

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public DateTimeOffset DateTime { get; }
```

Property Value

Type: DateTimeOffset

Reference[ValueChangedBaseEventArgs Class \[▸ 2968\]](#)[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)**6.11.135.2.2 ValueChangedBaseEventArgs.Symbol Property**

Gets the symbol.


Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**






```
public ISymbol Symbol { get; }
```

Property ValueType: [ISymbol \[▸ 2691\]](#)

The symbol.

Reference[ValueChangedBaseEventArgs Class \[▸ 2968\]](#)[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)**6.11.135.3 ValueChangedBaseEventArgs Methods**The [ValueChangedBaseEventArgs \[▸ 2968\]](#) type exposes the following members.**Methods**

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)

	Name	Description
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[ValueChangedBaseEventArgs Class \[▶ 2968\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.136 ValueChangedEventArgs Class

Event args for the [ValueChanged \[▶ 2790\]](#) event.

Inheritance Hierarchy

System.Object
 System.EventArgs
 [TwinCAT.TypeSystem.ValueChangedEventArgs \[▶ 2968\]](#)
 TwinCAT.TypeSystem.ValueChangedEventArgs

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229




Syntax

C#




```
public class ValueChangedEventArgs : ValueChangedEventArgs
```




The ValueChangedEventArgs type exposes the following members.

Properties

	Name	Description
	DateTime [▶ 2970]	Notification timestamp (Inherited from ValueChangedBaseEventArgs [▶ 2968] .)
	Symbol [▶ 2970]	Gets the symbol. (Inherited from ValueChangedBaseEventArgs [▶ 2968] .)
	Value [▶ 2972]	The new received Value

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)

	Name	Description
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)




Reference

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.136.1 ValueChangedEventArgs Properties

The [ValueChangedEventArgs \[► 2971\]](#) type exposes the following members.

Properties

	Name	Description
	DateTime [► 2970]	Notification timestamp (Inherited from ValueChangedBaseEventArgs [► 2968].)
	Symbol [► 2970]	Gets the symbol. (Inherited from ValueChangedBaseEventArgs [► 2968].)
	Value [► 2972]	The new received Value

Reference

[ValueChangedEventArgs Class \[► 2971\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.136.1.1 ValueChangedEventArgs.Value Property

The new received Value

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object Value { get; }
```

Property Value

Type: Object

Reference







[ValueChangedEventArgs Class \[► 2971\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.136.2 ValueChangedEventArgs Methods

The [ValueChangedEventArgs \[► 2971\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[ValueChangedEventArgs Class \[▶ 2971\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.137 AlignedMemberCollection Class

Class AlignedMemberCollection. Implements the [MemberCollection \[▶ 2802\]](#)

Inheritance Hierarchy

System.Object

[TwinCAT.TypeSystem.Generic.InstanceCollection \[▶ 3104\].IMember \[▶ 2561\].](#)

[TwinCAT.TypeSystem.MemberCollection \[▶ 2802\]](#)

TwinCAT.TypeSystem.AlignedMemberCollection

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public class AlignedMemberCollection : MemberCollection
```

The AlignedMemberCollection type exposes the following members.

Constructors

	Name	Description
	AlignedMemberCollection [▶ 2975]	Initializes a new instance of the AlignedMemberCollection class.

Properties

	Name	Description
	Count [▶ 3107]	Gets the collection count. (Inherited from InstanceCollection.T. [▶ 3104].)
	Empty [▶ 2976]	Returns an Empty Member Collection.
	InnerList [▶ 3108]	Gets the List of instances. (Inherited from InstanceCollection.T. [▶ 3104].)

Name	Description
InnerPathDict [▶ 3108]	The Path dictionary (Inherited from InstanceCollection.T. [▶ 3104].)
Instances [▶ 2807]	Gets the Instance members (non static) (Inherited from MemberCollection [▶ 2802].)
IsReadOnly [▶ 3109]	Gets a value indicating whether this instance is read only. (Inherited from InstanceCollection.T. [▶ 3104].)
Item.Int32. [▶ 3109]	Gets or sets the Instance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
Item.String. [▶ 3110]	Gets the Instance [▶ 2549] with the specified instance path. (Inherited from InstanceCollection.T. [▶ 3104].)
Mode [▶ 3111]	The mode this InstanceCollection.T. [▶ 3104] is working in. (Inherited from InstanceCollection.T. [▶ 3104].)
Statics [▶ 2807]	Gets the Static Members (Inherited from MemberCollection [▶ 2802].)

Methods

Name	Description
Add [▶ 2978]	Adds the specified item. (Overrides MemberCollection.Add(IMember) [▶ 2811].)
AddRange [▶ 3113]	Adds the specified items to this collection. (Inherited from InstanceCollection.T. [▶ 3104].)
AsReadOnly [▶ 2809]	Returns a read only copy of this collection (shallow copy) (Inherited from MemberCollection [▶ 2802].)
CalcSize [▶ 2811]	Calculates the Byte Size of the IMemberCollection [▶ 2565] (Inherited from MemberCollection [▶ 2802].)
Clear [▶ 3114]	Clears this instance. (Inherited from InstanceCollection.T. [▶ 3104].)
Clone [▶ 2809]	Clones this MemberCollection [▶ 2802]. (Inherited from MemberCollection [▶ 2802].)
Contains(T) [▶ 3115]	Determines whether this collection contains the specified Instance [▶ 2549] (Inherited from InstanceCollection.T. [▶ 3104].)
Contains(String) [▶ 3114]	Determines whether this collection contains an Instance [▶ 2549] with the specified InstanceName / InstancePath (Inherited from InstanceCollection.T. [▶ 3104].)
ContainsName [▶ 3116]	Determines whether the specified instance name contains name. (Inherited from InstanceCollection.T. [▶ 3104].)
CopyTo [▶ 3116]	Copies this InstanceCollection.T. [▶ 3104] to the specified array. (Inherited from InstanceCollection.T. [▶ 3104].)
Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
GetEnumerator [▶ 3117]	Gets the enumerator. (Inherited from InstanceCollection.T. [▶ 3104].)
GetHashCode	Serves as the default hash function. (Inherited from Object.)
GetInstance [▶ 3117]	Gets the Instance [▶ 2549] by instance path. (Inherited from InstanceCollection.T. [▶ 3104].)
GetInstanceByName [▶ 3118]	Gets the name of the instance by. (Inherited from InstanceCollection.T. [▶ 3104].)

	Name	Description
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 3119]	Determines the index of the specified IInstance [▶ 2549]. (Inherited from InstanceCollection.T. [▶ 3104].)
	Insert [▶ 2979]	Inserts the specified IInstance [▶ 2549] at the specified index. (Overrides InstanceCollection.T..Insert(Int32, T) [▶ 3119].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 3120]	Removes the specified item. (Inherited from InstanceCollection.T. [▶ 3104].)
	RemoveAt [▶ 2979]	Removes the IInstance [▶ 2549] at the specified index. (Overrides InstanceCollection.T..RemoveAt(Int32) [▶ 3121].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInstance [▶ 3121]	Tries to get the IInstance [▶ 2549] of the specified path. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetInstanceByName [▶ 3122]	Tries to get Instances by name. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetMember [▶ 2810]	Tries to get the specified member (Inherited from MemberCollection [▶ 2802].)

Extension Methods

	Name	Description
	AddAligned [▶ 2981]	Adds a member to the StructType [▶ 1846] (Defined by FluentAlignedMemberCollectionExtension [▶ 2980].)

Remarks

The [AlignedMemberCollection](#) calculates its member offsets by itself. Dependant on the pack mode

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

[TwinCAT.TypeSystem.MemberCollection](#) [[▶ 2802](#)]

6.11.137.1 AlignedMemberCollection Constructor

Initializes a new instance of the [AlignedMemberCollection](#) [[▶ 2973](#)] class.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AlignedMemberCollection()
```

Reference

[AlignedMemberCollection Class](#) [[▶ 2973](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.137.2 AlignedMemberCollection Properties

The [AlignedMemberCollection](#) [▶ 2973] type exposes the following members.

Properties

	Name	Description
	Count [▶ 3107]	Gets the collection count. (Inherited from InstanceCollection.T. [▶ 3104].)
	Empty [▶ 2976]	Returns an Empty Member Collection.
	InnerList [▶ 3108]	Gets the List of instances. (Inherited from InstanceCollection.T. [▶ 3104].)
	InnerPathDict [▶ 3108]	The Path dictionary (Inherited from InstanceCollection.T. [▶ 3104].)
	Instances [▶ 2807]	Gets the Instance members (non static) (Inherited from MemberCollection [▶ 2802].)
	IsReadOnly [▶ 3109]	Gets a value indicating whether this instance is read only. (Inherited from InstanceCollection.T. [▶ 3104].)
	Item.Int32. [▶ 3109]	Gets or sets the Instance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	Item.String. [▶ 3110]	Gets the Instance [▶ 2549] with the specified instance path. (Inherited from InstanceCollection.T. [▶ 3104].)
	Mode [▶ 3111]	The mode this InstanceCollection.T. [▶ 3104] is working in. (Inherited from InstanceCollection.T. [▶ 3104].)
	Statics [▶ 2807]	Gets the Static Members (Inherited from MemberCollection [▶ 2802].)

Reference

[AlignedMemberCollection Class](#) [▶ 2973]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.137.2.1 AlignedMemberCollection.Empty Property

Returns an Empty Member Collection.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AlignedMemberCollection Empty { get; }
```

Return Value

Type: [AlignedMemberCollection](#) [▶ 2973]

MemberCollection.

Reference

[AlignedMemberCollection Class](#) [▶ 2973]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.137.3 AlignedMemberCollection Methods

The [AlignedMemberCollection](#) [▶ 2973] type exposes the following members.

Methods

Name	Description
Add [▶ 2978]	Adds the specified item. (Overrides MemberCollection.Add(IMember) [▶ 2811].)
AddRange [▶ 3113]	Adds the specified items to this collection. (Inherited from InstanceCollection.T. [▶ 3104].)
AsReadOnly [▶ 2809]	Returns a read only copy of this collection (shallow copy) (Inherited from MemberCollection [▶ 2802].)
CalcSize [▶ 2811]	Calculates the Byte Size of the IMemberCollection [▶ 2565] (Inherited from MemberCollection [▶ 2802].)
Clear [▶ 3114]	Clears this instance. (Inherited from InstanceCollection.T. [▶ 3104].)
Clone [▶ 2809]	Clones this MemberCollection [▶ 2802]. (Inherited from MemberCollection [▶ 2802].)
Contains(T) [▶ 3115]	Determines whether this collection contains the specified IInstance [▶ 2549] (Inherited from InstanceCollection.T. [▶ 3104].)
Contains(String) [▶ 3114]	Determines whether this collection contains an IInstance [▶ 2549] with the specified InstanceName / InstancePath (Inherited from InstanceCollection.T. [▶ 3104].)
ContainsName [▶ 3116]	Determines whether the specified instance name contains name. (Inherited from InstanceCollection.T. [▶ 3104].)
CopyTo [▶ 3116]	Copies this InstanceCollection.T. [▶ 3104] to the specified array. (Inherited from InstanceCollection.T. [▶ 3104].)
Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
GetEnumerator [▶ 3117]	Gets the enumerator. (Inherited from InstanceCollection.T. [▶ 3104].)
GetHashCode	Serves as the default hash function. (Inherited from Object.)
GetInstance [▶ 3117]	Gets the IInstance [▶ 2549] by instance path. (Inherited from InstanceCollection.T. [▶ 3104].)
GetInstanceByName [▶ 3118]	Gets the name of the instance by. (Inherited from InstanceCollection.T. [▶ 3104].)
GetType	Gets the Type of the current instance. (Inherited from Object.)
IndexOf [▶ 3119]	Determines the index of the specified IInstance [▶ 2549]. (Inherited from InstanceCollection.T. [▶ 3104].)
Insert [▶ 2979]	Inserts the specified IInstance [▶ 2549] at the specified index. (Overrides InstanceCollection.T..Insert(Int32, T) [▶ 3119].)
MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
Remove [▶ 3120]	Removes the specified item. (Inherited from InstanceCollection.T. [▶ 3104].)
RemoveAt [▶ 2979]	Removes the IInstance [▶ 2549] at the specified index. (Overrides InstanceCollection.T..RemoveAt(Int32) [▶ 3121].)
ToString	Returns a string that represents the current object. (Inherited from Object.)
TryGetInstance [▶ 3121]	Tries to get the IInstance [▶ 2549]. of the specified path. (Inherited from InstanceCollection.T. [▶ 3104].)

	Name	Description
	TryGetInstanceByName [▶ 3122]	Tries to get Instances by name. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetMember [▶ 2810]	Tries to get the specified member (Inherited from MemberCollection [▶ 2802].)

Extension Methods

	Name	Description
	AddAligned [▶ 2981]	Adds a member to the StructType [▶ 1846] (Defined by FluentAlignedMemberCollectionExtension [▶ 2980].)

Reference

[AlignedMemberCollection Class](#) [[▶ 2973](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.137.3.1 AlignedMemberCollection.Add Method

Adds the specified item.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override void Add(
    IMember item
)
```

Parameters

item	Type: TwinCAT.TypeSystem.IMember [▶ 2561] The item.
------	--

Implements

[ICollection.T..Add\(T\)](#)

[ICollection.T..Add\(T\)](#)

Exceptions

Exception	Condition
ArgumentNullException	item
NotSupportedException	Static members are not supported yet!

Reference

[AlignedMemberCollection Class](#) [[▶ 2973](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.137.3.2 AlignedMemberCollection.Insert Method

Inserts the specified [IInstance](#) [[▶ 2549](#)] at the specified index.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override void Insert(  
    int index,  
    IMember instance  
)
```

Parameters

index	Type: System.Int32 The instance.
instance	Type: TwinCAT.TypeSystem.IMember [▶ 2561] The item.

Implements

[IList.T..Insert\(Int32, T\)](#)

Exceptions

Exception	Condition
NotSupportedException	

Reference

[AlignedMemberCollection Class](#) [[▶ 2973](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.137.3.3 AlignedMemberCollection.RemoveAt Method

Removes the [IInstance](#) [[▶ 2549](#)] at the specified index.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override void RemoveAt(  
    int index  
)
```

Parameters

index	Type: System.Int32 The index.
-------	----------------------------------

Implements

[IList.T..RemoveAt\(Int32\)](#)

Exceptions

Exception	Condition
NotSupportedException	

Reference

[AlignedMemberCollection Class \[► 2973\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.138 FluentAlignedMemberCollectionExtension Class

Class FluentStructTypeExtension.

Inheritance Hierarchy

System.Object

TwinCAT.TypeSystem.FluentAlignedMemberCollectionExtension

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static class FluentAlignedMemberCollectionExtension
```

The [FluentAlignedMemberCollectionExtension](#) type exposes the following members.

Methods

	Name	Description
	AddAligned [► 2981]	Adds a member to the StructType [► 1846]

Remarks

Fluent interface for adding members to [StructType \[► 1846\]](#)s.

Reference

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.138.1 FluentAlignedMemberCollectionExtension Methods

The [FluentAlignedMemberCollectionExtension \[► 2980\]](#) type exposes the following members.

Methods

	Name	Description
	AddAligned [► 2981]	Adds a member to the StructType [► 1846]

Reference

[FluentAlignedMemberCollectionExtension Class \[► 2980\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.138.1.1 **FluentAlignedMemberCollectionExtension.AddAligned Method**

Adds a member to the [StructType](#) [► 1846]

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static AlignedMemberCollection AddAligned(  
    this AlignedMemberCollection str,  
    IMember member  
)
```

Parameters

str	Type: TwinCAT.TypeSystem.AlignedMemberCollection [► 2973] The string.
member	Type: TwinCAT.TypeSystem.IMember [► 2561] The member.

Return Value

Type: [AlignedMemberCollection](#) [► 2973]
StructType.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [AlignedMemberCollection](#) [► 2973]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[FluentAlignedMemberCollectionExtension Class](#) [► 2980]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.139 **IDimensionCollectionExtension Class**

Class IDimensionCollectionExtension.

Inheritance Hierarchy

System.Object
TwinCAT.TypeSystem.IDimensionCollectionExtension

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static class IDimensionCollectionExtension
```

The IDimensionCollectionExtension type exposes the following members.

Methods

	Name	Description
	AddDimension [▶ 2982]	Adds a Dimension (FluentInterface)

Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.139.1 IDimensionCollectionExtension Methods

The [IDimensionCollectionExtension](#) [▶ 2981] type exposes the following members.

Methods

	Name	Description
	AddDimension [▶ 2982]	Adds a Dimension (FluentInterface)

Reference

[IDimensionCollectionExtension Class](#) [▶ 2981]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.139.1.1 IDimensionCollectionExtension.AddDimension Method

Adds a Dimension (FluentInterface)

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static IDimensionCollection AddDimension(
    this IDimensionCollection coll,
    IDimension dim
)
```

Parameters

coll	Type: TwinCAT.TypeSystem.IDimensionCollection [▶ 2492] The coll.
dim	Type: TwinCAT.TypeSystem.IDimension [▶ 2490] The dim.

Return Value

Type: [IDimensionCollection](#) [▶ 2492]

DimensionCollection.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IDimensionCollection](#) [▶ 2492]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[IDimensionCollectionExtension Class \[▸ 2981\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.140 InterfaceType Interface

Interface representing an Interface Data type. Implements the [IDataType \[▸ 2475\]](#)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public interface IInterfaceType : IDatatype,
    IBitSize, IRpcCallableType
```

The IInterfaceType type exposes the following members.

Properties

Name	Description
AllMembers [▸ 2987]	Gets all members (down the derivation hierarchy)
Attributes [▸ 2478]	Gets the attributes of the IDataType [▸ 2475] (Inherited from IDataType [▸ 2475] .)
BaseType [▸ 2988]	Gets the structs Base Type (Null if not derived).
BaseTypeName [▸ 2988]	Gets the the Name of the Base class (if derived)
BitSize [▸ 2473]	Gets the size of the IDataType [▸ 2475] in bits. (Inherited from IBitSize [▸ 2472] .)
ByteSize [▸ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▸ 2472] .)
Category [▸ 2478]	Gets the Data Type category (Inherited from IDataType [▸ 2475] .)
Comment [▸ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▸ 2475] .)
FullName [▸ 2479]	Gets the full name of the IDataType [▸ 2475] (Namespace + Name) (Inherited from IDataType [▸ 2475] .)
HasRpcMethods [▸ 2986]	Gets a value indicating whether this instance has RPC methods (Struct types only)
Id [▸ 2480]	Gets the ID of the DataType (Inherited from IDataType [▸ 2475] .)
InterfaceImplementationNames [▸ 2987]	Gets the names of the interfaces, this IDataType [▸ 2475] implements.
InterfaceImplementations [▸ 2987]	Gets the resolved interface types, this IDataType [▸ 2475] implements.
IsBitType [▸ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▸ 2472] .)
IsByteAligned [▸ 2475]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from IBitSize [▸ 2472] .)
Members [▸ 2988]	Gets a readonly collection of the Members [▸ 2561] of the IInterfaceType.

Name	Description
Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475].)
Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475].)
RpcMethods [▶ 2624]	Gets the Method descriptions for the IRpcCallableType [▶ 2624] (Inherited from IRpcCallableType [▶ 2624].)
Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Extension Methods

Name	Description
IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)

	Name	Description
	Resolve(DataTypeResolveStrategy) [▸ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▸ 3021] .)

Reference

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

[TwinCAT.TypeSystem.IDataType \[▸ 2475\]](#)

Also see about this

- [IDataType.IsContainer Property \[▸ 2480\]](#)
- [IDataType.IsPointer Property \[▸ 2481\]](#)
- [IDataType.IsPrimitive Property \[▸ 2481\]](#)
- [IDataType.IsReference Property \[▸ 2482\]](#)

6.11.140.1 InterfaceType Properties

The [InterfaceType \[▸ 2983\]](#) type exposes the following members.

Properties

	Name	Description
	AllMembers [▸ 2987]	Gets all members (down the derivation hierarchy)
	Attributes [▸ 2478]	Gets the attributes of the IDataType [▸ 2475] (Inherited from IDataType [▸ 2475] .)
	BaseType [▸ 2988]	Gets the structs Base Type (Null if not derived).
	BaseTypeName [▸ 2988]	Gets the the Name of the Base class (if derived)
	BitSize [▸ 2473]	Gets the size of the IDataType [▸ 2475] in bits. (Inherited from IBitSize [▸ 2472] .)
	ByteSize [▸ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▸ 2472] .)
	Category [▸ 2478]	Gets the Data Type category (Inherited from IDataType [▸ 2475] .)
	Comment [▸ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▸ 2475] .)
	FullName [▸ 2479]	Gets the full name of the IDataType [▸ 2475] (Namespace + Name) (Inherited from IDataType [▸ 2475] .)
	HasRpcMethods [▸ 2986]	Gets a value indicating whether this instance has RPC methods (Struct types only)
	Id [▸ 2480]	Gets the ID of the DataType (Inherited from IDataType [▸ 2475] .)
	InterfaceImplementationNames [▸ 2987]	Gets the names of the interfaces, this IDataType [▸ 2475] implements.
	InterfaceImplementations [▸ 2987]	Gets the resolved interface types, this IDataType [▸ 2475] implements.
	IsBitType [▸ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▸ 2472] .)

	Name	Description
	IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from IBitSize [▶ 2472].)
	Members [▶ 2988]	Gets a readonly collection of the Members [▶ 2561] of the IInterfaceType [▶ 2983].
	Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475].)
	Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475].)
	RpcMethods [▶ 2624]	Gets the Method descriptions for the IRpcCallableType [▶ 2624] (Inherited from IRpcCallableType [▶ 2624].)
	Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Reference

[IInterfaceType Interface](#) [▶ 2983]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

Also see about this

[IDataType.IsContainer Property](#) [▶ 2480]

[IDataType.IsPointer Property](#) [▶ 2481]

[IDataType.IsPrimitive Property](#) [▶ 2481]

[IDataType.IsReference Property](#) [▶ 2482]

6.11.140.1.1 IInterfaceType.HasRpcMethods Property

Gets a value indicating whether this instance has RPC methods (Struct types only)

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool HasRpcMethods { get; }
```

Property Value

Type: Boolean

true if this instance has RPC methods; otherwise, false.

Remarks

The DataType (Structure) must be marked with the PlcAttribute 'TcRpcEnable' to enable RpcMethods, otherwise RpcMethods are not passed through to the ADS symbolic information.

Reference

[IInterfaceType Interface](#) [▶ 2983]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.140.1.2 **InterfaceType.InterfaceImplementationNames Property**

Gets the names of the interfaces, this [IDataType](#) [[▶ 2475](#)] implements.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string[] InterfaceImplementationNames { get; }
```

Property Value

Type: .String.

The interface implementations.

Reference

[IInterfaceType Interface](#) [[▶ 2983](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.140.1.3 **InterfaceType.InterfaceImplementations Property**

Gets the resolved interface types, this [IDataType](#) [[▶ 2475](#)] implements.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IInterfaceType?[]? InterfaceImplementations { get; }
```

Property Value

Type: [.IInterfaceType](#) [[▶ 2983](#)].

The implements.

Reference

[IInterfaceType Interface](#) [[▶ 2983](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.140.1.4 **InterfaceType.AllMembers Property**

Gets all members (down the derivation hierarchy)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IMemberCollection AllMembers { get; }
```

Property Value

Type: [IMemberCollection](#) [▶ 2565]
All members.

Reference

[IInterfaceType Interface](#) [▶ 2983]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.140.1.5 IInterfaceType.BaseType Property

Gets the structs Base Type (Null if not derived).

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDataType? BaseType { get; }
```

Property Value

Type: [IDataType](#) [▶ 2475]

Reference

[IInterfaceType Interface](#) [▶ 2983]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.140.1.6 IInterfaceType.BaseTypeName Property

Gets the the Name of the Base class (if derived)

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string BaseTypeName { get; }
```

Property Value

Type: String
Empty if not derived.

Reference

[IInterfaceType Interface](#) [▶ 2983]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.140.1.7 IInterfaceType.Members Property

Gets a readonly collection of the [Members](#) [▶ 2561] of the [IInterfaceType](#) [▶ 2983].

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IMemberCollection Members { get; }
```

Property Value

Type: [IMemberCollection](#) [[▶ 2565](#)]

The members as readonly collection.

Remarks

If the [IStructType](#) [[▶ 2671](#)] is derived, only the extended members are returned. To get all supported members down the inheritance chain, use the [AllMembers](#) [[▶ 2987](#)] property.

Reference

[IInterfaceType Interface](#) [[▶ 2983](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.140.2 IInterfaceType Methods

The [IInterfaceType](#) [[▶ 2983](#)] type exposes the following members.

Extension Methods

Name	Description
IsArrayOfPrimitives [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsContainer [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)

	Name	Description
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[IInterfaceType](#) Interface [[▶ 2983](#)]

[TwinCAT.TypeSystem](#) Namespace [[▶ 2083](#)]

6.11.141 IRpcStructType Interface

Note: This API is now obsolete.

Interface [IRpcStructType](#) Implements the [IRpcCallableType](#) [[▶ 2624](#)] Implements the [IStructType](#) [[▶ 2671](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
[ObsoleteAttribute("Use IStructType instead")]
public interface IRpcStructType : IRpcCallableType,
    IStructType, IInterfaceType, IDataTypes, IBitSize
```

The [IRpcStructType](#) type exposes the following members.

Properties

	Name	Description
	AllMembers [▶ 2987]	Gets all members (down the derivation hierarchy) (Inherited from IInterfaceType [▶ 2983].)
	Attributes [▶ 2478]	Gets the attributes of the IDataType [▶ 2475] (Inherited from IDataType [▶ 2475].)
	BaseType [▶ 2988]	Gets the structs Base Type (Null if not derived). (Inherited from IInterfaceType [▶ 2983].)
	BaseTypeName [▶ 2988]	Gets the the Name of the Base class (if derived) (Inherited from IInterfaceType [▶ 2983].)
	BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)

Name	Description
ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)
FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from IDataType [▶ 2475].)
HasRpcMethods [▶ 2986]	Gets a value indicating whether this instance has RPC methods (Struct types only) (Inherited from IInterfaceType [▶ 2983].)
Id [▶ 2480]	Gets the ID of the DataType (Inherited from IDataType [▶ 2475].)
InterfacelImplementationNames [▶ 2987]	Gets the names of the interfaces, this IDataType [▶ 2475] implements. (Inherited from IInterfaceType [▶ 2983].)
InterfacelImplementations [▶ 2987]	Gets the resolved interface types, this IDataType [▶ 2475] implements. (Inherited from IInterfaceType [▶ 2983].)
IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
Members [▶ 2988]	Gets a readonly collection of the Members [▶ 2561] of the IInterfaceType [▶ 2983]. (Inherited from IInterfaceType [▶ 2983].)
Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475].)
Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475].)
RpcMethods [▶ 2624]	Gets the Method descriptions for the IRpcCallableType [▶ 2624] (Inherited from IRpcCallableType [▶ 2624].)
Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Extension Methods

Name	Description
IsArrayOfPrimitives. [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsArrayOfPrimitives(Boolean) [▶ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▶ 3021].)
IsContainer. [▶ 3027]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)
IsContainer(Boolean) [▶ 3028]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is a container type (Defined by DataTypeExtension [▶ 3021].)

	Name	Description
	IsPointer [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive . [▶ 3033]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsPrimitive(Boolean) [▶ 3034]	Overloaded. Gets a value indicating whether this IDataType [▶ 2475] is primitive (Defined by DataTypeExtension [▶ 3021].)
	IsReference [▶ 3036]	Gets a value indicating whether this IDataType [▶ 2475] is a reference type (Defined by DataTypeExtension [▶ 3021].)
	ReferencesExternalData [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive . [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	ResolvableAsPrimitive(Boolean) [▶ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▶ 3021].)
	Resolve . [▶ 3040]	Overloaded. Resolves the specified data type for AliasReference [▶ 3042] (Defined by DataTypeExtension [▶ 3021].)
	Resolve(DataTypeResolveStrategy) [▶ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

[TwinCAT.TypeSystem.IRpcCallableType](#) [[▶ 2624](#)]

[TwinCAT.TypeSystem.IStructType](#) [[▶ 2671](#)]

Also see about this

- [IDataType.IsContainer Property](#) [[▶ 2480](#)]
- [IDataType.IsPointer Property](#) [[▶ 2481](#)]
- [IDataType.IsPrimitive Property](#) [[▶ 2481](#)]
- [IDataType.IsReference Property](#) [[▶ 2482](#)]

6.11.141.1 IRpcStructType Properties

The [IRpcStructType](#) [[▶ 2990](#)] type exposes the following members.

Properties

	Name	Description
	AllMembers [▶ 2987]	Gets all members (down the derivation hierarchy) (Inherited from InterfaceType [▶ 2983].)

Name	Description
Attributes [▶ 2478]	Gets the attributes of the IDataType [▶ 2475] (Inherited from IDataType [▶ 2475].)
BaseType [▶ 2988]	Gets the structs Base Type (Null if not derived). (Inherited from IInterfaceType [▶ 2983].)
BaseTypeName [▶ 2988]	Gets the the Name of the Base class (if derived) (Inherited from IInterfaceType [▶ 2983].)
BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
Category [▶ 2478]	Gets the Data Type category (Inherited from IDataType [▶ 2475].)
Comment [▶ 2479]	Gets the comment behind the variable declaration. (Inherited from IDataType [▶ 2475].)
FullName [▶ 2479]	Gets the full name of the IDataType [▶ 2475] (Namespace + Name) (Inherited from IDataType [▶ 2475].)
HasRpcMethods [▶ 2986]	Gets a value indicating whether this instance has RPC methods (Struct types only) (Inherited from IInterfaceType [▶ 2983].)
Id [▶ 2480]	Gets the ID of the Data Type (Inherited from IDataType [▶ 2475].)
InterfaceImplementationNames [▶ 2987]	Gets the names of the interfaces, this IDataType [▶ 2475] implements. (Inherited from IInterfaceType [▶ 2983].)
InterfaceImplementations [▶ 2987]	Gets the resolved interface types, this IDataType [▶ 2475] implements. (Inherited from IInterfaceType [▶ 2983].)
IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full Data Type but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
Members [▶ 2988]	Gets a readonly collection of the Members [▶ 2561] of the IInterfaceType [▶ 2983]. (Inherited from IInterfaceType [▶ 2983].)
Name [▶ 2482]	Gets the name of the Data Type (without namespace) (Inherited from IDataType [▶ 2475].)
Namespace [▶ 2482]	Gets the namespace string within the IDataType [▶ 2475] exists. (Inherited from IDataType [▶ 2475].)
RpcMethods [▶ 2624]	Gets the Method descriptions for the IRpcCallableType [▶ 2624] (Inherited from IRpcCallableType [▶ 2624].)
Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)

Reference

[IRpcStructType Interface](#) [[▶ 2990](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

Also see about this

- [IDataType.IsContainer Property](#) [[▶ 2480](#)]
- [IDataType.IsPointer Property](#) [[▶ 2481](#)]

 [IDataType.IsPrimitive Property \[▸ 2481\]](#)

 [IDataType.IsReference Property \[▸ 2482\]](#)

6.11.141.2 IRpcStructType Methods

The [IRpcStructType \[▸ 2990\]](#) type exposes the following members.

Extension Methods

Name	Description
IsArrayOfPrimitives. [▸ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▸ 3021].)
IsArrayOfPrimitives(Boolean) [▸ 3025]	Overloaded. Determines whether the dataType is an array of Primitives. (Defined by DataTypeExtension [▸ 3021].)
IsContainer. [▸ 3027]	Overloaded. Gets a value indicating whether this IDataType [▸ 2475] is a container type (Defined by DataTypeExtension [▸ 3021].)
IsContainer(Boolean) [▸ 3028]	Overloaded. Gets a value indicating whether this IDataType [▸ 2475] is a container type (Defined by DataTypeExtension [▸ 3021].)
IsPointer [▸ 3031]	Gets a value indicating whether this IDataType [▸ 2475] is a pointer type (Defined by DataTypeExtension [▸ 3021].)
IsPrimitive. [▸ 3033]	Overloaded. Gets a value indicating whether this IDataType [▸ 2475] is primitive (Defined by DataTypeExtension [▸ 3021].)
IsPrimitive(Boolean) [▸ 3034]	Overloaded. Gets a value indicating whether this IDataType [▸ 2475] is primitive (Defined by DataTypeExtension [▸ 3021].)
IsReference [▸ 3036]	Gets a value indicating whether this IDataType [▸ 2475] is a reference type (Defined by DataTypeExtension [▸ 3021].)
ReferencesExternalD ata [▸ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▸ 3021].)
ResolvableAsPrimiti ve. [▸ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▸ 3021].)
ResolvableAsPrimiti ve(Boolean) [▸ 3039]	Overloaded. Indicates, that this data type can be resolved as primitive or primitive Array. (Defined by DataTypeExtension [▸ 3021].)
Resolve. [▸ 3040]	Overloaded. Resolves the specified data type for AliasReference [▸ 3042] (Defined by DataTypeExtension [▸ 3021].)
Resolve(DataTypeRe solveStrategy) [▸ 3041]	Overloaded. Resolves the specified strategy. (Defined by DataTypeExtension [▸ 3021].)

Reference

[IRpcStructType Interface \[▶ 2990\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.142 DynamicInterfaceInstance Class

Dynamic struct instance

Inheritance Hierarchy

- System.Object
- System.Dynamic.DynamicObject
- [TwinCAT.TypeSystem.DynamicSymbol \[▶ 2266\]](#)
- [TwinCAT.TypeSystem.DynamicInterfaceInstance](#)
- [TwinCAT.TypeSystem.DynamicStructInstance \[▶ 2251\]](#)

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public class DynamicInterfaceInstance : DynamicSymbol,
    IInterfaceInstance, ISymbol, IAttributedInstance, IInstance, IBitSize,
    IRpcCallableInstance
```

The DynamicInterfaceInstance type exposes the following members.

Properties

Name	Description
_InnerSymbol [▶ 2274]	Gets the inner symbol of this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266] .)
AccessRights [▶ 2274]	Gets the access rights. (Inherited from DynamicSymbol [▶ 2266] .)
AllowIIOAccess [▶ 2275]	Indicates, that the aggregates symbols is an IProcessImageAddress (and most probably IAdsSymbol) (Inherited from DynamicSymbol [▶ 2266] .)
Attributes [▶ 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [▶ 2266] .)
BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from DynamicSymbol [▶ 2266] .)
ByteSize [▶ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [▶ 2266] .)
Category [▶ 2277]	Gets the category. (Inherited from DynamicSymbol [▶ 2266] .)
Comment [▶ 2277]	Gets the comment of the IInstance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266] .)
Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266] .)
DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549] . (Inherited from DynamicSymbol [▶ 2266] .)
HasRpcMethods [▶ 3001]	Gets a value indicating whether this instance has RPC methods
HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances, what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266] .)

Name	Description
InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266].)
IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from DynamicSymbol [▶ 2266].)
IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from DynamicSymbol [▶ 2266].)
IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266].)
IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266].)
IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266].)
IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266].)
IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266].)
IsReference [▶ 2284]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)
MemberInstances [▶ 3002]	Gets the member instances of the Struct Instance [▶ 2666].
NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)
NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
RpcMethods [▶ 3003]	Gets the Method descriptions for the IRpcCallableType [▶ 2624]
Size [▶ 2286]	Gets the size of the IInstance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)

Methods

Name	Description
Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetDynamicMemberNames [▶ 3006]	Returns the enumeration of all dynamic member names. (Overrides DynamicSymbol.GetDynamicMemberNames . [▶ 2293].)
	GetHashCode [▶ 2293]	Gets the GetHashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	InvokeRpcMethodAsync [▶ 3007]	Invokes the specified RPC Method asynchronously
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	OnReadAnyValue [▶ 2293]	Handler function for reading ADS 'Any' Values. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadRawValue [▶ 2294]	Handler function for reading Raw symbol value. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadRawValueAsync [▶ 2294]	Handler function reading the raw value of the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	OnReadValue [▶ 2295]	Handler function for the (Inherited from DynamicSymbol [▶ 2266].)
	OnReadValueAsync [▶ 2296]	Handler function reading the DynamicSymbols [▶ 2266] value asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	OnSetInstanceName [▶ 2296]	Sets a new InstanceName InstancePath (Inherited from DynamicSymbol [▶ 2266].)
	OnTryReadValue [▶ 2297]	Handler function for the (Inherited from DynamicSymbol [▶ 2266].)
	OnTryWriteValue [▶ 2297]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteRawValue [▶ 2298]	Handler function for reading symbols raw value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteRawValueAsync [▶ 2298]	Handler function for writing the raw DynamicSymbol [▶ 2266] value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteValue [▶ 2299]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	OnWriteValueAsync [▶ 2300]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
	ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValue . [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)
	ReadRawValueAsync [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
	ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject .)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryGetIndex	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicObject .)
	TryGetMember [▶ 3008]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Overrides DynamicSymbol.TryGetMember(GetMemberBinder, Object.) [▶ 2308].)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject .)
	TryInvokeMember [▶ 3009]	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Overrides DynamicObject.TryInvokeMember(InvokeMemberBinder, Object, Object.) .)
	TryInvokeRpcMethod(String, Object, Object.) [▶ 3010]	Invokes the specified RPC Method

	Name	Description
	TryInvokeRpcMethod(String, Object, Object, Object.) [▶ 3012]	Invokes the specified RPC Method
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	TrySetIndex	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicObject .)
	TrySetMember [▶ 3013]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Overrides DynamicObject.TrySetMember(SetMemberBinder, Object) .)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject .)
	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from DynamicSymbol [▶ 2266].)
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte.) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte, Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)

Events

	Name	Description
	RawValueChanged [▶ 2323]	Occurs when the RawValue of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)
	ValueChanged [▶ 2323]	Occurs when the (Primitive) value of the IValueSymbol [▶ 2775] has changed. (Inherited from DynamicSymbol [▶ 2266].)

Fields

	Name	Description
	syncObject [▶ 2327]	Synchronization object (Inherited from DynamicSymbol [▶ 2266].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.142.1 DynamicInterfaceInstance Properties

The [DynamicInterfaceInstance](#) [[▶ 2995](#)] type exposes the following members.

Properties

	Name	Description
	_InnerSymbol [▶ 2274]	Gets the inner symbol of this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	AccessRights [▶ 2274]	Gets the access rights. (Inherited from DynamicSymbol [▶ 2266].)
	AllowGIOAccess [▶ 2275]	Indicates, that the aggregates symbols is an IProcessImageAddress (and most probably IAdsSymbol) (Inherited from DynamicSymbol [▶ 2266].)
	Attributes [▶ 2275]	Gets the Symbol Attributes (Inherited from DynamicSymbol [▶ 2266].)
	BitSize [▶ 2276]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from DynamicSymbol [▶ 2266].)
	ByteSize [▶ 2276]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from DynamicSymbol [▶ 2266].)
	Category [▶ 2277]	Gets the category. (Inherited from DynamicSymbol [▶ 2266].)
	Comment [▶ 2277]	Gets the comment of the IInstance [▶ 2549] (Inherited from DynamicSymbol [▶ 2266].)
	Connection [▶ 2277]	Gets the connection bound to this DynamicSymbol [▶ 2266] (Inherited from DynamicSymbol [▶ 2266].)
	DataType [▶ 2278]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	HasRpcMethods [▶ 3001]	Gets a value indicating whether this instance has RPC methods
	HasValue [▶ 2278]	Gets a value indicating whether this instance has value (only the non VirtualInstances , what means the Symbols with locations). (Inherited from DynamicSymbol [▶ 2266].)
	InstanceName [▶ 2279]	Gets the name of the instance (without periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	InstancePath [▶ 2279]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from DynamicSymbol [▶ 2266].)
	IsBitType [▶ 2280]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	IsByteAligned [▶ 2280]	Indicates that the Size of the Object is Byte aligned (BitSize % 8 == 0) (Inherited from DynamicSymbol [▶ 2266].)
	IsContainerType [▶ 2281]	Gets a value indicating whether this Symbol is a container type. (Inherited from DynamicSymbol [▶ 2266].)
	IsPersistent [▶ 2281]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from DynamicSymbol [▶ 2266].)
	IsPointer [▶ 2282]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsPrimitiveType [▶ 2282]	Gets a value indicating whether this instance is a primitive type. (Inherited from DynamicSymbol [▶ 2266].)
	IsReadOnly [▶ 2283]	Indicates that this instance is read only. (Inherited from DynamicSymbol [▶ 2266].)
	IsRecursive [▶ 2283]	Gets a value indicating whether this instance is recursive. (Inherited from DynamicSymbol [▶ 2266].)
	IsReference [▶ 2284]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from DynamicSymbol [▶ 2266].)
	IsStatic [▶ 2284]	Gets a value indicating whether this instance is static. (Inherited from DynamicSymbol [▶ 2266].)
	MemberInstances [▶ 3002]	Gets the member instances of the Struct Instance [▶ 2666].
	NormalizedName [▶ 2285]	Gets the normalized instance name (fixed name for dynamic property access that doesn't contain invalid characters), (Inherited from DynamicSymbol [▶ 2266].)
	NotificationSettings [▶ 2285]	Gets the notification settings. (Inherited from DynamicSymbol [▶ 2266].)
	Parent [▶ 2286]	Gets the parent Symbol (Inherited from DynamicSymbol [▶ 2266].)
	RpcMethods [▶ 3003]	Gets the Method descriptions for the IRpcCallableType [▶ 2624]
	Size [▶ 2286]	Gets the size of the IInstance [▶ 2549] in bytes. (Inherited from DynamicSymbol [▶ 2266].)
	SubSymbols [▶ 2287]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from DynamicSymbol [▶ 2266].)
	TypeName [▶ 2287]	Gets the name of the DataType [▶ 2475] that is used for this IInstance [▶ 2549]. (Inherited from DynamicSymbol [▶ 2266].)
	ValueEncoding [▶ 2288]	Gets the value encoding. (Inherited from DynamicSymbol [▶ 2266].)

Reference

[DynamicInterfaceInstance Class](#) [▶ 2995]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.142.1.1 DynamicInterfaceInstance.HasRpcMethods Property

Gets a value indicating whether this instance has RPC methods

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool HasRpcMethods { get; }
```

Property Value

Type: Boolean
true if this instance has RPC methods; otherwise, false.

Implements

[IInterfaceInstance.HasRpcMethods](#) [[▶ 3018](#)]

Remarks

If the struct instance supports RPC Methods, then the instance class is also supporting [IRpcStructInstance](#) [[▶ 2642](#)]

Reference

[DynamicInterfaceInstance Class](#) [[▶ 2995](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

[TwinCAT.TypeSystem.IRpcStructInstance](#) [[▶ 2642](#)]

[TwinCAT.TypeSystem.IRpcMethod](#) [[▶ 2625](#)]

[TwinCAT.TypeSystem.IRpcMethodParameter](#) [[▶ 2635](#)]

6.11.142.1.2 DynamicInterfaceInstance.MemberInstances Property

Gets the member instances of the [Struct Instance](#) [[▶ 2666](#)].

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ISymbolCollection<ISymbol> MemberInstances { get; }
```

Property Value

Type: [ISymbolCollection](#) [[▶ 2700](#)].[ISymbol](#) [[▶ 2691](#)].
The member instances.

Implements

[IInterfaceInstance.MemberInstances](#) [[▶ 3018](#)]

Reference

[DynamicInterfaceInstance Class](#) [[▶ 2995](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.142.1.3 DynamicInterfaceInstance.RpcMethods Property

Gets the Method descriptions for the [IRpcCallableType](#) [[▶ 2624](#)]

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IRpcMethodCollection RpcMethods { get; }
```

Property Value

Type: [IRpcMethodCollection](#) [[▶ 2629](#)]

The methods.

Implements

[IRpcCallableInstance.RpcMethods](#) [[▶ 2607](#)]

Reference

[DynamicInterfaceInstance Class](#) [[▶ 2995](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.142.2 DynamicInterfaceInstance Methods

The [DynamicInterfaceInstance](#) [[▶ 2995](#)] type exposes the following members.

Methods

	Name	Description
	Equals [▶ 2292]	Equals (Inherited from DynamicSymbol [▶ 2266].)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetDynamicMemberNames [▶ 3006]	Returns the enumeration of all dynamic member names. (Overrides DynamicSymbol.GetDynamicMemberNames . [▶ 2293].)
	GetHashCode [▶ 2293]	Gets the HashCode of the Address (Inherited from DynamicSymbol [▶ 2266].)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	InvokeRpcMethodAsync [▶ 3007]	Invokes the specified RPC Method asynchronously
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	OnReadAnyValue [▶ 2293]	Handler function for reading ADS 'Any' Values. (Inherited from DynamicSymbol [▶ 2266].)

Name	Description
OnReadRawValue [▶ 2294]	Handler function for reading Raw symbol value. (Inherited from DynamicSymbol [▶ 2266].)
OnReadRawValueAsync [▶ 2294]	Handler function reading the raw value of the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
OnReadValue [▶ 2295]	Handler function for the (Inherited from DynamicSymbol [▶ 2266].)
OnReadValueAsync [▶ 2296]	Handler function reading the DynamicSymbols [▶ 2266] value asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
OnSetInstanceName [▶ 2296]	Sets a new InstanceName InstancePath (Inherited from DynamicSymbol [▶ 2266].)
OnTryReadValue [▶ 2297]	Handler function for the (Inherited from DynamicSymbol [▶ 2266].)
OnTryWriteValue [▶ 2297]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
OnWriteRawValue [▶ 2298]	Handler function for reading symbols raw value. (Inherited from DynamicSymbol [▶ 2266].)
OnWriteRawValueAsync [▶ 2298]	Handler function for writing the raw DynamicSymbol [▶ 2266] value. (Inherited from DynamicSymbol [▶ 2266].)
OnWriteValue [▶ 2299]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
OnWriteValueAsync [▶ 2300]	Handler Function for writing value. (Inherited from DynamicSymbol [▶ 2266].)
ReadAnyValue [▶ 2300]	Reads the value of this Value [▶ 2775] into a new created instance of the managed type (Inherited from DynamicSymbol [▶ 2266].)
ReadRawValue. [▶ 2301]	Reads the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
ReadRawValue(Int32) [▶ 2302]	Reads the Symbols raw value (Inherited from DynamicSymbol [▶ 2266].)
ReadRawValueAsync [▶ 2302]	Read raw value as an asynchronous operation. (Inherited from DynamicSymbol [▶ 2266].)
ReadValue. [▶ 2303]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
ReadValue(Int32) [▶ 2306]	Reads the value of this DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
ReadValueAsync [▶ 2307]	Reads the Value of the IValueSymbol [▶ 2775] asynchronously. (Inherited from DynamicSymbol [▶ 2266].)
ToString [▶ 2307]	Returns a String that represents this instance. (Inherited from DynamicSymbol [▶ 2266].)
TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
TryConvert	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicObject .)

	Name	Description
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetIndex	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicObject.)
	TryGetMember [▶ 3008]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Overrides DynamicSymbol.TryGetMember(GetMemberBinder, Object.) [▶ 2308].)
	TryInvoke	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicObject.)
	TryInvokeMember [▶ 3009]	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Overrides DynamicObject.TryInvokeMember(InvokeMemberBinder, Object, Object.) .)
	TryInvokeRpcMethod(String, Object, Object.) [▶ 3010]	Invokes the specified RPC Method
	TryInvokeRpcMethod(String, Object, Object, Object.) [▶ 3012]	Invokes the specified RPC Method
	TryReadValue [▶ 2309]	Reads the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	TrySetIndex	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicObject.)
	TrySetMember [▶ 3013]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Overrides DynamicObject.TrySetMember(SetMemberBinder, Object.) .)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
	TryWriteValue [▶ 2309]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	UpdateAnyValue [▶ 2312]	Reads the value of this Value [▶ 2775] into the specified managed value. (Inherited from DynamicSymbol [▶ 2266].)

	Name	Description
	WriteAnyValue [▶ 2313]	Writes the value represented by the managed value to this Value [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValue(Byte, Int32) [▶ 2314]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteRawValueAsync [▶ 2315]	Writes the raw value of the IValueSymbol [▶ 2775] (Ads Read / Write) (Inherited from DynamicSymbol [▶ 2266].)
	WriteValue(Object) [▶ 2316]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	WriteValue(Object, Int32) [▶ 2319]	Writes the specified value to the DynamicSymbol [▶ 2266]. (Inherited from DynamicSymbol [▶ 2266].)
	WriteValueAsync [▶ 2321]	Writes the Value of the IValueSymbol [▶ 2775] (Inherited from DynamicSymbol [▶ 2266].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[DynamicInterfaceInstance Class](#) [▶ 2995]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.142.2.1 DynamicInterfaceInstance.GetDynamicMemberNames Method

Returns the enumeration of all dynamic member names.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override IEnumerable<string> GetDynamicMemberNames()
```

Return Value

Type: [IEnumerable.String](#).

A sequence that contains dynamic member names.

Reference

[DynamicInterfaceInstance Class](#) [▶ 2995]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.142.2.2 DynamicInterfaceInstance.InvokeRpcMethodAsync Method

Invokes the specified RPC Method asynchronously

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Task<ResultRpcMethodAccess> InvokeRpcMethodAsync(
    string methodName,
    Object[]? inParameters,
    CancellationToken cancel
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object. The parameters.
cancel	Type: System.Threading.CancellationToken The cancellation token

Return Value

Type: [Task.ResultRpcMethodAccess](#) [[▸ 3213](#)].

A task that represents the asynchronous 'InvokeRpcMethod' operation. The [ResultRpcMethodAccess](#) [[▸ 3213](#)] results contains the return value ([ReturnValue](#) [[▸ 3216](#)]) together with the output parameters. The succeeded communication is indicated by the [ErrorCode](#) property ([ErrorCode](#) [[▸ 3202](#)]) after the communication.

Implements

[IRpcCallableInstance.InvokeRpcMethodAsync\(String, .Object., CancellationToken\)](#) [[▸ 2614](#)]

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [[▸ 2094](#)] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);
        }
    }
}
```



```

SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.VirtualTree);
ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

// Get the Symbols (Dynamic Symbols)

IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
the PLC Program

// Call a Method that has the following signature (within MAIN Program)
/* {attribute 'TcRpcEnable'}
METHOD PUBLIC M_Add : INT
VAR_INPUT
    i1 : INT := 0;
    i2 : INT := 0;
END_VAR
*/

short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

// Call a Method that has no parameter and returns VOID
main.InvokeRpcMethod("M_Method1", new object[] {});

//Browsing RpcMethods
foreach(IRpcMethod method in main.RpcMethods)
{
    string methodName = method.Name;

    foreach(IRpcMethodParameter parameter in method.Parameters)
    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
}
}

```

Reference

[DynamicInterfaceInstance Class \[► 2995\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.142.2.3 DynamicInterfaceInstance.TryGetMember Method

Provides the implementation for operations that get member values. Classes derived from the `DynamicObject` class can override this method to specify dynamic behavior for operations such as getting a value for a property.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public override bool TryGetMember(
    GetMemberBinder binder,
    out Object?? result
)

```

Parameters

binder	Type: <code>System.Dynamic.GetMemberBinder</code> Provides information about the object that called the dynamic operation. The <code>binder.Name</code> property provides the name of the member on which the dynamic operation is performed. For example, for the <code>Console.WriteLine(sampleObject.SampleProperty)</code> statement, where <code>sampleObject</code> is an instance of the class derived from the <code>DynamicObject</code> class, <code>binder.Name</code> returns "SampleProperty". The <code>binder.IgnoreCase</code> property specifies whether the member name is case-sensitive.
--------	---

result	Type: System.Object. The result of the get operation. For example, if the method is called for a property, you can assign the property value to result.
--------	--

Return Value

Type: Boolean
true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a run-time exception is thrown.)

Reference

[DynamicInterfaceInstance Class \[► 2995\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.142.2.4 DynamicInterfaceInstance.TryInvokeMember Method

Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool TryInvokeMember(
    InvokeMemberBinder binder,
    Object?[]? args,
    out Object?? returnValue
)
```

Parameters

binder	Type: System.Dynamic.InvokeMemberBinder Provides information about the dynamic operation. The binder.Name property provides the name of the member on which the dynamic operation is performed. For example, for the statement sampleObject.SampleMethod(100), where sampleObject is an instance of the class derived from the DynamicObject class, binder.Name returns "SampleMethod". The binder.IgnoreCase property specifies whether the member name is case-sensitive.
args	Type: .System.Object. The arguments that are passed to the object member during the invoke operation. For example, for the statement sampleObject.SampleMethod(100), where sampleObject is derived from the DynamicObject class, args[] is equal to 100.
returnValue	Type: System.Object. The result of the member invocation.

Return Value

Type: Boolean
true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a language-specific run-time exception is thrown.)

Reference

[DynamicInterfaceInstance Class \[► 2995\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.142.2.5 DynamicInterfaceInstance.TryInvokeRpcMethod Method

Overload List

	Name	Description
	TryInvokeRpcMethod(String, .Object., Object.) [▶ 3010]	Invokes the specified RPC Method
	TryInvokeRpcMethod(String, .Object., .Object., Object.) [▶ 3012]	Invokes the specified RPC Method

Reference

[DynamicInterfaceInstance Class](#) [[▶ 2995](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.142.2.5.1 DynamicInterfaceInstance.TryInvokeRpcMethod Method (String, .Object., Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int TryInvokeRpcMethod(
    string methodName,
    Object[]? inParameters,
    out Object?? retValue
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object. The parameters.
retValue	Type: System.Object. The return value of the RPC method as object.

Return Value

Type: Int32

The return value of the Method (as object).

Implements

[IRpcCallableInstance.TryInvokeRpcMethod\(String, .Object., Object.\)](#) [[▶ 2619](#)]

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier \[► 2094\]](#) specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
            client.Connect(address);

            SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.VirtualTree);
            ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

            // Get the Symbols (Dynamic Symbols)

            IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
            the PLC Program

            // Call a Method that has the following signature (within MAIN Program)
            /* {attribute 'TcRpcEnable'}
            METHOD PUBLIC M_Add : INT
            VAR_INPUT
                i1 : INT := 0;
                i2 : INT := 0;
            END_VAR
            */

            short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

            // Call a Method that has no parameter and returns VOID
            main.InvokeRpcMethod("M_Method1", new object[] {});

            //Browsing RpcMethods
            foreach(IRpcMethod method in main.RpcMethods)
            {
                string methodName = method.Name;

                foreach(IRpcMethodParameter parameter in method.Parameters)
                {
                    string parameterName = parameter.Name;
                    string parameterType = parameter.TypeName;
                }
            }
        }
    }
}
```

Reference

[DynamicInterfaceInstance Class \[► 2995\]](#)

[TryInvokeRpcMethod Overload \[► 3010\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.142.2.5.2 DynamicInterfaceInstance.TryInvokeRpcMethod Method (String, .Object., .Object., Object.)

Invokes the specified RPC Method

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int TryInvokeRpcMethod(
    string methodName,
    Object[]? inParameters,
    out Object[]? outParameters,
    out Object?? retValue
)
```

Parameters

methodName	Type: System.String The method name.
inParameters	Type: .System.Object. The parameters.
outParameters	Type: .System.Object.. The out parameters.
retValue	Type: System.Object. The return value of the RPC method as object.

Return Value

Type: Int32

The ADS Error Code.

Implements

[IRpcCallableInstance.TryInvokeRpcMethod\(String, .Object., .Object., Object.\)](#) [[▶ 2620](#)]

Remarks

Because this overload doesn't provide any [AnyTypeSpecifier](#) [[▶ 2094](#)] specifications, only primitive datatypes will be correctly marshalled by this method. Complex types will fall back to byte[] arrays.

Examples

The following sample shows how to call (Remote Procedures / Methods) within the PLC.

Dynamic Tree Mode

```
class RpcCallVirtualProgram
{
    /// <summary>
    /// Defines the entry point of the application.
    /// </summary>
    /// <param name="args">The arguments.</param>
    static void Main(string[] args)
    {
        // Get the AdsAddress from command-line arguments
        AmsAddress address = ArgParser.Parse(args);

        using (AdsClient client = new AdsClient())
        {
            //client.Synchronize = false;

            // Connect to the target device
```

```

client.Connect(address);

SymbolLoaderSettings settings = new SymbolLoaderSettings(SymbolsLoadMode.VirtualTree);
ISymbolLoader loader = SymbolLoaderFactory.Create(client, settings);

// Get the Symbols (Dynamic Symbols)

IStructInstance main = (IStructInstance)loader.Symbols["MAIN"]; // Gets the MAIN Instance of
the PLC Program

// Call a Method that has the following signature (within MAIN Program)
/* {attribute 'TcRpcEnable'}
METHOD PUBLIC M_Add : INT
VAR_INPUT
    i1 : INT := 0;
    i2 : INT := 0;
END_VAR
*/

short result = (short)main.InvokeRpcMethod("M_Add", new object[] {(short) 3, (short) 4});

// Call a Method that has no parameter and returns VOID
main.InvokeRpcMethod("M_Method1", new object[] {});

//Browsing RpcMethods
foreach(IRpcMethod method in main.RpcMethods)
{
    string methodName = method.Name;

    foreach(IRpcMethodParameter parameter in method.Parameters)
    {
        string parameterName = parameter.Name;
        string parameterType = parameter.TypeName;
    }
}
}
}

```

Reference

[DynamicInterfaceInstance Class \[► 2995\]](#)

[TryInvokeRpcMethod Overload \[► 3010\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.142.2.6 DynamicInterfaceInstance.TrySetMember Method

Provides the implementation for operations that set member values. Classes derived from the `DynamicObject` class can override this method to specify dynamic behavior for operations such as setting a value for a property.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public override bool TrySetMember(
    SetMemberBinder binder,
    Object? value
)

```

Parameters

binder	Type: <code>System.Dynamic.SetMemberBinder</code> Provides information about the object that called the dynamic operation. The <code>binder.Name</code> property provides the name of the member to which the value is being assigned. For example, for the statement <code>sampleObject.SampleProperty =</code>
--------	---

	"Test", where sampleObject is an instance of the class derived from the DynamicObject class, binder.Name returns "SampleProperty". The binder.IgnoreCase property specifies whether the member name is case-sensitive.
value	Type: System.Object The value to set to the member. For example, for sampleObject.SampleProperty = "Test", where sampleObject is an instance of the class derived from the DynamicObject class, the value is "Test".

Return Value

Type: Boolean

true if the operation is successful; otherwise, false. If this method returns false, the run-time binder of the language determines the behavior. (In most cases, a language-specific run-time exception is thrown.)

Reference

[DynamicInterfaceInstance Class \[► 2995\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.142.3 DynamicInterfaceInstance Events

The [DynamicInterfaceInstance \[► 2995\]](#) type exposes the following members.

Events

	Name	Description
	RawValueChanged [► 2323]	Occurs when the RawValue of the IValueSymbol [► 2775] has changed. (Inherited from DynamicSymbol [► 2266] .)
	ValueChanged [► 2323]	Occurs when the (Primitive) value of the IValueSymbol [► 2775] has changed. (Inherited from DynamicSymbol [► 2266] .)

Reference

[DynamicInterfaceInstance Class \[► 2995\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.142.4 DynamicInterfaceInstance Fields

The [DynamicInterfaceInstance \[► 2995\]](#) type exposes the following members.

Fields

	Name	Description
	syncObject [► 2327]	Synchronization object (Inherited from DynamicSymbol [► 2266] .)

Reference

[DynamicInterfaceInstance Class \[► 2995\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.143 IInterfaceInstance Interface

Interface representing an instance of a [IInterfaceType \[► 2983\]](#)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public interface IInterfaceInstance : ISymbol,
    IAttributedInstance, IInstance, IBitSize, IRpcCallableInstance
```

The IInterfaceInstance type exposes the following members.

Properties

Name	Description
Attributes [▸ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▸ 2469].)
BitSize [▸ 2473]	Gets the size of the IDataType [▸ 2475] in bits. (Inherited from IBitSize [▸ 2472].)
ByteSize [▸ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▸ 2472].)
Category [▸ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▸ 2691].)
Comment [▸ 2550]	Gets the comment of the IInstance [▸ 2549] (Inherited from IInstance [▸ 2549].)
DataType [▸ 2551]	Gets the IDataType [▸ 2475] of the IInstance [▸ 2549] . (Inherited from IInstance [▸ 2549].)
HasRpcMethods [▸ 3018]	Gets a value indicating whether this instance has RPC methods
InstanceName [▸ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▸ 2549].)
InstancePath [▸ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▸ 2549].)
IsBitType [▸ 2474]	Gets a value indicating whether this instance is not basing on a full IDataType but instead of some sort of bit mapping (Inherited from IBitSize [▸ 2472].)
IsByteAligned [▸ 2475]	Indicates that the Size of the Object is Byte aligned ($BitSize \% 8 == 0$) (Inherited from IBitSize [▸ 2472].)
IsContainerType [▸ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▸ 2691].)
IsPersistent [▸ 2694]	Gets a value indicating whether this ISymbol [▸ 2691] is persistent. (Inherited from ISymbol [▸ 2691].)
IsPointer [▸ 2552]	Indicates that the IInstance [▸ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▸ 2549].)
IsPrimitiveType [▸ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▸ 2691].)
IsReadOnly [▸ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▸ 2691].)
IsRecursive [▸ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▸ 2691].)
IsReference [▸ 2552]	Indicates that the IInstance [▸ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▸ 2549].)
IsStatic [▸ 2553]	Gets a value indicating whether this IInstance [▸ 2549] is static. (Inherited from IInstance [▸ 2549].)

Name	Description
MemberInstances [▶ 3018]	Gets the member instances of the Struct Instance.
Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
RpcMethods [▶ 2607]	Gets the Method descriptions for the IRpcCallableType [▶ 2624] (Inherited from IRpcCallableInstance [▶ 2606].)
Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)
TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from Instance [▶ 2549].)
ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from IAttributedInstance [▶ 2469].)

Methods

Name	Description
InvokeRpcMethod(String, Object.) [▶ 2609]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)
InvokeRpcMethod(String, Object., Object.) [▶ 2610]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)
InvokeRpcMethodAsync [▶ 2614]	Invokes the specified RPC Method asynchronously (Inherited from IRpcCallableInstance [▶ 2606].)
TryInvokeRpcMethod(String, Object., Object.) [▶ 2619]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)
TryInvokeRpcMethod(String, Object., Object., Object.) [▶ 2620]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)

Extension Methods

Name	Description
ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.143.1 InterfacelInstance Properties

The [IInterfacelInstance](#) [▶ 3014] type exposes the following members.

Properties

Name	Description
Attributes [▶ 2471]	Gets the Type Attributes. (Inherited from IAttributedInstance [▶ 2469].)
BitSize [▶ 2473]	Gets the size of the IDataType [▶ 2475] in bits. (Inherited from IBitSize [▶ 2472].)
ByteSize [▶ 2474]	Gets the (aligned) size of of the Type/Instance in Bytes (Inherited from IBitSize [▶ 2472].)
Category [▶ 2694]	Gets the Symbol/Datatype Category (Inherited from ISymbol [▶ 2691].)
Comment [▶ 2550]	Gets the comment of the IInstance [▶ 2549] (Inherited from IInstance [▶ 2549].)
DataType [▶ 2551]	Gets the IDataType [▶ 2475] of the IInstance [▶ 2549]. (Inherited from IInstance [▶ 2549].)
HasRpcMethods [▶ 3018]	Gets a value indicating whether this instance has RPC methods
InstanceName [▶ 2551]	Gets the name of the instance (without periods (.)) (Inherited from IInstance [▶ 2549].)
InstancePath [▶ 2552]	Gets the relative / absolute access path to the instance (with periods (.)) (Inherited from IInstance [▶ 2549].)
IsBitType [▶ 2474]	Gets a value indicating whether this instance is not basing on a full DataType but instead of some sort of bit mapping (Inherited from IBitSize [▶ 2472].)
IsByteAligned [▶ 2475]	Indicates that the Size of the Object is Byte aligned ($\text{BitSize} \% 8 == 0$) (Inherited from IBitSize [▶ 2472].)
IsContainerType [▶ 2694]	Gets a value indicating whether this Symbol is acontainer type. (Inherited from ISymbol [▶ 2691].)
IsPersistent [▶ 2694]	Gets a value indicating whether this ISymbol [▶ 2691] is persistent. (Inherited from ISymbol [▶ 2691].)
IsPointer [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Pointer type (Pointer TO) (Inherited from IInstance [▶ 2549].)
IsPrimitiveType [▶ 2695]	Gets a value indicating whether this instance is a primitive type. (Inherited from ISymbol [▶ 2691].)
IsReadOnly [▶ 2695]	Indicates that this instance is read only. (Inherited from ISymbol [▶ 2691].)
IsRecursive [▶ 2696]	Gets a value indicating whether this instance is recursive. (Inherited from ISymbol [▶ 2691].)
IsReference [▶ 2552]	Indicates that the IInstance [▶ 2549] represents a Reference type (REFERENCE TO) (Inherited from IInstance [▶ 2549].)
IsStatic [▶ 2553]	Gets a value indicating whether this IInstance [▶ 2549] is static. (Inherited from IInstance [▶ 2549].)
MemberInstances [▶ 3018]	Gets the member instances of the Struct Instance [▶ 3014].
Parent [▶ 2696]	Gets the parent Symbol (Inherited from ISymbol [▶ 2691].)
RpcMethods [▶ 2607]	Gets the Method descriptions for the IRpcCallableType [▶ 2624] (Inherited from IRpcCallableInstance [▶ 2606].)
Size [▶ 2475]	Gets the size of the object in bytes or Bits dependant on IsBitType [▶ 2474] (Inherited from IBitSize [▶ 2472].)
SubSymbols [▶ 2696]	Gets the SubSymbols of the ISymbol [▶ 2691] (Inherited from ISymbol [▶ 2691].)

	Name	Description
	TypeName [▶ 2553]	Gets the name of the DataType [▶ 2475] that is used for this Instance [▶ 2549]. (Inherited from Instance [▶ 2549].)
	ValueEncoding [▶ 2472]	Gets the value encoding. (Inherited from AttributedInstance [▶ 2469].)

Reference

[IInterfaceInstance Interface](#) [[▶ 3014](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.143.1.1 IInterfaceInstance.HasRpcMethods Property

Gets a value indicating whether this instance has RPC methods

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
bool HasRpcMethods { get; }
```

Property Value

Type: Boolean

true if this instance has RPC methods; otherwise, false.

Remarks

If the struct instance supports RPC Methods, then the instance class is also supporting [IRpcStructInstance](#) [[▶ 2642](#)]:

Reference

[IInterfaceInstance Interface](#) [[▶ 3014](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

[TwinCAT.TypeSystem.IRpcStructInstance](#) [[▶ 2642](#)]

[TwinCAT.TypeSystem.IRpcMethod](#) [[▶ 2625](#)]

[TwinCAT.TypeSystem.IRpcMethodParameter](#) [[▶ 2635](#)]

6.11.143.1.2 IInterfaceInstance.MemberInstances Property

Gets the member instances of the [Struct Instance](#) [[▶ 3014](#)].

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISymbolCollection<ISymbol> MemberInstances { get; }
```

Property Value

Type: [ISymbolCollection](#) [[▶ 2700](#)]. [ISymbol](#) [[▶ 2691](#)].
The member instances.

Reference

[IInterfaceInstance Interface](#) [[▶ 3014](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.143.2 IInterfaceInstance Methods

The [IInterfaceInstance](#) [[▶ 3014](#)] type exposes the following members.

Methods

	Name	Description
	InvokeRpcMethod(String, Object.) [▶ 2609]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)
	InvokeRpcMethod(String, Object., Object.) [▶ 2610]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)
	InvokeRpcMethodAsync [▶ 2614]	Invokes the specified RPC Method asynchronously (Inherited from IRpcCallableInstance [▶ 2606].)
	TryInvokeRpcMethod(String, Object., Object.) [▶ 2619]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)
	TryInvokeRpcMethod(String, Object., Object., Object.) [▶ 2620]	Invokes the specified RPC Method (Inherited from IRpcCallableInstance [▶ 2606].)

Extension Methods

	Name	Description
	ReferencesExternalData [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block. (Defined by DataTypeExtension [▶ 3021].)

Reference

[IInterfaceInstance Interface](#) [[▶ 3014](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.144 SymbolIterationMask Enumeration

Mask Flagset to specify filters for [SymbolIterator.T.](#) [[▶ 3172](#)].

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
[FlagsAttribute]
public enum SymbolIterationMask
```

Members

	Member name	Value	Description
	None	0	Uninitialized / None
	Structures	1	Iterates over Subelements of Structs
	Arrays	2	Iterates over Elements of Arrays
	Unions	4	Iterates over Subelements of Unions
	Pointer	8	Iterates over Pointer SubElements
	References	16	Iterates over References
	All	31	Iterates over All Complex/Combined types
	AllWithoutArrayElements	29	Iterates over All Complex/Combined types excluding Array elements

Reference

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.145 AnyTypeValueCreator Class

Class AnyTypeValueCreator.

Inheritance Hierarchy

System.Object
 TwinCAT.TypeSystem.AnyTypeValueCreator

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static class AnyTypeValueCreator
```

The AnyTypeValueCreator type exposes the following members.

Methods

	Name	Description
	CreateValue [▸ 3021]	Creates a value from the specified type.

Reference

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.145.1 AnyTypeValueCreator Methods

The [AnyTypeValueCreator](#) [[▸ 3020](#)] type exposes the following members.

Methods

	Name	Description
	CreateValue [▸ 3021]	Creates a value from the specified type.

Reference

[AnyTypeValueCreator Class](#) [[▸ 3020](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.145.1.1 AnyTypeValueCreator.CreateValue Method

Creates a value from the specified type.

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static Object? CreateValue(
    AnyTypeSpecifier spec
)
```

Parameters

spec	Type: TwinCAT.TypeSystem.AnyTypeSpecifier [▸ 2094] The spec.
------	---

Return Value

Type: Object
System.Nullable<System.Object>.

Reference

[AnyTypeValueCreator Class](#) [[▸ 3020](#)]

[TwinCAT.TypeSystem Namespace](#) [[▸ 2083](#)]

6.11.146 DataTypeExtension Class

Extends the [IDataType](#) [[▸ 2475](#)] interface with more sophisticated functions.

Inheritance Hierarchy

System.Object
TwinCAT.TypeSystem.DataTypeExtension

Namespace: [TwinCAT.TypeSystem](#) [[▸ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static class DataTypeExtension
```

Methods

	Name	Description
	IsArrayOfPrimitives(DataType) [► 3025]	Determines whether the dataType is an array of Primitives.
	IsArrayOfPrimitives(DataType, Boolean) [► 3025]	Determines whether the dataType is an array of Primitives.
	IsContainer(DataTypeCategory) [► 3026]	Gets a value indicating whether this IDataType [► 2475] is a container type
	IsContainer(IDataType) [► 3027]	Gets a value indicating whether this IDataType [► 2475] is a container type
	IsContainer(IDataType, Boolean) [► 3028]	Gets a value indicating whether this IDataType [► 2475] is a container type
	IsPointer(DataTypeCategory) [► 3030]	Gets a value indicating whether this IDataType [► 2475] is a pointer type
	IsPointer(IDataType) [► 3031]	Gets a value indicating whether this IDataType [► 2475] is a pointer type
	IsPrimitive(AdsDataTypeId) [► 3032]	Determines whether the specified type identifier is primitive.
	IsPrimitive(DataTypeCategory) [► 3032]	Gets a value indicating whether this IDataType [► 2475] is primitive
	IsPrimitive(IDataType) [► 3033]	Gets a value indicating whether this IDataType [► 2475] is primitive
	IsPrimitive(IDataType, Boolean) [► 3034]	Gets a value indicating whether this IDataType [► 2475] is primitive
	IsReference(DataTypeCategory) [► 3035]	Gets a value indicating whether this IDataType [► 2475] is a reference type
	IsReference(IDataType) [► 3036]	Gets a value indicating whether this IDataType [► 2475] is a reference type
	ReferencesExternalData(IDataType) [► 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block.

Name	Description
ReferencesExternalData(IInstance) [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block.
ResolvableAsPrimitive(IDataType) [▶ 3039]	Indicates, that this data type can be resolved as primitive or primitive Array.
ResolvableAsPrimitive(IDataType, Boolean) [▶ 3039]	Indicates, that this data type can be resolved as primitive or primitive Array.
Resolve(IDataType) [▶ 3040]	Resolves the specified data type for AliasReference [▶ 3042]
Resolve(IDataType, DataTypeResolveStrategy) [▶ 3041]	Resolves the specified strategy.

Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.146.1 DataTypeExtension Methods

Methods

Name	Description
IsArrayOfPrimitives(IDataType) [▶ 3025]	Determines whether the dataType is an array of Primitives.
IsArrayOfPrimitives(IDataType, Boolean) [▶ 3025]	Determines whether the dataType is an array of Primitives.
IsContainer(DataTypeCategory) [▶ 3026]	Gets a value indicating whether this IDataType [▶ 2475] is a container type
IsContainer(IDataType) [▶ 3027]	Gets a value indicating whether this IDataType [▶ 2475] is a container type
IsContainer(IDataType, Boolean) [▶ 3028]	Gets a value indicating whether this IDataType [▶ 2475] is a container type
IsPointer(DataTypeCategory) [▶ 3030]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type
IsPointer(IDataType) [▶ 3031]	Gets a value indicating whether this IDataType [▶ 2475] is a pointer type
IsPrimitive(AdsDataTypeId) [▶ 3032]	Determines whether the specified type identifier is primitive.

	Name	Description
	IsPrimitive(DataType Category) [► 3032]	Gets a value indicating whether this IDataType [► 2475] is primitive
	IsPrimitive(IDataType) [► 3033]	Gets a value indicating whether this IDataType [► 2475] is primitive
	IsPrimitive(IDataType, Boolean) [► 3034]	Gets a value indicating whether this IDataType [► 2475] is primitive
	IsReference(DataTypeCategory) [► 3035]	Gets a value indicating whether this IDataType [► 2475] is a reference type
	IsReference(IDataType) [► 3036]	Gets a value indicating whether this IDataType [► 2475] is a reference type
	ReferencesExternalData(IDataType) [► 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block.
	ReferencesExternalData(IInstance) [► 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block.
	ResolvableAsPrimitive(IDataType) [► 3039]	Indicates, that this data type can be resolved as primitive or primitive Array.
	ResolvableAsPrimitive(IDataType, Boolean) [► 3039]	Indicates, that this data type can be resolved as primitive or primitive Array.
	Resolve(IDataType) [► 3040]	Resolves the specified data type for AliasReference [► 3042]
	Resolve(IDataType, DataTypeResolveStrategy) [► 3041]	Resolves the specified strategy.

Reference

[DataTypeExtension Class \[► 3021\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.146.1.1 DataTypeExtension.IsArrayOfPrimitives Method

Overload List

	Name	Description
	IsArrayOfPrimitives(IDataType) [► 3025]	Determines whether the dataType is an array of Primitives.

	Name	Description
	IsArrayOfPrimitives(I DataType, Boolean) [▶ 3025]	Determines whether the dataType is an array of Primitives.

Reference

[DataTypeExtension Class](#) [▶ 3021]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.146.1.1.1 DataTypeExtension.IsArrayOfPrimitives Method (IDataType)

Determines whether the dataType is an array of Primitives.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool IsArrayOfPrimitives(
    this IDataType type
)
```

Parameters

type	Type: TwinCAT.TypeSystem.IDataType [▶ 2475] The type.
------	--

Return Value

Type: Boolean
true if [is array of primitives] [the specified type]; otherwise, false.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IDataType](#) [▶ 2475]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[DataTypeExtension Class](#) [▶ 3021]

[IsArrayOfPrimitives Overload](#) [▶ 3024]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.146.1.1.2 DataTypeExtension.IsArrayOfPrimitives Method (IDataType, Boolean)

Determines whether the dataType is an array of Primitives.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool IsArrayOfPrimitives(
    this IDataType type,
    bool resolve
)
```

Parameters

type	Type: TwinCAT.TypeSystem.IDataType [► 2475] The type.
resolve	Type: System.Boolean if set to true the array datatype and element datatypes will be resolved.

Return Value

Type: Boolean
true if [is array of primitives] [the specified type]; otherwise, false.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IDataType](#) [► 2475]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[DataTypeExtension Class](#) [► 3021]

[IsArrayOfPrimitives Overload](#) [► 3024]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.146.1.2 DataTypeExtension.IsContainer Method

Overload List

	Name	Description
	IsContainer(DataTypeCategory) [► 3026]	Gets a value indicating whether this IDataType [► 2475] is a container type
	IsContainer(IDataType) [► 3027]	Gets a value indicating whether this IDataType [► 2475] is a container type
	IsContainer(IDataType, Boolean) [► 3028]	Gets a value indicating whether this IDataType [► 2475] is a container type

Reference

[DataTypeExtension Class](#) [► 3021]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

6.11.146.1.2.1 DataTypeExtension.IsContainer Method (DataTypeCategory)

Gets a value indicating whether this [IDataType](#) [► 2475] is a container type

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool IsContainer(  
    this DataTypeCategory cat  
)
```

Parameters

cat	Type: TwinCAT.TypeSystem.DataTypeCategory [▸ 2111]
-----	--

Field Value

Type: Boolean

true if this instance is container type; otherwise, false.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [DataTypeCategory](#) [▸ 2111]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

Container Types are all types that contain SubElements like

- [Array](#) [▸ 2111]
- [Pointer](#) [▸ 2111]
- [Union](#) [▸ 2111]
- [Struct](#) [▸ 2111]
- [Function](#) [▸ 2111]
- [FunctionBlock](#) [▸ 2111]
- [Program](#) [▸ 2111]

and the [Alias](#) [▸ 2111] and [Reference](#) [▸ 2111] types, if they have a container type as base type.

Reference

[DataTypeExtension Class](#) [▸ 3021]

[IsContainer Overload](#) [▸ 3026]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

[IDataType.Category](#) [▸ 2478]

6.11.146.1.2.2 DataTypeExtension.IsContainer Method (IDataType)

Gets a value indicating whether this [IDataType](#) [▸ 2475] is a container type

Namespace: [TwinCAT.TypeSystem](#) [▸ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool IsContainer(  
    this IDataTypes type  
)
```

Parameters

type	Type: TwinCAT.TypeSystem.IDataTypes [▶ 2475]
------	--

Field Value

Type: Boolean

true if this instance is container type; otherwise, false.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IDataTypes](#) [▶ 2475]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

Container Types are all types that contain SubElements like

- [Array](#) [▶ 2111]
- [Pointer](#) [▶ 2111]
- [Union](#) [▶ 2111]
- [Struct](#) [▶ 2111]
- [Function](#) [▶ 2111]
- [FunctionBlock](#) [▶ 2111]
- [Program](#) [▶ 2111]

and the [Alias](#) [▶ 2111] and [Reference](#) [▶ 2111] types, if they have a container type as base type.

Reference

[DataTypeExtension Class](#) [▶ 3021]

[IsContainer Overload](#) [▶ 3026]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

[IDataTypes.Category](#) [▶ 2478]

6.11.146.1.2.3 **DataTypeExtension.IsContainer Method (IDataTypes, Boolean)**

Gets a value indicating whether this [IDataTypes](#) [▶ 2475] is a container type

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool IsContainer(  
    this IDataTypeInfo type,  
    bool resolve  
)
```

Parameters

type	Type: TwinCAT.TypeSystem.IDataTypeInfo [▸ 2475] The type.
resolve	Type: System.Boolean if set to true [resolve].

Field Value

Type: Boolean
true if this instance is container type; otherwise, false.

Return Value

Type: Boolean
true if the specified resolve is container; otherwise, false.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IDataTypeInfo](#) [▸ 2475]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

Container Types are all types that contain SubElements like

- [Array](#) [▸ 2111]
- [Pointer](#) [▸ 2111]
- [Union](#) [▸ 2111]
- [Struct](#) [▸ 2111]
- [Function](#) [▸ 2111]
- [FunctionBlock](#) [▸ 2111]
- [Program](#) [▸ 2111]

and the [Alias](#) [▸ 2111] and [Reference](#) [▸ 2111] types, if they have a container type as base type.

Reference

[DataTypeExtension Class](#) [▸ 3021]

[IsContainer Overload](#) [▸ 3026]

[TwinCAT.TypeSystem Namespace](#) [▸ 2083]

[IDataTypeInfo.Category](#) [▸ 2478]

6.11.146.1.3 DataTypeExtension.IsPointer Method

Overload List

	Name	Description
	IsPointer(DataTypeCategory) [► 3030]	Gets a value indicating whether this IDataType [► 2475] is a pointer type
	IsPointer(IDataType) [► 3031]	Gets a value indicating whether this IDataType [► 2475] is a pointer type

Reference

[DataTypeExtension Class \[► 3021\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.146.1.3.1 DataTypeExtension.IsPointer Method (DataTypeCategory)

Gets a value indicating whether this [IDataType \[► 2475\]](#) is a pointer type

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool IsPointer(
    this DataTypeCategory cat
)
```

Parameters

cat	Type: TwinCAT.TypeSystem.DataTypeCategory [► 2111]
-----	--

Field Value

Type: Boolean

true if this instance is pointer type; otherwise, false.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [DataTypeCategory \[► 2111\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

Pointer types can be dereferenced with the '^' operator.

Reference

[DataTypeExtension Class \[► 3021\]](#)

[IsPointer Overload \[► 3030\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

[IDataType.Category](#) [[▶ 2478](#)]

6.11.146.1.3.2 **DataTypeExtension.IsPointer Method (IDataType)**

Gets a value indicating whether this [IDataType](#) [[▶ 2475](#)] is a pointer type

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool IsPointer(
    this IDataType type
)
```

Parameters

type	Type: TwinCAT.TypeSystem.IDataType [▶ 2475]
------	---

Field Value

Type: Boolean
true if this instance is pointer type; otherwise, false.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IDataType](#) [[▶ 2475](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

Pointer types can be dereferenced with the '^' operator.

Reference

[DataTypeExtension Class](#) [[▶ 3021](#)]

[IsPointer Overload](#) [[▶ 3030](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

[IDataType.Category](#) [[▶ 2478](#)]

6.11.146.1.4 **DataTypeExtension.IsPrimitive Method**

Overload List

	Name	Description
	IsPrimitive(AdsDataTypeId) [▶ 3032]	Determines whether the specified type identifier is primitive.
	IsPrimitive(DataTypeCategory) [▶ 3032]	Gets a value indicating whether this IDataType [▶ 2475] is primitive

	Name	Description
	IsPrimitive(IDataType) [▸ 3033]	Gets a value indicating whether this IDataType [▸ 2475] is primitive
	IsPrimitive(IDataType, Boolean) [▸ 3034]	Gets a value indicating whether this IDataType [▸ 2475] is primitive

Reference

[DataTypeExtension Class \[▸ 3021\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.146.1.4.1 DataTypeExtension.IsPrimitive Method (AdsDataTypeId)

Determines whether the specified type identifier is primitive.

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool IsPrimitive(
    this AdsDataTypeId typeId
)
```

Parameters

typeId	Type: TwinCAT.Ads.AdsDataTypeId [▸ 663] The type identifier.
--------	---

Return Value

Type: Boolean

true if the specified type identifier is primitive; otherwise, false.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [AdsDataTypeId \[▸ 663\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[DataTypeExtension Class \[▸ 3021\]](#)

[IsPrimitive Overload \[▸ 3031\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.146.1.4.2 DataTypeExtension.IsPrimitive Method (DataTypeCategory)

Gets a value indicating whether this [IDataType \[▸ 2475\]](#) is primitive

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool IsPrimitive(
    this DataTypeCategory cat
)
```

Parameters

cat	Type: TwinCAT.TypeSystem.DataTypeCategory [▶ 2111] The cat.
-----	--

Field Value

Type: Boolean

true if this instance is primitive; otherwise, false.

Return Value

Type: Boolean

true if the specified type is primitive; otherwise, false.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [DataTypeCategory](#) [▶ 2111]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[DataTypeExtension Class](#) [▶ 3021]

[IsPrimitive Overload](#) [▶ 3031]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.146.1.4.3 DataTypeExtension.IsPrimitive Method (IDataType)

Gets a value indicating whether this [IDataType](#) [▶ 2475] is primitive

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool IsPrimitive(
    this IDataType type
)
```

Parameters

type	Type: TwinCAT.TypeSystem.IDataType [▶ 2475] The data type.
------	---

Field Value

Type: Boolean

true if this instance is primitive; otherwise, false.

Return Value

Type: Boolean
true if the specified type is primitive; otherwise, false.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IDataType](#) [[▶ 2475](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[DataTypeExtension Class](#) [[▶ 3021](#)]

[IsPrimitive Overload](#) [[▶ 3031](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.146.1.4.4 DataTypeExtension.IsPrimitive Method (IDataType, Boolean)

Gets a value indicating whether this [IDataType](#) [[▶ 2475](#)] is primitive

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static bool IsPrimitive(
    this IDataType type,
    bool resolve
)
```

Parameters

type	Type: TwinCAT.TypeSystem.IDataType [▶ 2475] The dataType.
resolve	Type: System.Boolean if set to true the data type will be resolved.

Field Value

Type: Boolean
true if this instance is primitive; otherwise, false.

Return Value

Type: Boolean
true if the specified type is primitive; otherwise, false.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IDataType](#) [[▶ 2475](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[DataTypeExtension Class](#) [[▶ 3021](#)]

[IsPrimitive Overload \[▸ 3031\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.146.1.5 DataTypeExtension.IsReference Method

Overload List

	Name	Description
	IsReference(DataTypeCategory) [▸ 3035]	Gets a value indicating whether this IDataType [▸ 2475] is a reference type
	IsReference(IDataType) [▸ 3036]	Gets a value indicating whether this IDataType [▸ 2475] is a reference type

Reference

[DataTypeExtension Class \[▸ 3021\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.146.1.5.1 DataTypeExtension.IsReference Method (DataTypeCategory)

Gets a value indicating whether this [IDataType \[▸ 2475\]](#) is a reference type

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool IsReference(
    this DataTypeCategory cat
)
```

Parameters

cat	Type: TwinCAT.TypeSystem.DataTypeCategory [▸ 2111]
-----	--

Field Value

Type: Boolean
true if this instance is container type; otherwise, false.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [DataTypeCategory \[▸ 2111\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

Reference types can be dereferenced.

Reference

[DataTypeExtension Class](#) [► 3021]

[IsReference Overload](#) [► 3035]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

[IDataType.Category](#) [► 2478]

6.11.146.1.5.2 DataTypeExtension.IsReference Method (IDataType)

Gets a value indicating whether this [IDataType](#) [► 2475] is a reference type

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool IsReference(  
    this IDataType type  
)
```

Parameters

type	Type: TwinCAT.TypeSystem.IDataType [► 2475]
------	---

Field Value

Type: Boolean

true if this instance is container type; otherwise, false.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IDataType](#) [► 2475]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

Reference types can be dereferenced.

Reference

[DataTypeExtension Class](#) [► 3021]

[IsReference Overload](#) [► 3035]

[TwinCAT.TypeSystem Namespace](#) [► 2083]

[IDataType.Category](#) [► 2478]

6.11.146.1.6 DataTypeExtension.ReferencesExternalData Method

Overload List

	Name	Description
	ReferencesExternalData(IDataType) [▶ 3037]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block.
	ReferencesExternalData(IInstance) [▶ 3038]	Indicates, that the Instance value includes process memory data, that cannot be read in one memory block.

Reference

[DataTypeExtension Class](#) [▶ 3021]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.146.1.6.1 DataTypeExtension.ReferencesExternalData Method (IDataType)

Indicates, that the Instance value includes process memory data, that cannot be read in one memory block.

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool ReferencesExternalData(
    this IDataType type
)
```

Parameters

type	Type: TwinCAT.TypeSystem.IDataType [▶ 2475] The dataType.
------	--

Return Value

Type: Boolean

true if the value of this datatype is split into more than one memory block, false otherwise.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IDataType](#) [▶ 2475]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

This is the case, if the DataType contains static fields, Pointers or References that contain these types of instances/datatypes. The Value of the instance cannot be read in one ADS roundtrip.

Reference

[DataTypeExtension Class](#) [▶ 3021]

[ReferencesExternalData Overload \[▶ 3037\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.146.1.6.2 DataTypeExtension.ReferencesExternalData Method (IInstance)

Indicates, that the Instance value includes process memory data, that cannot be read in one memory block.

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool ReferencesExternalData(
    this IInstance instance
)
```

Parameters

instance	Type: TwinCAT.TypeSystem.IInstance [▶ 2549] The instance.
----------	--

Return Value

Type: Boolean

true if the value of the instance is split into more than one memory block, false otherwise.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IInstance \[▶ 2549\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Remarks

This is the case, if the DataType contains static fields, Pointers or References that contain these types of instances/datatypes. The Value of the instance cannot be read in one ADS roundtrip.

Reference

[DataTypeExtension Class \[▶ 3021\]](#)

[ReferencesExternalData Overload \[▶ 3037\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.146.1.7 DataTypeExtension.ResolvableAsPrimitive Method

Overload List

	Name	Description
	ResolvableAsPrimitive(IDataType) [▶ 3039]	Indicates, that this data type can be resolved as primitive or primitive Array.

	Name	Description
	ResolvableAsPrimitive(IDataType, Boolean) [▶ 3039]	Indicates, that this data type can be resolved as primitive or primitive Array.

Reference

[DataTypeExtension Class](#) [[▶ 3021](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.146.1.7.1 DataTypeExtension.ResolvableAsPrimitive Method (IDataType)

Indicates, that this data type can be resolved as primitive or primitive Array.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool ResolvableAsPrimitive(
    this IDataType type
)
```

Parameters

type	Type: TwinCAT.TypeSystem.IDataType [▶ 2475] The type.
------	--

Return Value

Type: Boolean
 true if resolvable as primitive, false otherwise.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IDataType](#) [[▶ 2475](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[DataTypeExtension Class](#) [[▶ 3021](#)]

[ResolvableAsPrimitive Overload](#) [[▶ 3038](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.146.1.7.2 DataTypeExtension.ResolvableAsPrimitive Method (IDataType, Boolean)

Indicates, that this data type can be resolved as primitive or primitive Array.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool ResolvableAsPrimitive(
    this IDataTypeInfo type,
    bool resolve
)
```

Parameters

type	Type: TwinCAT.TypeSystem.IDataTypeInfo [2475] The type.
resolve	Type: System.Boolean if set to true the datatype will be resolved.

Return Value

Type: Boolean
true if resolvable as primitive, false otherwise.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IDataTypeInfo](#) [[2475](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[DataTypeExtension Class](#) [[3021](#)]

[ResolvableAsPrimitive Overload](#) [[3038](#)]

[TwinCAT.TypeSystem Namespace](#) [[2083](#)]

6.11.146.1.8 DataTypeExtension.Resolve Method

Overload List

	Name	Description
	Resolve(IDataTypeInfo) [3040]	Resolves the specified data type for AliasReference [3042]
	Resolve(IDataTypeInfo, DataTypeResolveStrategy) [3041]	Resolves the specified strategy.

Reference

[DataTypeExtension Class](#) [[3021](#)]

[TwinCAT.TypeSystem Namespace](#) [[2083](#)]

6.11.146.1.8.1 DataTypeExtension.Resolve Method (IDataTypeInfo)

Resolves the specified data type for [AliasReference](#) [[3042](#)]

Namespace: [TwinCAT.TypeSystem](#) [[2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IDataType Resolve(
    this IDataType dataType
)
```

Parameters

dataType	Type: TwinCAT.TypeSystem.IDataType [▶ 2475] Type of the data.
----------	---

Return Value

Type: [IDataType](#) [▶ [2475](#)]
IDataType.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IDataType](#) [▶ [2475](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[DataTypeExtension Class](#) [▶ [3021](#)]

[Resolve Overload](#) [▶ [3040](#)]

[TwinCAT.TypeSystem Namespace](#) [▶ [2083](#)]

6.11.146.1.8.2 DataTypeExtension.Resolve Method (IDataType, DataTypeResolveStrategy)

Resolves the specified strategy.

Namespace: [TwinCAT.TypeSystem](#) [▶ [2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IDataType Resolve(
    this IDataType dataType,
    DataTypeResolveStrategy strategy
)
```

Parameters

dataType	Type: TwinCAT.TypeSystem.IDataType [▶ 2475] Type of the data.
strategy	Type: TwinCAT.TypeSystem.DataTypeResolveStrategy [▶ 3042] The strategy.

Return Value

Type: [IDataType](#) [▶ [2475](#)]
IDataType.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IDataType](#) [[▶ 2475](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[DataTypeExtension Class](#) [[▶ 3021](#)]

[Resolve Overload](#) [[▶ 3040](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.147 DataTypeResolveStrategy Enumeration

Enum [DataTypeResolveStrategy](#)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum DataTypeResolveStrategy
```

Members

	Member name	Value	Description
	Alias	0	Resolve only chains of Alias(es)
	AliasReference	1	Resolve chains of Alias(es) and Reference(es)

Remarks

Used for resolving Reference and Alias datatypes to its base datatypes.

Reference

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.148 DynamicArrayValue Class

Class [DynamicArrayValue](#). Implements the [DynamicValue](#) [[▶ 2347](#)]

Inheritance Hierarchy

System.Object

System.Dynamic.DynamicObject

[TwinCAT.TypeSystem.DynamicValue](#) [[▶ 2347](#)]

[TwinCAT.TypeSystem.DynamicArrayValue](#)

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public class DynamicArrayValue : DynamicValue
```

The DynamicArrayValue type exposes the following members.

Properties

Name	Description
Age [▶ 2351]	Gets the age of the value (last successful read of the value) (Inherited from DynamicValue [▶ 2347].)
CachedRaw [▶ 2351]	Gets the cached Raw internal Data. (Inherited from DynamicValue [▶ 2347].)
DataType [▶ 2352]	Gets the data type bound to this IValue [▶ 2745] (Inherited from DynamicValue [▶ 2347].)
IsPrimitive [▶ 2353]	Gets a value indicating whether this IValue [▶ 2745] is a primitive value. (Inherited from DynamicValue [▶ 2347].)
ParentValue [▶ 2356]	Gets the parent value. (Inherited from DynamicValue [▶ 2347].)
ResolvedType [▶ 2353]	Gets the resolved type. (Inherited from DynamicValue [▶ 2347].)
Symbol [▶ 2353]	Gets the symbol that is bound to this value. (Inherited from DynamicValue [▶ 2347].)
TimeStamp [▶ 2354]	Gets the Time stamp of the last successful read of the Value. (Inherited from DynamicValue [▶ 2347].)
UpdateMode [▶ 2354]	Gets / Sets the update mode (Inherited from DynamicValue [▶ 2347].)
ValueFactory [▶ 2355]	The value factory (Inherited from DynamicValue [▶ 2347].)

Methods

Name	Description
Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
GetDynamicMemberNames [▶ 2358]	Returns the enumeration of all dynamic member names. (Inherited from DynamicValue [▶ 2347].)
GetHashCode	Serves as the default hash function. (Inherited from Object.)
GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
GetType	Gets the Type of the current instance. (Inherited from Object.)
MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
Read [▶ 2359]	Reads the value (via ADS) (Inherited from DynamicValue [▶ 2347].)
ReadAsync [▶ 2359]	read as an asynchronous operation. (Inherited from DynamicValue [▶ 2347].)
ResolveValue [▶ 2360]	Resolves the Value object to its primitive value. (Inherited from DynamicValue [▶ 2347].)
setMemberValue [▶ 2374]	Writes the specified member element. (Inherited from DynamicValue [▶ 2347].)

	Name	Description
	ToString [▶ 2361]	Returns a String that represents this instance. (Inherited from DynamicValue [▶ 2347].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert [▶ 2361]	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicValue [▶ 2347].)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)
	TryGetArrayElement Values [▶ 2362]	Returns Array Element values. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndex [▶ 2363]	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndexValue(.Int32., Object.) [▶ 2364]	Reads the specified array element. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndexValue(.Object., Object.) [▶ 2364]	Tries the get index value. (Inherited from DynamicValue [▶ 2347].)
	TryGetMember [▶ 2365]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Inherited from DynamicValue [▶ 2347].)
	TryGetMemberValue [▶ 2366]	Tries the get member value. (Inherited from DynamicValue [▶ 2347].)
	TryInvoke [▶ 2366]	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicValue [▶ 2347].)
	TryInvokeMember [▶ 2367]	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicValue [▶ 2347].)
	TryResolveValue [▶ 2368]	Tries to resolve the Value object to its primitive value. (Inherited from DynamicValue [▶ 2347].)

Name	Description
TrySetIndex [▶ 2369]	Provides the implementation for operations that set a value by index. Classes derived from the <code>DynamicObject</code> class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicValue [▶ 2347].)
TrySetIndexValue(Int32, Object) [▶ 2370]	Tries to set the indexed value on Arrays (Inherited from DynamicValue [▶ 2347].)
TrySetIndexValue(Object, Object) [▶ 2370]	Tries to set the indexed value on Arrays (Inherited from DynamicValue [▶ 2347].)
TrySetMember [▶ 2371]	Provides the implementation for operations that set member values. Classes derived from the <code>DynamicObject</code> class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicValue [▶ 2347].)
TrySetMemberValue [▶ 2372]	Tries to Set a Member/Property Value (Inherited from DynamicValue [▶ 2347].)
TryUnaryOperation	Provides implementation for unary operations. Classes derived from the <code>DynamicObject</code> class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from <code>DynamicObject</code> .)
Write [▶ 2372]	Writes the value (via ADS) (Inherited from DynamicValue [▶ 2347].)
WriteAsync [▶ 2373]	write as an asynchronous operation. (Inherited from DynamicValue [▶ 2347].)

Reference

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

[TwinCAT.TypeSystem.DynamicValue](#) [▶ 2347]

6.11.148.1 DynamicArrayValue Properties

The [DynamicArrayValue](#) [▶ 3042] type exposes the following members.

Properties

Name	Description
Age [▶ 2351]	Gets the age of the value (last successful read of the value) (Inherited from DynamicValue [▶ 2347].)
CachedRaw [▶ 2351]	Gets the cached Raw internal Data. (Inherited from DynamicValue [▶ 2347].)
DataType [▶ 2352]	Gets the data type bound to this IValue [▶ 2745] (Inherited from DynamicValue [▶ 2347].)
IsPrimitive [▶ 2353]	Gets a value indicating whether this IValue [▶ 2745] is a primitive value. (Inherited from DynamicValue [▶ 2347].)
ParentValue [▶ 2356]	Gets the parent value. (Inherited from DynamicValue [▶ 2347].)
ResolvedType [▶ 2353]	Gets the resolved type. (Inherited from DynamicValue [▶ 2347].)
Symbol [▶ 2353]	Gets the symbol that is bound to this value. (Inherited from DynamicValue [▶ 2347].)

	Name	Description
	TimeStamp [▶ 2354]	Gets the Time stamp of the last successful read of the Value. (Inherited from DynamicValue [▶ 2347].)
	UpdateMode [▶ 2354]	Gets / Sets the update mode (Inherited from DynamicValue [▶ 2347].)
	ValueFactory [▶ 2355]	The value factory (Inherited from DynamicValue [▶ 2347].)

Reference

[DynamicArrayValue Class](#) [[▶ 3042](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.148.2 DynamicArrayValue Methods

The [DynamicArrayValue](#) [[▶ 3042](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetDynamicMemberNames [▶ 2358]	Returns the enumeration of all dynamic member names. (Inherited from DynamicValue [▶ 2347].)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Read [▶ 2359]	Reads the value (via ADS) (Inherited from DynamicValue [▶ 2347].)
	ReadAsync [▶ 2359]	read as an asynchronous operation. (Inherited from DynamicValue [▶ 2347].)
	ResolveValue [▶ 2360]	Resolves the Value object to its primitive value. (Inherited from DynamicValue [▶ 2347].)
	setMemberValue [▶ 2374]	Writes the specified member element. (Inherited from DynamicValue [▶ 2347].)
	ToString [▶ 2361]	Returns a String that represents this instance. (Inherited from DynamicValue [▶ 2347].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert [▶ 2361]	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicValue [▶ 2347].)

	Name	Description
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetArrayElementValues [▶ 2362]	Returns Array Element values. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndex [▶ 2363]	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndexValue(Int32, Object) [▶ 2364]	Reads the specified array element. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndexValue(Object, Object) [▶ 2364]	Tries the get index value. (Inherited from DynamicValue [▶ 2347].)
	TryGetMember [▶ 2365]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Inherited from DynamicValue [▶ 2347].)
	TryGetMemberValue [▶ 2366]	Tries the get member value. (Inherited from DynamicValue [▶ 2347].)
	TryInvoke [▶ 2366]	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicValue [▶ 2347].)
	TryInvokeMember [▶ 2367]	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicValue [▶ 2347].)
	TryResolveValue [▶ 2368]	Tries to resolves the Value object to its primitive value. (Inherited from DynamicValue [▶ 2347].)
	TrySetIndex [▶ 2369]	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicValue [▶ 2347].)
	TrySetIndexValue(Int32, Object) [▶ 2370]	Tries to set the indexed value on Arrays (Inherited from DynamicValue [▶ 2347].)
	TrySetIndexValue(Object, Object) [▶ 2370]	Tries to set the indexed value on Arrays (Inherited from DynamicValue [▶ 2347].)

	Name	Description
	TrySetMember [▶ 2371]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicValue [▶ 2347].)
	TrySetMemberValue [▶ 2372]	Tries to Set a Member/Property Value (Inherited from DynamicValue [▶ 2347].)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject.)
	Write [▶ 2372]	Writes the value (via ADS) (Inherited from DynamicValue [▶ 2347].)
	WriteAsync [▶ 2373]	write as an asynchronous operation. (Inherited from DynamicValue [▶ 2347].)

Reference

[DynamicArrayValue Class](#) [▶ 3042]

[TwinCAT.TypeSystem Namespace](#) [▶ 2083]

6.11.149 DynamicEnumValue Class

Class DynamicEnumValue. Implements the [DynamicValue](#) [▶ 2347]

Inheritance Hierarchy

System.Object
 System.Dynamic.DynamicObject
 [TwinCAT.TypeSystem.DynamicValue](#) [▶ 2347]
 TwinCAT.TypeSystem.DynamicEnumValue

Namespace: [TwinCAT.TypeSystem](#) [▶ 2083]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public class DynamicEnumValue : DynamicValue
```

The DynamicEnumValue type exposes the following members.

Properties

	Name	Description
	Age [▶ 2351]	Gets the age of the value (last successful read of the value) (Inherited from DynamicValue [▶ 2347].)
	CachedRaw [▶ 2351]	Gets the cached Raw internal Data. (Inherited from DynamicValue [▶ 2347].)
	DataType [▶ 2352]	Gets the data type bound to this IValue [▶ 2745] (Inherited from DynamicValue [▶ 2347].)
	IsPrimitive [▶ 2353]	Gets a value indicating whether this IValue [▶ 2745] is a primitive value. (Inherited from DynamicValue [▶ 2347].)
	ParentValue [▶ 2356]	Gets the parent value. (Inherited from DynamicValue [▶ 2347].)

	Name	Description
	ResolvedType [▶ 2353]	Gets the resolved type. (Inherited from DynamicValue [▶ 2347].)
	Symbol [▶ 2353]	Gets the symbol that is bound to this value. (Inherited from DynamicValue [▶ 2347].)
	TimeStamp [▶ 2354]	Gets the Time stamp of the last successful read of the Value. (Inherited from DynamicValue [▶ 2347].)
	UpdateMode [▶ 2354]	Gets / Sets the update mode (Inherited from DynamicValue [▶ 2347].)
	ValueFactory [▶ 2355]	The value factory (Inherited from DynamicValue [▶ 2347].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetDynamicMemberNames [▶ 2358]	Returns the enumeration of all dynamic member names. (Inherited from DynamicValue [▶ 2347].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject .)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Read [▶ 2359]	Reads the value (via ADS) (Inherited from DynamicValue [▶ 2347].)
	ReadAsync [▶ 2359]	read as an asynchronous operation. (Inherited from DynamicValue [▶ 2347].)
	ResolveValue [▶ 2360]	Resolves the Value object to its primitive value. (Inherited from DynamicValue [▶ 2347].)
	setMemberValue [▶ 2374]	Writes the specified member element. (Inherited from DynamicValue [▶ 2347].)
	ToString [▶ 2361]	Returns a String that represents this instance. (Inherited from DynamicValue [▶ 2347].)
	TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject .)
	TryConvert [▶ 2361]	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicValue [▶ 2347].)
	TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject .)

	Name	Description
	TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
	TryGetArrayElement Values [▶ 2362]	Returns Array Element values. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndex [▶ 2363]	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndexValue(.Int32., Object.) [▶ 2364]	Reads the specified array element. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndexValue(Object., Object.) [▶ 2364]	Tries the get index value. (Inherited from DynamicValue [▶ 2347].)
	TryGetMember [▶ 2365]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Inherited from DynamicValue [▶ 2347].)
	TryGetMemberValue [▶ 2366]	Tries the get member value. (Inherited from DynamicValue [▶ 2347].)
	TryInvoke [▶ 2366]	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicValue [▶ 2347].)
	TryInvokeMember [▶ 2367]	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicValue [▶ 2347].)
	TryResolveValue [▶ 2368]	Tries to resolve the Value object to its primitive value. (Inherited from DynamicValue [▶ 2347].)
	TrySetIndex [▶ 2369]	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicValue [▶ 2347].)
	TrySetIndexValue(Int32., Object.) [▶ 2370]	Tries to set the indexed value on Arrays (Inherited from DynamicValue [▶ 2347].)
	TrySetIndexValue(Object., Object.) [▶ 2370]	Tries to set the indexed value on Arrays (Inherited from DynamicValue [▶ 2347].)
	TrySetMember [▶ 2371]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicValue [▶ 2347].)

	Name	Description
	TrySetMemberValue [▶ 2372]	Tries to Set a Member/Property Value (Inherited from DynamicValue [▶ 2347].)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject .)
	Write [▶ 2372]	Writes the value (via ADS) (Inherited from DynamicValue [▶ 2347].)
	WriteAsync [▶ 2373]	write as an asynchronous operation. (Inherited from DynamicValue [▶ 2347].)

Reference

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

[TwinCAT.TypeSystem.DynamicValue \[▶ 2347\]](#)

6.11.149.1 DynamicEnumValue Properties

The [DynamicEnumValue \[▶ 3048\]](#) type exposes the following members.

Properties

	Name	Description
	Age [▶ 2351]	Gets the age of the value (last successful read of the value) (Inherited from DynamicValue [▶ 2347].)
	CachedRaw [▶ 2351]	Gets the cached Raw internal Data. (Inherited from DynamicValue [▶ 2347].)
	DataType [▶ 2352]	Gets the data type bound to this IValue [▶ 2745] (Inherited from DynamicValue [▶ 2347].)
	IsPrimitive [▶ 2353]	Gets a value indicating whether this IValue [▶ 2745] is a primitive value. (Inherited from DynamicValue [▶ 2347].)
	ParentValue [▶ 2356]	Gets the parent value. (Inherited from DynamicValue [▶ 2347].)
	ResolvedType [▶ 2353]	Gets the resolved type. (Inherited from DynamicValue [▶ 2347].)
	Symbol [▶ 2353]	Gets the symbol that is bound to this value. (Inherited from DynamicValue [▶ 2347].)
	TimeStamp [▶ 2354]	Gets the Time stamp of the last successful read of the Value. (Inherited from DynamicValue [▶ 2347].)
	UpdateMode [▶ 2354]	Gets / Sets the update mode (Inherited from DynamicValue [▶ 2347].)
	ValueFactory [▶ 2355]	The value factory (Inherited from DynamicValue [▶ 2347].)

Reference

[DynamicEnumValue Class \[▶ 3048\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.149.2 DynamicEnumValue Methods

The [DynamicEnumValue \[▶ 3048\]](#) type exposes the following members.

Methods

Name	Description
Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
GetDynamicMemberNames [▶ 2358]	Returns the enumeration of all dynamic member names. (Inherited from DynamicValue [▶ 2347].)
GetHashCode	Serves as the default hash function. (Inherited from Object.)
GetMetaObject	Provides a DynamicMetaObject that dispatches to the dynamic virtual methods. The object can be encapsulated inside another DynamicMetaObject to provide custom behavior for individual actions. This method supports the Dynamic Language Runtime infrastructure for language implementers and it is not intended to be used directly from your code. (Inherited from DynamicObject.)
GetType	Gets the Type of the current instance. (Inherited from Object.)
MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
Read [▶ 2359]	Reads the value (via ADS) (Inherited from DynamicValue [▶ 2347].)
ReadAsync [▶ 2359]	read as an asynchronous operation. (Inherited from DynamicValue [▶ 2347].)
ResolveValue [▶ 2360]	Resolves the Value object to its primitive value. (Inherited from DynamicValue [▶ 2347].)
setMemberValue [▶ 2374]	Writes the specified member element. (Inherited from DynamicValue [▶ 2347].)
ToString [▶ 2361]	Returns a String that represents this instance. (Inherited from DynamicValue [▶ 2347].)
TryBinaryOperation	Provides implementation for binary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as addition and multiplication. (Inherited from DynamicObject.)
TryConvert [▶ 2361]	Provides implementation for type conversion operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that convert an object from one type to another. (Inherited from DynamicValue [▶ 2347].)
TryCreateInstance	Provides the implementation for operations that initialize a new instance of a dynamic object. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
TryDeleteIndex	Provides the implementation for operations that delete an object by index. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
TryDeleteMember	Provides the implementation for operations that delete an object member. This method is not intended for use in C# or Visual Basic. (Inherited from DynamicObject.)
TryGetArrayElementValues [▶ 2362]	Returns Array Element values. (Inherited from DynamicValue [▶ 2347].)
TryGetIndex [▶ 2363]	Provides the implementation for operations that get a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for indexing operations. (Inherited from DynamicValue [▶ 2347].)

	Name	Description
	TryGetIndexValue(Int32, Object) [▶ 2364]	Reads the specified array element. (Inherited from DynamicValue [▶ 2347].)
	TryGetIndexValue(Object, Object) [▶ 2364]	Tries the get index value. (Inherited from DynamicValue [▶ 2347].)
	TryGetMember [▶ 2365]	Provides the implementation for operations that get member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as getting a value for a property. (Inherited from DynamicValue [▶ 2347].)
	TryGetMemberValue [▶ 2366]	Tries the get member value. (Inherited from DynamicValue [▶ 2347].)
	TryInvoke [▶ 2366]	Provides the implementation for operations that invoke an object. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as invoking an object or a delegate. (Inherited from DynamicValue [▶ 2347].)
	TryInvokeMember [▶ 2367]	Provides the implementation for operations that invoke a member. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as calling a method. (Inherited from DynamicValue [▶ 2347].)
	TryResolveValue [▶ 2368]	Tries to resolve the Value object to its primitive value. (Inherited from DynamicValue [▶ 2347].)
	TrySetIndex [▶ 2369]	Provides the implementation for operations that set a value by index. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations that access objects by a specified index. (Inherited from DynamicValue [▶ 2347].)
	TrySetIndexValue(Int32, Object) [▶ 2370]	Tries to set the indexed value on Arrays (Inherited from DynamicValue [▶ 2347].)
	TrySetIndexValue(Object, Object) [▶ 2370]	Tries to set the indexed value on Arrays (Inherited from DynamicValue [▶ 2347].)
	TrySetMember [▶ 2371]	Provides the implementation for operations that set member values. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as setting a value for a property. (Inherited from DynamicValue [▶ 2347].)
	TrySetMemberValue [▶ 2372]	Tries to Set a Member/Property Value (Inherited from DynamicValue [▶ 2347].)
	TryUnaryOperation	Provides implementation for unary operations. Classes derived from the DynamicObject class can override this method to specify dynamic behavior for operations such as negation, increment, or decrement. (Inherited from DynamicObject .)
	Write [▶ 2372]	Writes the value (via ADS) (Inherited from DynamicValue [▶ 2347].)
	WriteAsync [▶ 2373]	write as an asynchronous operation. (Inherited from DynamicValue [▶ 2347].)

Reference

[DynamicEnumValue Class \[► 3048\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.150 UInt32Ptr Structure

Represents an 32-Bit Pointer (Process independant)

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public struct UInt32Ptr
```

The UInt32Ptr type exposes the following members.

Constructors

	Name	Description
	UInt32Ptr. [► 3056]	Initializes a new instance of the UInt32Ptr struct.
	UInt32Ptr(UInt32) [► 3056]	Initializes a new instance of the UInt32Ptr struct.
	UInt32Ptr(Void*) [► 3056]	Initializes a new instance of the UInt32Ptr struct.

Properties

	Name	Description
	Size [► 3057]	Gets the size (4-Byte)

Methods

	Name	Description
	Add [► 3058]	Adds an offset to the pointer.
	Equals [► 3059]	Determines whether the specified Object is equal to this instance. (Overrides ValueType.Equals(Object).)
	GetHashCode [► 3059]	Returns a hash code for this instance. (Overrides ValueType.GetHashCode..)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	Subtract [► 3060]	Subtracts an offset from the specified pointer.
	ToPointer [► 3060]	Converts this 32-Pointer object to an unsafe void* pointer
	ToString [► 3060]	Returns a String that represents this instance. (Overrides ValueType.ToString..)
	ToUInt32 [► 3061]	Converts the 32-Bit Pointer to uint
	ToUInt64 [► 3061]	Converts the 32-Bit Pointer to ulong.

Operators

	Name	Description
	Addition [► 3062]	Implements the + operator.
	Equality [► 3063]	Implements the == operator.
	.UInt32 to UInt32Ptr [► 3064]	Performs an explicit conversion from UInt32 to UInt32Ptr.
	.UInt32Ptr to UInt32 [► 3065]	Performs an explicit conversion from UInt32Ptr to UInt32.
	.UInt32Ptr to UInt64 [► 3065]	Performs an explicit conversion from UInt32Ptr to UInt64.
	.UInt32Ptr to UIntPtr [► 3066]	Performs an explicit conversion from UInt32Ptr to UIntPtr.
	.UInt32Ptr to Void* [► 3066]	Performs an explicit conversion from UInt32Ptr to Void.
	.UInt64 to UInt32Ptr [► 3067]	Performs an explicit conversion from UInt64 to UInt32Ptr.
	.UInt64Ptr to UInt32Ptr [► 3067]	Performs an explicit conversion from UInt64Ptr [► 3070] to UInt32Ptr.
	.Void* to UInt32Ptr [► 3068]	Performs an explicit conversion from Void to UInt32Ptr.
	Inequality [► 3068]	Implements the != operator.
	Subtraction [► 3069]	Implements the - operator.

Fields

	Name	Description
	Zero [► 3070]	Zero/Null Pointer

Reference

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

System.UIntPtr

6.11.150.1 UInt32Ptr Constructor

Overload List

	Name	Description
	UInt32Ptr. [► 3056]	Initializes a new instance of the UInt32Ptr [► 3054] struct.
	UInt32Ptr(UInt32) [► 3056]	Initializes a new instance of the UInt32Ptr [► 3054] struct.
	UInt32Ptr(Void*) [► 3056]	Initializes a new instance of the UInt32Ptr [► 3054] struct.

Reference[UInt32Ptr Structure \[► 3054\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.150.1.1 UInt32Ptr Constructor**Initializes a new instance of the [UInt32Ptr \[► 3054\]](#) struct.**Namespace:** [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public UInt32Ptr()
```

Reference[UInt32Ptr Structure \[► 3054\]](#)[UInt32Ptr Overload \[► 3055\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.150.1.2 UInt32Ptr Constructor (UInt32)**Initializes a new instance of the [UInt32Ptr \[► 3054\]](#) struct.**Namespace:** [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public UInt32Ptr(
    uint pVoid
)
```

Parameters

pVoid	Type: System.UInt32 The Pointer address in 32-Bit.
-------	---

Reference[UInt32Ptr Structure \[► 3054\]](#)[UInt32Ptr Overload \[► 3055\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.150.1.3 UInt32Ptr Constructor (Void.)**Initializes a new instance of the [UInt32Ptr \[► 3054\]](#) struct.**Namespace:** [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public UInt32Ptr(
    void* value
)
```

Parameters

value	Type: System.Void* The value.
-------	----------------------------------

Reference

[UInt32Ptr Structure \[▶ 3054\]](#)

[UInt32Ptr Overload \[▶ 3055\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.150.2 UInt32Ptr Properties

The [UInt32Ptr \[▶ 3054\]](#) type exposes the following members.

Properties

	Name	Description
	Size [▶ 3057]	Gets the size (4-Byte)

Reference

[UInt32Ptr Structure \[▶ 3054\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.150.2.1 UInt32Ptr.Size Property

Gets the size (4-Byte)

Namespace: [TwinCAT.TypeSystem \[▶ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static int Size { get; }
```

Property Value

Type: Int32
The size.

Reference

[UInt32Ptr Structure \[▶ 3054\]](#)

[TwinCAT.TypeSystem Namespace \[▶ 2083\]](#)

6.11.150.3 UInt32Ptr Methods

The [UInt32Ptr](#) [[▶ 3054](#)] type exposes the following members.

Methods

	Name	Description
	Add [▶ 3058]	Adds an offset to the pointer.
	Equals [▶ 3059]	Determines whether the specified Object is equal to this instance. (Overrides <code>ValueType.Equals(Object)</code> .)
	GetHashCode [▶ 3059]	Returns a hash code for this instance. (Overrides <code>ValueType.GetHashCode()</code> .)
	GetType	Gets the Type of the current instance. (Inherited from <code>Object</code> .)
	Subtract [▶ 3060]	Subtracts an offset from the specified pointer.
	ToPointer [▶ 3060]	Converts this 32-Pointer object to an unsafe <code>void*</code> pointer
	ToString [▶ 3060]	Returns a String that represents this instance. (Overrides <code>ValueType.ToString()</code> .)
	ToUInt32 [▶ 3061]	Converts the 32-Bit Pointer to <code>uint</code>
	ToUInt64 [▶ 3061]	Converts the 32-Bit Pointer to <code>ulong</code> .

Reference

[UInt32Ptr Structure](#) [[▶ 3054](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.150.3.1 UInt32Ptr.Add Method

Adds an offset to the pointer.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static UInt32Ptr Add(
    UInt32Ptr pointer,
    int offset
)
```

Parameters

pointer	Type: TwinCAT.TypeSystem.UInt32Ptr [▶ 3054] The pointer.
offset	Type: <code>System.Int32</code> The offset.

Return Value

Type: [UInt32Ptr](#) [[▶ 3054](#)]
`UInt32Ptr`.

Reference

[UInt32Ptr Structure \[► 3054\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.150.3.2 UInt32Ptr.Equals Method

Determines whether the specified Object is equal to this instance.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool Equals(  
    Object? obj  
)
```

Parameters

obj	Type: System.Object The object to compare with the current instance.
-----	---

Return Value

Type: Boolean

true if the specified Object is equal to this instance; otherwise, false.

Reference

[UInt32Ptr Structure \[► 3054\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.150.3.3 UInt32Ptr.GetHashCode Method

Returns a hash code for this instance.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override int GetHashCode()
```

Return Value

Type: Int32

A hash code for this instance, suitable for use in hashing algorithms and data structures like a hash table.

Reference

[UInt32Ptr Structure \[► 3054\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.150.3.4 UInt32Ptr.Subtract Method

Subtracts an offset from the specified pointer.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static UInt32Ptr Subtract(  
    UInt32Ptr pointer,  
    int offset  
)
```

Parameters

pointer	Type: TwinCAT.TypeSystem.UInt32Ptr [▶ 3054] The pointer.
offset	Type: System.Int32 The offset.

Return Value

Type: [UInt32Ptr](#) [[▶ 3054](#)]

UInt32Ptr.

Reference

[UInt32Ptr Structure](#) [[▶ 3054](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.150.3.5 UInt32Ptr.ToPointer Method

Converts this 32-Pointer object to an unsafe void* pointer

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void* ToPointer()
```

Return Value

Type: Void*

Reference

[UInt32Ptr Structure](#) [[▶ 3054](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.150.3.6 UInt32Ptr.ToString Method

Returns a String that represents this instance.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```

Return Value

Type: String

A String that represents this instance.

Reference

[UInt32Ptr Structure](#) [[▶ 3054](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.150.3.7 UInt32Ptr.ToUInt32 Method

Converts the 32-Bit Pointer to uint

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint ToUInt32()
```

Return Value

Type: UInt32

System.UInt32.

Reference

[UInt32Ptr Structure](#) [[▶ 3054](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.150.3.8 UInt32Ptr.ToUInt64 Method

Converts the 32-Bit Pointer to ulong.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ulong ToUInt64()
```

Return Value

Type: UInt64

System.UInt64.

Reference

[UInt32Ptr Structure](#) [[▶ 3054](#)]

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.150.4 UInt32Ptr Operators and Type Conversions

The [UInt32Ptr \[► 3054\]](#) type exposes the following members.

Operators

Name	Description
Addition [► 3062]	Implements the + operator.
Equality [► 3063]	Implements the == operator.
.(UInt32 to UInt32Ptr) [► 3064]	Performs an explicit conversion from UInt32 to UInt32Ptr [► 3054] .
.(UInt32Ptr to UInt32) [► 3065]	Performs an explicit conversion from UInt32Ptr [► 3054] to UInt32.
.(UInt32Ptr to UInt64) [► 3065]	Performs an explicit conversion from UInt32Ptr [► 3054] to UInt64.
.(UInt32Ptr to UIntPtr) [► 3066]	Performs an explicit conversion from UInt32Ptr [► 3054] to UIntPtr.
.(UInt32Ptr to Void*) [► 3066]	Performs an explicit conversion from UInt32Ptr [► 3054] to Void.
.(UInt64 to UInt32Ptr) [► 3067]	Performs an explicit conversion from UInt64 to UInt32Ptr [► 3054] .
.(UInt64Ptr to UInt32Ptr) [► 3067]	Performs an explicit conversion from UInt64Ptr [► 3070] to UInt32Ptr [► 3054] .
.(Void* to UInt32Ptr) [► 3068]	Performs an explicit conversion from Void to UInt32Ptr [► 3054] .
Inequality [► 3068]	Implements the != operator.
Subtraction [► 3069]	Implements the - operator.

Reference

[UInt32Ptr Structure \[► 3054\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.150.4.1 UInt32Ptr.Addition Operator

Implements the + operator.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static UInt32Ptr operator +(
    UInt32Ptr pointer,
    int offset
)
```

Parameters

pointer	Type: TwinCAT.TypeSystem.UInt32Ptr [▶ 3054] The pointer.
offset	Type: System.Int32 The offset.

Return Value

Type: [UInt32Ptr](#) [[▶ 3054](#)]
The result of the operator.

Reference

[UInt32Ptr Structure](#) [[▶ 3054](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.150.4.2 UInt32Ptr.Equality Operator

Implements the == operator.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax**C#**

```
public static bool operator ==(
    UInt32Ptr value1,
    UInt32Ptr value2
)
```

Parameters

value1	Type: TwinCAT.TypeSystem.UInt32Ptr [▶ 3054] The value1.
value2	Type: TwinCAT.TypeSystem.UInt32Ptr [▶ 3054] The value2.

Return Value

Type: [Boolean](#)
The result of the operator.

Reference

[UInt32Ptr Structure](#) [[▶ 3054](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.150.4.3 UInt32Ptr . Conversion Operators**Overload List**

	Name	Description
	.(UInt32 to UInt32Ptr) [▶ 3064]	Performs an explicit conversion from UInt32 to UInt32Ptr [▶ 3054].

	Name	Description
	.(UInt32Ptr to UInt32) [▸ 3065]	Performs an explicit conversion from UInt32Ptr [▸ 3054] to UInt32 .
	.(UInt32Ptr to UInt64) [▸ 3065]	Performs an explicit conversion from UInt32Ptr [▸ 3054] to UInt64 .
	.(UInt32Ptr to UIntPtr) [▸ 3066]	Performs an explicit conversion from UInt32Ptr [▸ 3054] to UIntPtr .
	.(UInt32Ptr to Void*) [▸ 3066]	Performs an explicit conversion from UInt32Ptr [▸ 3054] to Void .
	.(UInt64 to UInt32Ptr) [▸ 3067]	Performs an explicit conversion from UInt64 to UInt32Ptr [▸ 3054] .
	.(UInt64Ptr to UInt32Ptr) [▸ 3067]	Performs an explicit conversion from UInt64Ptr [▸ 3070] to UInt32Ptr [▸ 3054] .
	.(Void* to UInt32Ptr) [▸ 3068]	Performs an explicit conversion from Void to UInt32Ptr [▸ 3054] .

Reference

[UInt32Ptr Structure \[▸ 3054\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.150.4.3.1 UInt32Ptr . Conversion (UInt32 to UInt32Ptr)

Performs an explicit conversion from [UInt32](#) to [UInt32Ptr \[▸ 3054\]](#).

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: [TwinCAT.Ads.Abstractions \(in TwinCAT.Ads.Abstractions.dll\) Version: 6.0.328+39e3229](#)

Syntax

C#

```
public static explicit operator UInt32Ptr (
    uint value
)
```

Parameters

value	Type: System.UInt32 The value.
-------	---

Return Value

Type: [UInt32Ptr \[▸ 3054\]](#)

The result of the conversion.

Reference

[UInt32Ptr Structure \[▸ 3054\]](#)

[. Overload \[▸ 3063\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.150.4.3.2 UInt32Ptr . Conversion (UInt32Ptr to UInt32)

Performs an explicit conversion from [UInt32Ptr \[► 3054\]](#) to UInt32.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static explicit operator uint (  
    UInt32Ptr value  
)
```

Parameters

value	Type: TwinCAT.TypeSystem.UInt32Ptr [► 3054] The value.
-------	---

Return Value

Type: UInt32

The result of the conversion.

Reference

[UInt32Ptr Structure \[► 3054\]](#)

[.Overload \[► 3063\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.150.4.3.3 UInt32Ptr . Conversion (UInt32Ptr to UInt64)

Performs an explicit conversion from [UInt32Ptr \[► 3054\]](#) to UInt64.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static explicit operator ulong (  
    UInt32Ptr value  
)
```

Parameters

value	Type: TwinCAT.TypeSystem.UInt32Ptr [► 3054] The value.
-------	---

Return Value

Type: UInt64

The result of the conversion.

Reference

[UInt32Ptr Structure \[► 3054\]](#)

[.Overload \[► 3063\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.150.4.3.4 UInt32Ptr . Conversion (UInt32Ptr to UIntPtr)

Performs an explicit conversion from [UInt32Ptr \[► 3054\]](#) to [UIntPtr](#).

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static explicit operator UIntPtr (
    UInt32Ptr value
)
```

Parameters

value	Type: TwinCAT.TypeSystem.UInt32Ptr [► 3054] The value.
-------	---

Return Value

Type: [UIntPtr](#)
The result of the conversion.

Reference

[UInt32Ptr Structure \[► 3054\]](#)

[. Overload \[► 3063\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.150.4.3.5 UInt32Ptr . Conversion (UInt32Ptr to Void.)

Performs an explicit conversion from [UInt32Ptr \[► 3054\]](#) to [Void](#).

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static explicit operator void* (
    UInt32Ptr value
)
```

Parameters

value	Type: TwinCAT.TypeSystem.UInt32Ptr [► 3054] The value.
-------	---

Return Value

Type: [Void*](#)
The result of the conversion.

Reference

[UInt32Ptr Structure \[► 3054\]](#)

[.Overload \[► 3063\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.150.4.3.6 UInt32Ptr . Conversion (UInt64 to UInt32Ptr)

Performs an explicit conversion from [UInt64](#) to [UInt32Ptr](#) [► 3054].

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static explicit operator UInt32Ptr (  
    ulong value  
)
```

Parameters

value	Type: System.UInt64 The value.
-------	-----------------------------------

Return Value

Type: [UInt32Ptr](#) [► 3054]

The result of the conversion.

Reference

[UInt32Ptr Structure \[► 3054\]](#)

[.Overload \[► 3063\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.150.4.3.7 UInt32Ptr . Conversion (UInt64Ptr to UInt32Ptr)

Performs an explicit conversion from [UInt64Ptr](#) [► 3070] to [UInt32Ptr](#) [► 3054].

Namespace: [TwinCAT.TypeSystem](#) [► 2083]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static explicit operator UInt32Ptr (  
    UInt64Ptr value  
)
```

Parameters

value	Type: TwinCAT.TypeSystem.UInt64Ptr [► 3070] The value.
-------	---

Return ValueType: [UInt32Ptr](#) [[▶ 3054](#)]

The result of the conversion.

Reference[UInt32Ptr Structure](#) [[▶ 3054](#)][.Overload](#) [[▶ 3063](#)][TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]**6.11.150.4.3.8 UInt32Ptr . Conversion (Void. to UInt32Ptr)**Performs an explicit conversion from Void to [UInt32Ptr](#) [[▶ 3054](#)].**Namespace:** [TwinCAT.TypeSystem](#) [[▶ 2083](#)]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static explicit operator UInt32Ptr (
    void* value
)
```

Parameters

value	Type: System.Void* The value.
-------	----------------------------------

Return ValueType: [UInt32Ptr](#) [[▶ 3054](#)]

The result of the conversion.

Reference[UInt32Ptr Structure](#) [[▶ 3054](#)][.Overload](#) [[▶ 3063](#)][TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]**6.11.150.4.4 UInt32Ptr.Inequality Operator**

Implements the != operator.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static bool operator !=(
    UInt32Ptr value1,
    UInt32Ptr value2
)
```

Parameters

value1	Type: TwinCAT.TypeSystem.UInt32Ptr [▶ 3054] The value1.
value2	Type: TwinCAT.TypeSystem.UInt32Ptr [▶ 3054] The value2.

Return Value

Type: Boolean
The result of the operator.

Reference

[UInt32Ptr Structure](#) [[▶ 3054](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.150.4.5 UInt32Ptr.Subtraction Operator

Implements the - operator.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static UInt32Ptr operator -(
    UInt32Ptr pointer,
    int offset
)
```

Parameters

pointer	Type: TwinCAT.TypeSystem.UInt32Ptr [▶ 3054] The pointer.
offset	Type: System.Int32 The offset.

Return Value

Type: [UInt32Ptr](#) [[▶ 3054](#)]
The result of the operator.

Reference

[UInt32Ptr Structure](#) [[▶ 3054](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.150.5 UInt32Ptr Fields

The [UInt32Ptr](#) [[▶ 3054](#)] type exposes the following members.

Fields

	Name	Description
	Zero [▶ 3070]	Zero/Null Pointer

	Name	Description

Reference

[UInt32Ptr Structure \[▸ 3054\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.150.5.1 UInt32Ptr.Zero Field

Zero/Null Pointer

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static readonly UIntPtr Zero
```

Field Value

Type: UIntPtr

Reference

[UInt32Ptr Structure \[▸ 3054\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.151 UInt64Ptr Structure

Represents an 64-Bit Pointer (Process independant)

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public struct UInt64Ptr
```

The UInt64Ptr type exposes the following members.

Constructors

	Name	Description
	UInt64Ptr(UInt64) [▸ 3072]	Initializes a new instance of the UInt64Ptr struct.
	UInt64Ptr(Void*) [▸ 3072]	Initializes a new instance of the UInt64Ptr struct.

Properties

	Name	Description
	Size [▸ 3073]	Gets the size (8-Byte)

Methods

Name	Description
Add [▶ 3074]	Adds an offset to the pointer.
Equals [▶ 3075]	Determines whether the specified Object is equal to this instance. (Overrides ValueType.Equals(Object).)
GetHashCode [▶ 3075]	Returns a hash code for this instance. (Overrides ValueType.GetHashCode..)
GetType	Gets the Type of the current instance. (Inherited from Object.)
Subtract [▶ 3076]	Subtracts an offset from the pointer.
ToPointer [▶ 3076]	Converts this 64-Pointer object to an unsafe void* pointer
ToString [▶ 3076]	Returns a String that represents this instance. (Overrides ValueType.ToString..)
ToUInt32 [▶ 3077]	Converts this 64-Bit Pointer to an UInt32 value
ToUInt64 [▶ 3077]	Converts this 64-Bit Pointer to an UInt64 value

Operators

Name	Description
Addition [▶ 3078]	Implements the + operator.
Equality [▶ 3079]	Implements the == operator.
.UInt32 to UInt64Ptr [▶ 3080]	Performs an explicit conversion from UInt32 to UInt64Ptr.
.UInt32Ptr to UInt64Ptr [▶ 3081]	Performs an explicit conversion from UInt32Ptr [▶ 3054] to UInt64Ptr.
.UInt64 to UInt64Ptr [▶ 3081]	Performs an explicit conversion from UInt64 to UInt64Ptr.
.UInt64Ptr to UInt32 [▶ 3082]	Performs an explicit conversion from UInt64Ptr to UInt32.
.UInt64Ptr to UInt64 [▶ 3082]	Performs an explicit conversion from UInt64Ptr to UInt64.
.UInt64Ptr to UIntPtr [▶ 3083]	Performs an explicit conversion from UInt64Ptr to UIntPtr.
.UInt64Ptr to Void* [▶ 3083]	Performs an explicit conversion from UInt64Ptr to Void.
.Void* to UInt64Ptr [▶ 3084]	Performs an explicit conversion from Void to UInt64Ptr.
Inequality [▶ 3084]	Implements the != operator.
Subtraction [▶ 3085]	Subtracts an offset from the pointer.

Fields

Name	Description
Zero [▶ 3086]	The zero/null pointer

	Name	Description

Reference

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

System.UIntPtr

6.11.151.1 UInt64Ptr Constructor

Overload List

	Name	Description
	UInt64Ptr(UInt64) [► 3072]	Initializes a new instance of the UInt64Ptr [► 3070] struct.
	UInt64Ptr(Void*) [► 3072]	Initializes a new instance of the UInt64Ptr [► 3070] struct.

Reference

[UInt64Ptr Structure \[► 3070\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.151.1.1 UInt64Ptr Constructor (UInt64)

Initializes a new instance of the [UInt64Ptr \[► 3070\]](#) struct.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public UInt64Ptr(
    ulong value
)
```

Parameters

value	Type: System.UInt64 The value.
-------	-----------------------------------

Reference

[UInt64Ptr Structure \[► 3070\]](#)

[UInt64Ptr Overload \[► 3072\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.151.1.2 UInt64Ptr Constructor (Void.)

Initializes a new instance of the [UInt64Ptr \[► 3070\]](#) struct.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public UInt64Ptr(  
    void* value  
)
```

Parameters

value	Type: System.Void* The value.
-------	----------------------------------

Reference

[UInt64Ptr Structure \[► 3070\]](#)

[UInt64Ptr Overload \[► 3072\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.151.2 UInt64Ptr Properties

The [UInt64Ptr \[► 3070\]](#) type exposes the following members.

Properties

	Name	Description
	Size [► 3073]	Gets the size (8-Byte)

Reference

[UInt64Ptr Structure \[► 3070\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.151.2.1 UInt64Ptr.Size Property

Gets the size (8-Byte)

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static int Size { get; }
```

Property Value

Type: Int32
The size.

Reference

[UInt64Ptr Structure \[► 3070\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.151.3 UInt64Ptr Methods

The [UInt64Ptr](#) [[▶ 3070](#)] type exposes the following members.

Methods

Name	Description
Add [▶ 3074]	Adds an offset to the pointer.
Equals [▶ 3075]	Determines whether the specified Object is equal to this instance. (Overrides <code>ValueType.Equals(Object)</code> .)
GetHashCode [▶ 3075]	Returns a hash code for this instance. (Overrides <code>ValueType.GetHashCode()</code> .)
GetType	Gets the Type of the current instance. (Inherited from <code>Object</code> .)
Subtract [▶ 3076]	Subtracts an offset from the pointer.
ToPointer [▶ 3076]	Converts this 64-Pointer object to an unsafe <code>void*</code> pointer
ToString [▶ 3076]	Returns a String that represents this instance. (Overrides <code>ValueType.ToString()</code> .)
ToUInt32 [▶ 3077]	Converts this 64-Bit Pointer to an <code>UInt32</code> value
ToUInt64 [▶ 3077]	Converts this 64-Bit Pointer to an <code>UInt64</code> value

Reference

[UInt64Ptr Structure](#) [[▶ 3070](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.151.3.1 UInt64Ptr.Add Method

Adds an offset to the pointer.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static UInt64Ptr Add(
    UInt64Ptr pointer,
    int offset
)
```

Parameters

pointer	Type: TwinCAT.TypeSystem.UInt64Ptr [▶ 3070] The pointer.
offset	Type: <code>System.Int32</code> The offset.

Return Value

Type: [UInt64Ptr](#) [[▶ 3070](#)]
`UInt64Ptr`.

Reference

[UInt64Ptr Structure \[► 3070\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.151.3.2 UInt64Ptr.Equals Method

Determines whether the specified Object is equal to this instance.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override bool Equals(  
    Object? obj  
)
```

Parameters

obj	Type: System.Object The object to compare with the current instance.
-----	---

Return Value

Type: Boolean

true if the specified Object is equal to this instance; otherwise, false.

Reference

[UInt64Ptr Structure \[► 3070\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.151.3.3 UInt64Ptr.GetHashCode Method

Returns a hash code for this instance.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override int GetHashCode()
```

Return Value

Type: Int32

A hash code for this instance, suitable for use in hashing algorithms and data structures like a hash table.

Reference

[UInt64Ptr Structure \[► 3070\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.151.3.4 UInt64Ptr.Subtract Method

Subtracts an offset from the pointer.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static UInt64Ptr Subtract(  
    UInt64Ptr pointer,  
    int offset  
)
```

Parameters

pointer	Type: TwinCAT.TypeSystem.UInt64Ptr [▶ 3070] The pointer.
offset	Type: System.Int32 The offset.

Return Value

Type: [UInt64Ptr](#) [[▶ 3070](#)]

UInt64Ptr.

Reference

[UInt64Ptr Structure](#) [[▶ 3070](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.151.3.5 UInt64Ptr.ToPointer Method

Converts this 64-Pointer object to an unsafe void* pointer

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void* ToPointer()
```

Return Value

Type: Void*

Reference

[UInt64Ptr Structure](#) [[▶ 3070](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.151.3.6 UInt64Ptr.ToString Method

Returns a String that represents this instance.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```

Return Value

Type: String

A String that represents this instance.

Reference

[UInt64Ptr Structure \[► 3070\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.151.3.7 UInt64Ptr.ToUInt32 Method

Converts this 64-Bit Pointer to an UInt32 value

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint ToUInt32()
```

Return Value

Type: UInt32

System.UInt32.

Reference

[UInt64Ptr Structure \[► 3070\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.151.3.8 UInt64Ptr.ToUInt64 Method

Converts this 64-Bit Pointer to an UInt64 value

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ulong ToUInt64()
```

Return Value

Type: UInt64

System.UInt64.

Reference

[UInt64Ptr Structure \[► 3070\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.151.4 UInt64Ptr Operators and Type Conversions

The [UInt64Ptr \[► 3070\]](#) type exposes the following members.

Operators

Name	Description
Addition [► 3078]	Implements the + operator.
Equality [► 3079]	Implements the == operator.
.(UInt32 to UInt64Ptr) [► 3080]	Performs an explicit conversion from UInt32 to UInt64Ptr [► 3070] .
.(UInt32Ptr to UInt64Ptr) [► 3081]	Performs an explicit conversion from UInt32Ptr [► 3054] to UInt64Ptr [► 3070] .
.(UInt64 to UInt64Ptr) [► 3081]	Performs an explicit conversion from UInt64 to UInt64Ptr [► 3070] .
.(UInt64Ptr to UInt32) [► 3082]	Performs an explicit conversion from UInt64Ptr [► 3070] to UInt32.
.(UInt64Ptr to UInt64) [► 3082]	Performs an explicit conversion from UInt64Ptr [► 3070] to UInt64.
.(UInt64Ptr to UIntPtr) [► 3083]	Performs an explicit conversion from UInt64Ptr [► 3070] to UIntPtr.
.(UInt64Ptr to Void*) [► 3083]	Performs an explicit conversion from UInt64Ptr [► 3070] to Void.
.(Void* to UInt64Ptr) [► 3084]	Performs an explicit conversion from Void to UInt64Ptr [► 3070] .
Inequality [► 3084]	Implements the != operator.
Subtraction [► 3085]	Subtracts an offset from the pointer.

Reference

[UInt64Ptr Structure \[► 3070\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.151.4.1 UInt64Ptr.Addition Operator

Implements the + operator.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static UInt64Ptr operator +(
    UInt64Ptr pointer,
    int offset
)
```

Parameters

pointer	Type: TwinCAT.TypeSystem.UInt64Ptr [▶ 3070] The pointer.
offset	Type: System.Int32 The offset.

Return Value

Type: [UInt64Ptr](#) [[▶ 3070](#)]
The result of the operator.

Reference

[UInt64Ptr Structure](#) [[▶ 3070](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.151.4.2 UInt64Ptr.Equality Operator

Implements the == operator.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool operator ==(
    UInt64Ptr value1,
    UInt64Ptr value2
)
```

Parameters

value1	Type: TwinCAT.TypeSystem.UInt64Ptr [▶ 3070] The value1.
value2	Type: TwinCAT.TypeSystem.UInt64Ptr [▶ 3070] The value2.

Return Value

Type: Boolean
The result of the operator.

Reference

[UInt64Ptr Structure](#) [[▶ 3070](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.151.4.3 UInt64Ptr . Conversion Operators

Overload List

	Name	Description
	.(UInt32 to UInt64Ptr) [▶ 3080]	Performs an explicit conversion from UInt32 to UInt64Ptr [▶ 3070].

	Name	Description
	.(UInt32Ptr to UInt64Ptr) [► 3081]	Performs an explicit conversion from UInt32Ptr [► 3054] to UInt64Ptr [► 3070] .
	.(UInt64 to UInt64Ptr) [► 3081]	Performs an explicit conversion from UInt64 to UInt64Ptr [► 3070] .
	.(UInt64Ptr to UInt32) [► 3082]	Performs an explicit conversion from UInt64Ptr [► 3070] to UInt32 .
	.(UInt64Ptr to UInt64) [► 3082]	Performs an explicit conversion from UInt64Ptr [► 3070] to UInt64 .
	.(UInt64Ptr to UIntPtr) [► 3083]	Performs an explicit conversion from UInt64Ptr [► 3070] to UIntPtr .
	.(UInt64Ptr to Void*) [► 3083]	Performs an explicit conversion from UInt64Ptr [► 3070] to Void .
	.(Void* to UInt64Ptr) [► 3084]	Performs an explicit conversion from Void to UInt64Ptr [► 3070] .

Reference

[UInt64Ptr Structure \[► 3070\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.151.4.3.1 UInt64Ptr . Conversion (UInt32 to UInt64Ptr)

Performs an explicit conversion from [UInt32](#) to [UInt64Ptr \[► 3070\]](#).

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: [TwinCAT.Ads.Abstractions \(in TwinCAT.Ads.Abstractions.dll\) Version: 6.0.328+39e3229](#)

Syntax

C#

```
public static explicit operator UInt64Ptr (
    uint value
)
```

Parameters

value	Type: System.UInt32 The value.
-------	---

Return Value

Type: [UInt64Ptr \[► 3070\]](#)

The result of the conversion.

Reference

[UInt64Ptr Structure \[► 3070\]](#)

[. Overload \[► 3079\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.151.4.3.2 UInt64Ptr . Conversion (UInt32Ptr to UInt64Ptr)

Performs an explicit conversion from [UInt32Ptr \[▸ 3054\]](#) to [UInt64Ptr \[▸ 3070\]](#).

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static explicit operator UInt64Ptr (  
    UInt32Ptr value  
)
```

Parameters

value	Type: TwinCAT.TypeSystem.UInt32Ptr [▸ 3054] The value.
-------	---

Return Value

Type: [UInt64Ptr \[▸ 3070\]](#)

The result of the conversion.

Reference

[UInt64Ptr Structure \[▸ 3070\]](#)

[.Overload \[▸ 3079\]](#)

[TwinCAT.TypeSystem Namespace \[▸ 2083\]](#)

6.11.151.4.3.3 UInt64Ptr . Conversion (UInt64 to UInt64Ptr)

Performs an explicit conversion from [UInt64](#) to [UInt64Ptr \[▸ 3070\]](#).

Namespace: [TwinCAT.TypeSystem \[▸ 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static explicit operator UInt64Ptr (  
    ulong value  
)
```

Parameters

value	Type: System.UInt64 The value.
-------	---

Return Value

Type: [UInt64Ptr \[▸ 3070\]](#)

The result of the conversion.

Reference

[UInt64Ptr Structure \[▸ 3070\]](#)

[.Overload \[▸ 3079\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.151.4.3.4 UInt64Ptr . Conversion (UInt64Ptr to UInt32)

Performs an explicit conversion from [UInt64Ptr \[► 3070\]](#) to UInt32.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static explicit operator uint (
    UInt64Ptr value
)
```

Parameters

value	Type: TwinCAT.TypeSystem.UInt64Ptr [► 3070] The value.
-------	---

Return Value

Type: UInt32
The result of the conversion.

Reference

[UInt64Ptr Structure \[► 3070\]](#)

[. Overload \[► 3079\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.151.4.3.5 UInt64Ptr . Conversion (UInt64Ptr to UInt64)

Performs an explicit conversion from [UInt64Ptr \[► 3070\]](#) to UInt64.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static explicit operator ulong (
    UInt64Ptr value
)
```

Parameters

value	Type: TwinCAT.TypeSystem.UInt64Ptr [► 3070] The value.
-------	---

Return Value

Type: UInt64
The result of the conversion.

Reference[UInt64Ptr Structure \[► 3070\]](#)[.Overload \[► 3079\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.151.4.3.6 UInt64Ptr . Conversion (UInt64Ptr to UIntPtr)**Performs an explicit conversion from [UInt64Ptr \[► 3070\]](#) to [UIntPtr](#).**Namespace:** [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static explicit operator UIntPtr (
    UInt64Ptr value
)
```

Parameters

value	Type: TwinCAT.TypeSystem.UInt64Ptr [► 3070] The value.
-------	---

Return Value

Type: [UIntPtr](#)
The result of the conversion.

Reference[UInt64Ptr Structure \[► 3070\]](#)[.Overload \[► 3079\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.151.4.3.7 UInt64Ptr . Conversion (UInt64Ptr to Void.)**Performs an explicit conversion from [UInt64Ptr \[► 3070\]](#) to [Void](#).**Namespace:** [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static explicit operator void* (
    UInt64Ptr value
)
```

Parameters

value	Type: TwinCAT.TypeSystem.UInt64Ptr [► 3070] The value.
-------	---

Return Value

Type: Void*

The result of the conversion.

Reference[UInt64Ptr Structure \[► 3070\]](#)[.Overload \[► 3079\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.151.4.3.8 UInt64Ptr . Conversion (Void. to UInt64Ptr)**Performs an explicit conversion from Void to [UInt64Ptr \[► 3070\]](#).**Namespace:** [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static explicit operator UInt64Ptr (
    void* pVoid
)
```

Parameters

pVoid	Type: System.Void* The void pointer value..
-------	--

Return ValueType: [UInt64Ptr \[► 3070\]](#)

The result of the conversion.

Reference[UInt64Ptr Structure \[► 3070\]](#)[.Overload \[► 3079\]](#)[TwinCAT.TypeSystem Namespace \[► 2083\]](#)**6.11.151.4.4 UInt64Ptr.Inequality Operator**

Implements the != operator.

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static bool operator !=(
    UInt64Ptr value1,
    UInt64Ptr value2
)
```

Parameters

value1	Type: TwinCAT.TypeSystem.UInt64Ptr [▶ 3070] The value1.
value2	Type: TwinCAT.TypeSystem.UInt64Ptr [▶ 3070] The value2.

Return Value

Type: Boolean
The result of the operator.

Reference

[UInt64Ptr Structure](#) [[▶ 3070](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.151.4.5 UInt64Ptr.Subtraction Operator

Subtracts an offset from the pointer.

Namespace: [TwinCAT.TypeSystem](#) [[▶ 2083](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static UInt64Ptr operator -(
    UInt64Ptr pointer,
    int offset
)
```

Parameters

pointer	Type: TwinCAT.TypeSystem.UInt64Ptr [▶ 3070] The pointer.
offset	Type: System.Int32 The offset.

Return Value

Type: [UInt64Ptr](#) [[▶ 3070](#)]
The result of the operator.

Reference

[UInt64Ptr Structure](#) [[▶ 3070](#)]

[TwinCAT.TypeSystem Namespace](#) [[▶ 2083](#)]

6.11.151.5 UInt64Ptr Fields

The [UInt64Ptr](#) [[▶ 3070](#)] type exposes the following members.

Fields

	Name	Description
	Zero [▶ 3086]	The zero/null pointer

	Name	Description

Reference

[UInt64Ptr Structure \[► 3070\]](#)

[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.11.151.5.1 UInt64Ptr.Zero Field

The zero/null pointer

Namespace: [TwinCAT.TypeSystem \[► 2083\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static readonly UIntPtr Zero
```

Field Value

Type: UIntPtr

Reference






[UInt64Ptr Structure \[► 3070\]](#)






[TwinCAT.TypeSystem Namespace \[► 2083\]](#)

6.12 TwinCAT.TypeSystem.Generic Namespace



Namespace for the dynamic part of the common type system.

Classes

	Class	Description
	DataTypeCollection.T. [► 3087]	Data type collection
	InstanceCollection.T. [► 3104]	Base class for IInstance [► 2549] object collections (abstract).
	NamespaceCollection.T. [► 3126]	Generic class for Namespace collections
	ReadOnlyDataTypeCollection.T. [► 3139]	ReadOnly DataType collection
	ReadOnlyInstanceCollection.T. [► 3145]	ReadOnly Instance collection

	Class	Description
	ReadOnlyNamespaceCollection.T. [▶ 3154]	Read Only namespace collection
	ReadOnlySymbolCollection.T. [▶ 3161]	Read only symbol collection.
	SymbolCollection.T. [▶ 3165]	Interface represents a collection of ISymbol [▶ 2691] objects.
 	SymbolIterator.T. [▶ 3172]	Iterator class for enumerations of Symbols [▶ 2691].

Interfaces

	Interface	Description
	INamespace.TType. [▶ 3102]	Namespace interface
	ISymbolProvider.TNamespace, TDataType, TSymbol. [▶ 3122]	Symbol provider interface

6.12.1 DataTypeCollection.T. Class

Data type collection

Inheritance Hierarchy

System.Object
 TwinCAT.TypeSystem.Generic.DataTypeCollection.T.
 TwinCAT.TypeSystem.DataTypeCollection [▶ 2112]

Namespace: TwinCAT.TypeSystem.Generic [▶ 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#


```
public class DataTypeCollection<T> : IList<T>,
    ICollection<T>, IEnumerable<T>, IEnumerable, IDatatypeCollection<T>
where T : class, IDatatype
```


Type Parameters

T





The DataTypeCollection.T. type exposes the following members.

Constructors




















	Name	Description
	DataTypeCollection.T.. [▶ 3089]	Initializes a new instance of the DataTypeCollection [▶ 2112] class.



	Name	Description
	DataTypeCollection.T.(IEnumerable.T.) [▶ 3090]	Initializes a new instance of the DataTypeCollection.T. class.

Properties


	Name	Description
	Count [▶ 3090]	Gets the count of contained IDataType [▶ 2475]s.
	IsReadOnly [▶ 3091]	Gets a value indicating whether this instance is read only.
	Item.Int32. [▶ 3092]	Gets or sets the IDataType [▶ 2475] at the specified index.
	Item.String. [▶ 3092]	Gets the IDataType [▶ 2475] with the specified name.

Methods

	Name	Description
	Add [▶ 3094]	Adds the specified item to the collection.
	AddRange [▶ 3094]	Adds a range of types
	AsReadOnly [▶ 3095]	Converts the DataTypeCollection.T. into a ReadOnlyCollection.T.
	Clear [▶ 3095]	Clears the collection.
	Clone [▶ 3095]	Clones this instance.
	Contains [▶ 3096]	Determines whether this DataTypeCollection [▶ 2112] contains the specified IDataType [▶ 2475].
	ContainsType [▶ 3096]	Determines whether the container contains the specified IDataType [▶ 2475].
	CopyTo [▶ 3097]	Copies the data types to the specified array, starting at the array index.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 3098]	Gets the enumerator.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 3098]	Determines the Index of the specified IDataType [▶ 2475].
	Insert [▶ 3099]	Inserts an IDataType [▶ 2475] into the DataTypeCollection [▶ 2112].
	LookupType [▶ 3099]	Determines the specified IDataType [▶ 2475]
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 3100]	Removes the specified IDataType [▶ 2475].
	RemoveAt [▶ 3100]	Removes the IDataType [▶ 2475] object at the specified index.

	Name	Description
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetType [▶ 3101]	Tries to get the specified IDataType [▶ 2475] from the IDataTypeCollection.T. [▶ 2486] .

Fields



	Name	Description
	list [▶ 3102]	Internal list of data types

Reference

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.1.1 DataTypeCollection.T. Constructor

Overload List

	Name	Description
	DataTypeCollection.T.. [▶ 3089]	Initializes a new instance of the DataTypeCollection [▶ 2112] class.
	DataTypeCollection.T.(IEnumerable.T.) [▶ 3090]	Initializes a new instance of the DataTypeCollection.T. [▶ 3087] class.

Reference

[DataTypeCollection.T. Class \[▶ 3087\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.1.1.1 DataTypeCollection.T. Constructor

Initializes a new instance of the [DataTypeCollection \[▶ 2112\]](#) class.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DataTypeCollection()
```

Reference

[DataTypeCollection.T. Class \[▶ 3087\]](#)

[DataTypeCollection.T. Overload \[▶ 3089\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.1.1.2 DataTypeCollection.T. Constructor (IEnumerable.T.)

Initializes a new instance of the [DataTypeCollection.T.](#) [[▶ 3087](#)] class.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DataTypeCollection(
    IEnumerable<T> types
)
```

Parameters

types	Type: System.Collections.Generic.IEnumerable.T [▶ 3087]. The types.
-------	--

Reference

[DataTypeCollection.T. Class](#) [[▶ 3087](#)]





[DataTypeCollection.T. Overload](#) [[▶ 3089](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.1.2 DataTypeCollection.T. Properties

The [DataTypeCollection.T.](#) [[▶ 3087](#)] generic type exposes the following members.

Properties

	Name	Description
	Count [▶ 3090]	Gets the count of contained IDataType [▶ 2475]s.
	IsReadOnly [▶ 3091]	Gets a value indicating whether this instance is read only.
	Item.Int32. [▶ 3092]	Gets or sets the IDataType [▶ 2475] at the specified index.
	Item.String. [▶ 3092]	Gets the IDataType [▶ 2475] with the specified name.

Reference

[DataTypeCollection.T. Class](#) [[▶ 3087](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.1.2.1 DataTypeCollection.T..Count Property

Gets the count of contained [IDataType](#) [[▶ 2475](#)]s.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Count { get; }
```

Property Value

Type: Int32
The count.

Implements

ICollection.T..Count

Reference

[DataTypeCollection.T. Class \[▸ 3087\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▸ 3086\]](#)

6.12.1.2.2 DataTypeCollection.T..IsReadOnly Property

Gets a value indicating whether this instance is read only.

Namespace: [TwinCAT.TypeSystem.Generic \[▸ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsReadOnly { get; }
```

Property Value

Type: Boolean
true if this instance is read only; otherwise, false.

Implements

ICollection.T..IsReadOnly



Reference

[DataTypeCollection.T. Class \[▸ 3087\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▸ 3086\]](#)

6.12.1.2.3 DataTypeCollection.T..Item Property

Overload List

	Name	Description
	Item.Int32. [▸ 3092]	Gets or sets the IDataType [▸ 2475] at the specified index.
	Item.String. [▸ 3092]	Gets the IDataType [▸ 2475] with the specified name.

Reference

[DataTypeCollection.T. Class \[▸ 3087\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▸ 3086\]](#)

6.12.1.2.3.1 DataTypeCollection.T..Item Property (Int32)

Gets or sets the [IDataType](#) [[▶ 2475](#)] at the specified index.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T this[
    int index
] { get; set; }
```

Parameters

index	Type: System.Int32 The index.
-------	----------------------------------

Return Value

Type: [T](#) [[▶ 3087](#)]

T.

Implements

[IList.T..Item.Int32](#).

Exceptions

Exception	Condition
NotImplementedException	

Reference

[DataTypeCollection.T. Class](#) [[▶ 3087](#)]

[Item Overload](#) [[▶ 3091](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.1.2.3.2 DataTypeCollection.T..Item Property (String)

Gets the [IDataType](#) [[▶ 2475](#)] with the specified name.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T this[
    string name
] { get; }
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: [T](#) [[▶ 3087](#)]
 T.

Implements

[IDataTypeCollection.T..Item.String.](#) [[▶ 2488](#)]

Reference

[DataTypeCollection.T. Class](#) [[▶ 3087](#)]



















[Item Overload](#) [[▶ 3091](#)]




[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.1.3 DataTypeCollection.T. Methods

The [DataTypeCollection.T.](#) [[▶ 3087](#)] generic type exposes the following members.

Methods

	Name	Description
	Add [▶ 3094]	Adds the specified item to the collection.
	AddRange [▶ 3094]	Adds a range of types
	AsReadOnly [▶ 3095]	Converts the DataTypeCollection.T. [▶ 3087] into a ReadOnlyCollection.T.
	Clear [▶ 3095]	Clears the collection.
	Clone [▶ 3095]	Clones this instance.
	Contains [▶ 3096]	Determines whether this DataTypeCollection [▶ 2112] contains the specified IDataType [▶ 2475].
	ContainsType [▶ 3096]	Determines whether the container contains the specified IDataType [▶ 2475].
	CopyTo [▶ 3097]	Copies the data types to the specified array, starting at the array index.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 3098]	Gets the enumerator.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 3098]	Determines the Index of the specified IDataType [▶ 2475].
	Insert [▶ 3099]	Inserts an IDataType [▶ 2475] into the DataTypeCollection [▶ 2112].
	LookupType [▶ 3099]	Determines the specified IDataType [▶ 2475]
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 3100]	Removes the specified IDataType [▶ 2475].

	Name	Description
	RemoveAt [▶ 3100]	Removes the IDataType [▶ 2475] object at the specified index.
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetType [▶ 3101]	Tries to get the specified IDataType [▶ 2475] from the IDataTypeCollection.T. [▶ 2486] .

Reference

[DataTypeCollection.T. Class \[▶ 3087\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.1.3.1 DataTypeCollection.T..Add Method

Adds the specified item to the collection.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Add(
    T item
)
```

Parameters

item	Type: T [▶ 3087] The item.
------	---

Implements

[ICollection.T..Add\(T\)](#)

Reference

[DataTypeCollection.T. Class \[▶ 3087\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.1.3.2 DataTypeCollection.T..AddRange Method

Adds a range of types

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void AddRange(
    IEnumerable<T> types
)
```

Parameters

types	Type: System.Collections.Generic.IEnumerable.T [▶ 3087]. The types.
-------	--

Reference

[DataTypeCollection.T. Class \[▶ 3087\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.1.3.3 **DataTypeCollection.T..AsReadOnly Method**

Converts the [DataTypeCollection.T. \[▶ 3087\]](#) into a [ReadOnlyCollection.T.](#)

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyDataTypeCollection<T> AsReadOnly()
```

Return Value

Type: [ReadOnlyDataTypeCollection \[▶ 3139\].T \[▶ 3087\]](#).

[ReadOnlyDataTypeCollection<T>](#).

Reference

[DataTypeCollection.T. Class \[▶ 3087\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.1.3.4 **DataTypeCollection.T..Clear Method**

Clears the collection.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Clear()
```

Implements

[ICollection.T..Clear.](#)

Reference

[DataTypeCollection.T. Class \[▶ 3087\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.1.3.5 **DataTypeCollection.T..Clone Method**

Clones this instance.

Namespace: [TwinCAT.TypeSystem.Generic](#) [▶ 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DataTypeCollection<T> Clone()
```

Return Value

Type: [DataTypeCollection](#) [▶ 3087]. [T](#) [▶ 3087].

DataTypeCollection<T>.

Reference

[DataTypeCollection.T. Class](#) [▶ 3087]

[TwinCAT.TypeSystem.Generic Namespace](#) [▶ 3086]

6.12.1.3.6 DataTypeCollection.T..Contains Method

Determines whether this [DataTypeCollection](#) [▶ 2112] contains the specified [IDataType](#) [▶ 2475].

Namespace: [TwinCAT.TypeSystem.Generic](#) [▶ 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(
    T item
)
```

Parameters

item	Type: T [▶ 3087] The item.
------	---

Return Value

Type: Boolean

true if [contains] [the specified item]; otherwise, false.

Implements

[ICollection.T..Contains\(T\)](#)

Reference

[DataTypeCollection.T. Class](#) [▶ 3087]

[TwinCAT.TypeSystem.Generic Namespace](#) [▶ 3086]

6.12.1.3.7 DataTypeCollection.T..ContainsType Method

Determines whether the container contains the specified [IDataType](#) [▶ 2475].

Namespace: [TwinCAT.TypeSystem.Generic](#) [▶ 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool ContainsType(  
    string name  
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: Boolean
true if contained; otherwise, false.

Implements

[IDataTypeCollection.T..ContainsType\(String\)](#) [► 2489]

Reference

[DataTypeCollection.T. Class](#) [► 3087]

[TwinCAT.TypeSystem.Generic Namespace](#) [► 3086]

6.12.1.3.8 DataTypeCollection.T..CopyTo Method

Copies the data types to the specified array, starting at the array index.

Namespace: [TwinCAT.TypeSystem.Generic](#) [► 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void CopyTo(  
    T[] array,  
    int arrayIndex  
)
```

Parameters

array	Type: .T [► 3087]. The array.
arrayIndex	Type: System.Int32 Index of the array.

Implements

[ICollection.T..CopyTo\(.T., Int32\)](#)

Reference

[DataTypeCollection.T. Class](#) [► 3087]

[TwinCAT.TypeSystem.Generic Namespace](#) [► 3086]

6.12.1.3.9 DataTypeCollection.T..GetEnumerator Method

Gets the enumerator.

Namespace: [TwinCAT.TypeSystem.Generic](#) [► 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IEnumerator<T> GetEnumerator()
```

Return Value

Type: [IEnumerator.T](#) [► 3087].

A [IEnumerator.T](#). that can be used to iterate through the collection.

Implements

[IEnumerable.T..GetEnumerator](#).

Reference

[DataTypeCollection.T. Class](#) [► 3087]

[TwinCAT.TypeSystem.Generic Namespace](#) [► 3086]

6.12.1.3.10 DataTypeCollection.T..IndexOf Method

Determines the Index of the specified [IDataType](#) [► 2475].

Namespace: [TwinCAT.TypeSystem.Generic](#) [► 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int IndexOf(
    T item
)
```

Parameters

item	Type: T [► 3087] The item.
------	---

Return Value

Type: [Int32](#)

The index of item if found in the list; otherwise, -1.

Implements

[IList.T..IndexOf\(T\)](#)

Reference

[DataTypeCollection.T. Class](#) [► 3087]

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.1.3.11 **DataTypeCollection.T..Insert Method**

Inserts an [IDataType \[► 2475\]](#) into the [DataTypeCollection \[► 2112\]](#).

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Insert(
    int index,
    T item
)
```

Parameters

index	Type: System.Int32 The index.
item	Type: T [► 3087] The item.

Implements

[IList.T..Insert\(Int32, T\)](#)

Reference

[DataTypeCollection.T. Class \[► 3087\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.1.3.12 **DataTypeCollection.T..LookupType Method**

Determines the specified [IDataType \[► 2475\]](#)

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T LookupType(
    string name
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: [T \[► 3087\]](#)

The [IDataType \[► 2475\]](#) if found, otherwise NULL

Reference[DataTypeCollection.T. Class \[► 3087\]](#)[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)**6.12.1.3.13 DataTypeCollection.T..Remove Method**Removes the specified [IDataType \[► 2475\]](#).**Namespace:** [TwinCAT.TypeSystem.Generic \[► 3086\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public bool Remove(
    T item
)
```

Parameters

item	Type: T [► 3087] The item.
------	---

Return Value

Type: Boolean

true if item was successfully removed from the ICollection.T.; otherwise, false. This method also returns false if item is not found in the original ICollection.T..

Implements

ICollection.T..Remove(T)

Reference[DataTypeCollection.T. Class \[► 3087\]](#)[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)**6.12.1.3.14 DataTypeCollection.T..RemoveAt Method**Removes the [IDataType \[► 2475\]](#) object at the specified index.**Namespace:** [TwinCAT.TypeSystem.Generic \[► 3086\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public void RemoveAt(
    int index
)
```

Parameters

index	Type: System.Int32 The index.
-------	----------------------------------

Implements

[IList.T..RemoveAt\(Int32\)](#)

Reference

[DataTypeCollection.T. Class \[▶ 3087\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.1.3.15 DataTypeCollection.T..TryGetType Method

Tries to get the specified [IDataType \[▶ 2475\]](#) from the [IDataTypeCollection.T. \[▶ 2486\]](#).

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetType(
    string name,
    out T? type
)
```

Parameters

name	Type: System.String The name.
type	Type: T [▶ 3087]. The type (Out parameter)

Return Value

Type: Boolean
true if found

Implements

[IDataTypeCollection.T..TryGetType\(String, T.\) \[▶ 2490\]](#)

Reference


[DataTypeCollection.T. Class \[▶ 3087\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.1.4 DataTypeCollection.T. Fields

The [DataTypeCollection.T. \[▶ 3087\]](#) generic type exposes the following members.

Fields

	Name	Description
	list [▶ 3102]	Internal list of data types

Reference[DataTypeCollection.T. Class \[► 3087\]](#)[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)**6.12.1.4.1 DataTypeCollection.T..list Field**

Internal list of data types

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
protected List<T> list
```

Field ValueType: [List.T \[► 3087\]](#).**Reference**[DataTypeCollection.T. Class \[► 3087\]](#)[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)**6.12.2 INamespace.TType. Interface**

Namespace interface

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**



```
public interface INamespace<TType>
where TType : class, IDataType
```

Type Parameters

TType	Data Type class used within this Namespace interface
-------	--

The INamespace.TType. type exposes the following members.

Properties



	Name	Description
	DataTypes [► 3103]	Data types organized by the INamespace.TType.
	Name [► 3103]	Gets the name/ identifier of the Namespace

Reference[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.2.1 INamespace.TType. Properties

The [INamespace.TType](#). [[▶ 3102](#)] generic type exposes the following members.

Properties

	Name	Description
	DataTypes [▶ 3103]	Data types organized by the INamespace.TType . [▶ 3102]
	Name [▶ 3103]	Gets the name/ identifier of the Namespace

Reference

[INamespace.TType](#). Interface [[▶ 3102](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.2.1.1 INamespace.TType..DataTypes Property

Data types organized by the [INamespace.TType](#). [[▶ 3102](#)]

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDataValueCollection<TType> DataTypes { get; }
```

Property Value

Type: [IDataValueCollection](#) [[▶ 2486](#)].[TType](#) [[▶ 3102](#)].
The data types.

Reference

[INamespace.TType](#). Interface [[▶ 3102](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.2.1.2 INamespace.TType..Name Property

Gets the name/ identifier of the Namespace

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string Name { get; }
```

Property Value

Type: String
The name.

Reference

[INamespace.TType. Interface \[► 3102\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.3 InstanceCollection.T. Class

Base class for [IInstance \[► 2549\]](#) object collections (abstract).

Inheritance Hierarchy

System.Object

TwinCAT.TypeSystem.Generic.InstanceCollection.T.

[TwinCAT.TypeSystem.FieldCollection \[► 2427\]](#)

[TwinCAT.TypeSystem.Generic.SymbolCollection.T. \[► 3165\]](#)

[TwinCAT.TypeSystem.MemberCollection \[► 2802\]](#)

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#



```
public abstract class InstanceCollection<T> : IList<T>,
    ICollection<T>, IEnumerable<T>, IEnumerable, IInstanceCollection<T>
where T : class, IInstance
```

Type Parameters





T




The InstanceCollection.T. type exposes the following members.

Constructors
























	Name	Description
	InstanceCollection.T : (InstanceCollection Mode) [► 3106]	Initializes a new instance of the InstanceCollection.T. class.
	InstanceCollection.T .(IEnumerable.T., InstanceCollectionMode) [► 3106]	Initializes a new instance of the InstanceCollection.T. class.

Properties

	Name	Description
	Count [► 3107]	Gets the collection count.
	InnerList [► 3108]	Gets the List of instances.
	InnerPathDict [► 3108]	The Path dictionary
	IsReadOnly [► 3109]	Gets a value indicating whether this instance is read only.

	Name	Description
	Item.Int32 [▶ 3109]	Gets or sets the Instance [▶ 2549] at the specified index.
	Item.String [▶ 3110]	Gets the Instance [▶ 2549] with the specified instance path.
	Mode [▶ 3111]	The mode this InstanceCollection.T. is working in.



Methods

	Name	Description
	Add [▶ 3112]	Adds the specified item.
	AddRange [▶ 3113]	Adds the specified items to this collection.
	AsReadOnly [▶ 3113]	Converts the InstanceCollection.T. to an ReadOnlyInstanceCollection.T. [▶ 3145]
	Clear [▶ 3114]	Clears this instance.
	Contains(T) [▶ 3115]	Determines whether this collection contains the specified Instance [▶ 2549]
	Contains(String) [▶ 3114]	Determines whether this collection contains an Instance [▶ 2549] with the specified InstanceName / InstancePath
	ContainsName [▶ 3116]	Determines whether the specified instance name contains name.
	CopyTo [▶ 3116]	Copies this InstanceCollection.T. to the specified array.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 3117]	Gets the enumerator.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetInstance [▶ 3117]	Gets the Instance [▶ 2549] by instance path.
	GetInstanceByName [▶ 3118]	Gets the name of the instance by.
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 3119]	Determines the index of the specified Instance [▶ 2549].
	Insert [▶ 3119]	Inserts the specified Instance [▶ 2549] at the specified index.
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 3120]	Removes the specified item.
	RemoveAt [▶ 3121]	Removes the Instance [▶ 2549] at the specified index.
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInstance [▶ 3121]	Tries to get the Instance [▶ 2549] of the specified path.
	TryGetInstanceByName [▶ 3122]	Tries to get Instances by name.

Reference

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.3.1 InstanceCollection.T. Constructor**Overload List**

	Name	Description
	InstanceCollection.T (InstanceCollectionMode) [► 3106]	Initializes a new instance of the InstanceCollection.T. [► 3104] class.
	InstanceCollection.T .(IEnumerable.T., InstanceCollectionMode) [► 3106]	Initializes a new instance of the InstanceCollection.T. [► 3104] class.

Reference

[InstanceCollection.T. Class \[► 3104\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.3.1.1 InstanceCollection.T. Constructor (InstanceCollectionMode)

Initializes a new instance of the [InstanceCollection.T. \[► 3104\]](#) class.

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
protected InstanceCollection(  
    InstanceCollectionMode mode  
)
```

Parameters

mode	Type: TwinCAT.TypeSystem.InstanceCollectionMode [► 2573] The mode.
------	---

Reference

[InstanceCollection.T. Class \[► 3104\]](#)

[InstanceCollection.T. Overload \[► 3106\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.3.1.2 InstanceCollection.T. Constructor (IEnumerable.T., InstanceCollectionMode)

Initializes a new instance of the [InstanceCollection.T. \[► 3104\]](#) class.

Namespace: [TwinCAT.TypeSystem.Generic](#) [▶ 3086]
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected InstanceCollection(
    IEnumerable<T> coll,
    InstanceCollectionMode mode
)
```

Parameters

coll	Type: System.Collections.Generic.IEnumerable.T [▶ 3104]. The copy collection
mode	Type: TwinCAT.TypeSystem.InstanceCollectionMode [▶ 2573] The mode.








Reference

- [InstanceCollection.T. Class](#) [▶ 3104]
- [InstanceCollection.T. Overload](#) [▶ 3106]
- [TwinCAT.TypeSystem.Generic Namespace](#) [▶ 3086]

6.12.3.2 InstanceCollection.T. Properties

The [InstanceCollection.T.](#) [▶ 3104] generic type exposes the following members.

Properties

	Name	Description
	Count [▶ 3107]	Gets the collection count.
	InnerList [▶ 3108]	Gets the List of instances.
	InnerPathDict [▶ 3108]	The Path dictionary
	IsReadOnly [▶ 3109]	Gets a value indicating whether this instance is read only.
	Item.Int32. [▶ 3109]	Gets or sets the Instance [▶ 2549] at the specified index.
	Item.String. [▶ 3110]	Gets the Instance [▶ 2549] with the specified instance path.
	Mode [▶ 3111]	The mode this InstanceCollection.T. [▶ 3104] is working in.

Reference

- [InstanceCollection.T. Class](#) [▶ 3104]
- [TwinCAT.TypeSystem.Generic Namespace](#) [▶ 3086]

6.12.3.2.1 InstanceCollection.T.Count Property

Gets the collection count.

Namespace: [TwinCAT.TypeSystem.Generic](#) [▶ 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Count { get; }
```

Property Value

Type: Int32
The count.

Implements

ICollection.T..Count

Reference

[InstanceCollection.T. Class](#) [▶ 3104]

[TwinCAT.TypeSystem.Generic Namespace](#) [▶ 3086]

6.12.3.2 InstanceCollection.T..InnerList Property

Gets the List of instances.

Namespace: [TwinCAT.TypeSystem.Generic](#) [▶ 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected IList<T> InnerList { get; }
```

Property Value

Type: [IList.T](#) [▶ 3104].
The inner list.

Reference

[InstanceCollection.T. Class](#) [▶ 3104]

[TwinCAT.TypeSystem.Generic Namespace](#) [▶ 3086]

6.12.3.2.3 InstanceCollection.T..InnerPathDict Property

The Path dictionary

Namespace: [TwinCAT.TypeSystem.Generic](#) [▶ 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected IDictionary<string, T> InnerPathDict { get; }
```

Property Value

Type: IDictionary.String, T [▶ 3104].

Reference

[InstanceCollection.T. Class \[▶ 3104\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.3.2.4 InstanceCollection.T..IsReadOnly Property

Gets a value indicating whether this instance is read only.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsReadOnly { get; }
```

Property Value

Type: Boolean
true if this instance is read only; otherwise, false.

Implements

ICollection.T..IsReadOnly



Reference

[InstanceCollection.T. Class \[▶ 3104\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.3.2.5 InstanceCollection.T..Item Property

Overload List

	Name	Description
	Item.Int32. [▶ 3109]	Gets or sets the Instance [▶ 2549] at the specified index.
	Item.String. [▶ 3110]	Gets the Instance [▶ 2549] with the specified instance path.

Reference

[InstanceCollection.T. Class \[▶ 3104\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.3.2.5.1 InstanceCollection.T..Item Property (Int32)

Gets or sets the [Instance \[▶ 2549\]](#) at the specified index.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T this[
    int index
] { get; set; }
```

Parameters

index	Type: System.Int32 The index.
-------	----------------------------------

Return Value

Type: [T](#) [[▶ 3104](#)]

T.

Implements

[IList.T..Item.Int32](#).

Exceptions

Exception	Condition
NotImplementedException	

Reference

[InstanceCollection.T. Class](#) [[▶ 3104](#)]

[Item Overload](#) [[▶ 3109](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.3.2.5.2 InstanceCollection.T..Item Property (String)

Gets the [IInstance](#) [[▶ 2549](#)] with the specified instance path.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T this[
    string instanceSpecifier
] { get; }
```

Parameters

instanceSpecifier	Type: System.String The instance path or Instance Name (dependent of Mode [▶ 3111] setting)
-------------------	--

Return Value

Type: [T](#) [[▶ 3104](#)]

T.

Implements

[IInstanceCollection.T..Item.String.](#) [[▶ 2556](#)]

Exceptions

Exception	Condition
ArgumentNullException	
ArgumentException	

Remarks

Dependent what this [InstanceCollection.T.](#) [[▶ 3104](#)] contains configured by the [InstanceCollectionMode](#) [[▶ 2573](#)] the instance specifier should be the [InstanceName](#) [[▶ 2551](#)] or the [InstancePath](#) [[▶ 2552](#)].

Reference

[InstanceCollection.T. Class](#) [[▶ 3104](#)]

[Item Overload](#) [[▶ 3109](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.3.2.6 InstanceCollection.T..Mode Property

The mode this [InstanceCollection.T.](#) [[▶ 3104](#)] is working in.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public InstanceCollectionMode Mode { get; }
```

Property Value

Type: [InstanceCollectionMode](#) [[▶ 2573](#)]

Implements

[IInstanceCollection.T..Mode](#) [[▶ 2556](#)]

Reference


[InstanceCollection.T. Class](#) [[▶ 3104](#)]























[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.3.3 InstanceCollection.T. Methods

The [InstanceCollection.T.](#) [[▶ 3104](#)] generic type exposes the following members.

Methods

	Name	Description
	Add [▶ 3112]	Adds the specified item.

	Name	Description
	AddRange [▶ 3113]	Adds the specified items to this collection.
	AsReadOnly [▶ 3113]	Converts the InstanceCollection.T. [▶ 3104] to an ReadOnlyInstanceCollection.T. [▶ 3145]
	Clear [▶ 3114]	Clears this instance.
	Contains(T) [▶ 3115]	Determines whether this collection contains the specified Instance [▶ 2549]
	Contains(String) [▶ 3114]	Determines whether this collection contains an Instance [▶ 2549] with the specified InstanceName / InstancePath
	ContainsName [▶ 3116]	Determines whether the specified instance name contains name.
	CopyTo [▶ 3116]	Copies this InstanceCollection.T. [▶ 3104] to the specified array.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 3117]	Gets the enumerator.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetInstance [▶ 3117]	Gets the Instance [▶ 2549] by instance path.
	GetInstanceByName [▶ 3118]	Gets the name of the instance by.
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 3119]	Determines the index of the specified Instance [▶ 2549] .
	Insert [▶ 3119]	Inserts the specified Instance [▶ 2549] at the specified index.
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 3120]	Removes the specified item.
	RemoveAt [▶ 3121]	Removes the Instance [▶ 2549] at the specified index.
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInstance [▶ 3121]	Tries to get the Instance [▶ 2549] of the specified path.
	TryGetInstanceByName [▶ 3122]	Tries to get Instances by name.

Reference

[InstanceCollection.T. Class \[▶ 3104\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.3.3.1 InstanceCollection.T..Add Method

Adds the specified item.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public virtual void Add(
    T item
)
```

Parameters

item	Type: T [▶ 3104] The item.
------	---

Implements

[ICollection.T..Add\(T\)](#)

Reference

[InstanceCollection.T. Class](#) [[▶ 3104](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.3.3.2 InstanceCollection.T..AddRange Method

Adds the specified items to this collection.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void AddRange(
    IEnumerable<T> items
)
```

Parameters

items	Type: System.Collections.Generic.IEnumerable.T [▶ 3104]. The items.
-------	--

Reference

[InstanceCollection.T. Class](#) [[▶ 3104](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.3.3.3 InstanceCollection.T..AsReadOnly Method

Converts the [InstanceCollection.T.](#) [[▶ 3104](#)] to an [ReadOnlyInstanceCollection.T.](#) [[▶ 3145](#)]

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyInstanceCollection<T> AsReadOnly()
```

Return Value

Type: [ReadOnlyInstanceCollection](#) [[▶ 3145](#)].T [[▶ 3104](#)].
 ReadOnlyInstanceCollection<T>.

Reference

[InstanceCollection.T. Class](#) [[▶ 3104](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.3.3.4 InstanceCollection.T..Clear Method

Clears this instance.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public void Clear()
```

Implements



[ICollection.T..Clear](#).

Reference

[InstanceCollection.T. Class](#) [[▶ 3104](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.3.3.5 InstanceCollection.T..Contains Method**Overload List**

	Name	Description
	Contains(T) [▶ 3115]	Determines whether this collection contains the specified IInstance [▶ 2549]
	Contains(String) [▶ 3114]	Determines whether this collection contains an IInstance [▶ 2549] with the specified InstanceName / InstancePath

Reference

[InstanceCollection.T. Class](#) [[▶ 3104](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.3.3.5.1 InstanceCollection.T..Contains Method (String)

Determines whether this collection contains an [IInstance](#) [[▶ 2549](#)] with the specified InstanceName / InstancePath

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(
    string instanceSpecifier
)
```

Parameters

instanceSpecifier	Type: System.String The instance path or Instance Name (dependent of Mode [▶_3111] setting)
-------------------	--

Return Value

Type: Boolean
true if [contains] [the specified instance path]; otherwise, false.

Implements

[ICollection.T.Contains\(String\) \[▶_2558\]](#)

Exceptions

Exception	Condition
ArgumentNullException	instancePath
ArgumentException	

Reference

[ICollection.T. Class \[▶_3104\]](#)

[Contains Overload \[▶_3114\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶_3086\]](#)

6.12.3.3.5.2 InstanceCollection.T.Contains Method (T)

Determines whether this collection contains the specified [Instance \[▶_2549\]](#)

Namespace: [TwinCAT.TypeSystem.Generic \[▶_3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(
    T item
)
```

Parameters

item	Type: T [▶_3104] The item.
------	---

Return Value

Type: Boolean
true if [contains] [the specified item]; otherwise, false.

Implements

[ICollection.T..Contains\(T\)](#)

Reference

[InstanceCollection.T. Class \[▶ 3104\]](#)

[Contains Overload \[▶ 3114\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.3.3.6 InstanceCollection.T..ContainsName Method

Determines whether the specified instance name contains name.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public bool ContainsName(
    string instanceName
)
```

Parameters

instanceName	Type: System.String Name of the instance.
--------------	--

Return Value

Type: Boolean
true if the specified instance name contains name; otherwise, false.

Implements

[ICollection.T..ContainsName\(String\) \[▶ 2558\]](#)

Exceptions

Exception	Condition
NotImplementedException	

Reference

[InstanceCollection.T. Class \[▶ 3104\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.3.3.7 InstanceCollection.T..CopyTo Method

Copies this [InstanceCollection.T. \[▶ 3104\]](#) to the specified array.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void CopyTo(  
    T[] array,  
    int arrayIndex  
)
```

Parameters

array	Type: T [▶ 3104]. The array.
arrayIndex	Type: System.Int32 Index of the array.

Implements

ICollection.T..CopyTo(.T., Int32)

Reference

[InstanceCollection.T. Class](#) [[▶ 3104](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.3.3.8 InstanceCollection.T..GetEnumerator Method

Gets the enumerator.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IEnumerator<T> GetEnumerator()
```

Return Value

Type: [IEnumerator.T](#) [[▶ 3104](#)].

A [IEnumerator.T](#). that can be used to iterate through the collection.

Implements

IEnumerable.T..GetEnumerator.

Reference

[InstanceCollection.T. Class](#) [[▶ 3104](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.3.3.9 InstanceCollection.T..GetInstance Method

Gets the [IInstance](#) [[▶ 2549](#)]by instance path.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T GetInstance(
    string instanceSpecifier
)
```

Parameters

instanceSpecifier	Type: System.String The instance path or Instance Name (dependent of Mode [▶ 3111] setting)
-------------------	--

Return Value

Type: [T \[▶ 3104\]](#)
T.

Implements

[IInstanceCollection.T..GetInstance\(String\) \[▶ 2559\]](#)

Exceptions

Exception	Condition
ArgumentException	Path not found!;instancePath

Reference

[InstanceCollection.T. Class \[▶ 3104\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.3.3.10 InstanceCollection.T..GetInstanceByName Method

Gets the name of the instance by.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IList<T> GetInstanceByName(
    string instanceName
)
```

Parameters

instanceName	Type: System.String Name of the instance.
--------------	--

Return Value

Type: [IList.T \[▶ 3104\]](#).
[IList<T>](#).

Implements

[IInstanceCollection.T..GetInstanceByName\(String\) \[▶ 2559\]](#)

Exceptions

Exception	Condition
ArgumentException	Name not found!;instanceName

Reference

[InstanceCollection.T. Class](#) [► 3104]

[TwinCAT.TypeSystem.Generic Namespace](#) [► 3086]

6.12.3.3.11 InstanceCollection.T..IndexOf Method

Determines the index of the specified [IInstance](#) [► 2549].

Namespace: [TwinCAT.TypeSystem.Generic](#) [► 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int IndexOf(  
    T item  
)
```

Parameters

item	Type: T [► 3104] The item.
------	---

Return Value

Type: Int32

The index of item if found in the list; otherwise, -1.

Implements

[IList.T..IndexOf\(T\)](#)

Reference

[InstanceCollection.T. Class](#) [► 3104]

[TwinCAT.TypeSystem.Generic Namespace](#) [► 3086]

6.12.3.3.12 InstanceCollection.T..Insert Method

Inserts the specified [IInstance](#) [► 2549] at the specified index.

Namespace: [TwinCAT.TypeSystem.Generic](#) [► 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public virtual void Insert(  
    int index,  
    T instance  
)
```

Parameters

index	Type: System.Int32 The instance.
instance	Type: T [▶ 3104] The item.

Implements

[ICollection.T..Insert\(Int32, T\)](#)

Exceptions

Exception	Condition
ArgumentOutOfRangeException	index or index
ArgumentNullException	

Reference

[InstanceCollection.T. Class](#) [[▶ 3104](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.3.3.13 InstanceCollection.T..Remove Method

Removes the specified item.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public bool Remove (
    T item
)
```

Parameters

item	Type: T [▶ 3104] The item.
------	---

Return Value

Type: Boolean

true if item was successfully removed from the [ICollection.T.](#); otherwise, false. This method also returns false if item is not found in the original [ICollection.T.](#).

Implements

[ICollection.T..Remove\(T\)](#)

Reference

[InstanceCollection.T. Class](#) [[▶ 3104](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.3.3.14 InstanceCollection.T..RemoveAt Method

Removes the [IInstance](#) [[▶ 2549](#)] at the specified index.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public virtual void RemoveAt(
    int index
)
```

Parameters

index	Type: System.Int32 The index.
-------	----------------------------------

Implements

[IList.T..RemoveAt\(Int32\)](#)

Reference

[InstanceCollection.T. Class](#) [[▶ 3104](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.3.3.15 InstanceCollection.T..TryGetInstance Method

Tries to get the [IInstance](#) [[▶ 2549](#)]. of the specified path.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetInstance(
    string instanceSpecifier,
    out T? symbol
)
```

Parameters

instanceSpecifier	Type: System.String The instance path or Instance Name (dependent of Mode [▶ 3111] setting)
symbol	Type: T [▶ 3104]. The symbol.

Return Value

Type: Boolean

true if the [IInstance](#) [[▶ 2549](#)] is found; otherwise, false

Implements

[IInstanceCollection.T..TryGetInstance\(String, T.\)](#) [[▶ 2560](#)]

Exceptions

Exception	Condition
ArgumentNullException	instancePath
ArgumentException	

Reference

[InstanceCollection.T. Class \[► 3104\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.3.3.16 InstanceCollection.T..TryGetInstanceByName Method

Tries to get Instances by name.

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public virtual bool TryGetInstanceByName (
    string instanceName,
    out IList<T>? instances
)
```

Parameters

instanceName	Type: System.String Name of the instance.
instances	Type: System.Collections.Generic.IList.T [► 3104].. The instances found.

Return Value

Type: Boolean

true if the [IInstance \[► 2549\]](#) is found; otherwise, false

Implements

[IInstanceCollection.T..TryGetInstanceByName\(String, IList.T..\) \[► 2560\]](#)

Reference

[InstanceCollection.T. Class \[► 3104\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.4 ISymbolProvider.TNamespace, TDataType, TSymbol. Interface

Symbol provider interface

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#






```
public interface ISymbolProvider<TNamespace, TDataType, TSymbol>
where TDataType : class, IDataType
where TSymbol : class, ISymbol
```

Type Parameters

TNamespace	Namespace type
TDataType	DataType type
TSymbol	Symbol type

The ISymbolProvider.TNamespace, TDataType, TSymbol. type exposes the following members.

Properties

	Name	Description
	DataTypes [▶ 3124]	Gets all data types from all Namespaces
	Namespaces [▶ 3124]	Get the Namespaces of DataTypes for this Symbol provider
	RootNamespace [▶ 3124]	Gets the root (main) namespace of the Symbol provider.
	RootNamespaceName [▶ 3125]	Gets the name of the root namespace
	Symbols [▶ 3125]	Gets the (root) symbols of the Symbol provider.






Reference

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.4.1 ISymbolProvider.TNamespace, TDataType, TSymbol. Properties

The [ISymbolProvider.TNamespace, TDataType, TSymbol. \[▶ 3122\]](#) generic type exposes the following members.

Properties

	Name	Description
	DataTypes [▶ 3124]	Gets all data types from all Namespaces
	Namespaces [▶ 3124]	Get the Namespaces of DataTypes for this Symbol provider
	RootNamespace [▶ 3124]	Gets the root (main) namespace of the Symbol provider.
	RootNamespaceName [▶ 3125]	Gets the name of the root namespace
	Symbols [▶ 3125]	Gets the (root) symbols of the Symbol provider.

Reference

[ISymbolProvider.TNamespace, TDataType, TSymbol. Interface \[▶ 3122\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.4.1.1 **ISymbolProvider.TNamespace, TDataType, TSymbol..DataTypes Property**

Gets all data types from all Namespaces

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IDataTypeCollection<TDataType> DataTypes { get; }
```

Property Value

Type: [IDataTypeCollection \[► 2486\]](#).[TDataType \[► 3122\]](#).
The data types.

Reference

[ISymbolProvider.TNamespace, TDataType, TSymbol. Interface \[► 3122\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.4.1.2 **ISymbolProvider.TNamespace, TDataType, TSymbol..Namespaces Property**

Get the Namespaces of DataTypes for this Symbol provider

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
INamespaceCollection<TDataType> Namespaces { get; }
```

Property Value

Type: [INamespaceCollection \[► 2571\]](#).[TDataType \[► 3122\]](#).
ReadOnly collection of the namespaces.

Reference

[ISymbolProvider.TNamespace, TDataType, TSymbol. Interface \[► 3122\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.4.1.3 **ISymbolProvider.TNamespace, TDataType, TSymbol..RootNamespace Property**

Gets the root (main) namespace of the Symbol provider.

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
INamespace<TDataType>? RootNamespace { get; }
```

Property Value

Type: [INamespace](#) [[▶ 3102](#)].[TDataType](#) [[▶ 3122](#)].

The root namespace.

Reference

[ISymbolProvider.TNamespace, TDataType, TSymbol. Interface](#) [[▶ 3122](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.4.1.4 **ISymbolProvider.TNamespace, TDataType, TSymbol..RootNamespaceName Property**

Gets the name of the root namespace

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
string RootNamespaceName { get; }
```

Property Value

Type: String

The namespace.

Reference

[ISymbolProvider.TNamespace, TDataType, TSymbol. Interface](#) [[▶ 3122](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.4.1.5 **ISymbolProvider.TNamespace, TDataType, TSymbol..Symbols Property**

Gets the (root) symbols of the Symbol provider.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
ISymbolCollection<TSymbol> Symbols { get; }
```

Property Value

Type: [ISymbolCollection](#) [[▶ 2700](#)].[TSymbol](#) [[▶ 3122](#)].

Read only collection of the Symbols

Reference

[ISymbolProvider.TNamespace, TDataType, TSymbol. Interface \[► 3122\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.5 NamespaceCollection.T. Class

Generic class for Namespace collections

Inheritance Hierarchy

System.Object

TwinCAT.TypeSystem.Generic.NamespaceCollection.T.

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#


```
public class NamespaceCollection<T> : IList<INamespace<T>>,
    ICollection<INamespace<T>>, IEnumerable<INamespace<T>>, IEnumerable<
where T : class, IDatatype
```

Type Parameters







T

The NamespaceCollection.T. type exposes the following members.



Constructors

	Name	Description
	NamespaceCollection.T. [► 3127]	Initializes a new instance of the NamespaceCollection.T. class.

Properties

	Name	Description
	AllTypes [► 3128]	Gets all types included in all namespaces.
	Count [► 3128]	Gets the number of elements contained in the ICollection.T..
	InnerAllTypes [► 3129]	Dictionary FullPath -> IDatatype
	IsReadOnly [► 3129]	Gets a value indicating whether the ICollection.T. is read-only.
	Item.Int32. [► 3130]	Gets or sets the element at the specified index.
	Item.String. [► 3131]	Gets or sets the element at the specified index.

Methods

	Name	Description
	Add [► 3132]	Adds an item to the ICollection.T..
	Clear [► 3133]	Removes all items from the ICollection.T..

	Name	Description
	Contains [▶ 3133]	Determines whether the ICollection.T. contains a specific value.
	ContainsNamespace [▶ 3134]	Determines whether the specified name contains namespace.
	CopyTo [▶ 3134]	Copies to.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 3135]	Returns an enumerator that iterates through the collection.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 3135]	Determines the index of a specific item in the IList.T..
	Insert [▶ 3136]	Inserts an item to the IList.T. at the specified index.
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 3136]	Removes the first occurrence of a specific object from the ICollection.T..
	RemoveAt [▶ 3137]	Removes the IList.T. item at the specified index.
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetNamespace [▶ 3137]	Tries to get the namespace object
	TryGetType [▶ 3138]	Tries to get the specified type.
	TryGetTypeByFullName [▶ 3139]	Tries to get the data type by full name.

Reference

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.5.1 NamespaceCollection.T. Constructor

Initializes a new instance of the [NamespaceCollection.T. \[▶ 3126\]](#) class.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public NamespaceCollection()
```

Reference







[NamespaceCollection.T. Class \[▶ 3126\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.5.2 NamespaceCollection.T. Properties

The [NamespaceCollection.T.](#) [[▶ 3126](#)] generic type exposes the following members.

Properties

	Name	Description
	AllTypes [▶ 3128]	Gets all types included in all namespaces.
	Count [▶ 3128]	Gets the number of elements contained in the ICollection.T..
	InnerAllTypes [▶ 3129]	Dictionary FullPath -> IDataType
	IsReadOnly [▶ 3129]	Gets a value indicating whether the ICollection.T. is read-only.
	Item.Int32. [▶ 3130]	Gets or sets the element at the specified index.
	Item.String. [▶ 3131]	Gets or sets the element at the specified index.

Reference

[NamespaceCollection.T. Class](#) [[▶ 3126](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.5.2.1 NamespaceCollection.T..AllTypes Property

Gets all types included in all namespaces.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IDataTypeCollection<T> AllTypes { get; }
```

Property Value

Type: [IDataTypeCollection](#) [[▶ 2486](#)].T [[▶ 3126](#)].

All types.

Reference

[NamespaceCollection.T. Class](#) [[▶ 3126](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.5.2.2 NamespaceCollection.T..Count Property

Gets the number of elements contained in the ICollection.T..

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int Count { get; }
```

Return Value

Type: Int32

The number of elements contained in the ICollection.T..

Implements

ICollection.T..Count

Reference

[NamespaceCollection.T. Class \[► 3126\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.5.2.3 NamespaceCollection.T..InnerAllTypes Property

Dictionary FullPath -> IDataType

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected IDictionary<string, T> InnerAllTypes { get; }
```

Property Value

Type: [IDictionary.String, T \[► 3126\]](#).

Reference

[NamespaceCollection.T. Class \[► 3126\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.5.2.4 NamespaceCollection.T..IsReadOnly Property

Gets a value indicating whether the ICollection.T. is read-only.

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsReadOnly { get; }
```

Return Value



Type: Boolean

true if the ICollection.T. is read-only; otherwise, false.

Implements

ICollection.T..IsReadOnly

Reference[NamespaceCollection.T. Class \[▸ 3126\]](#)[TwinCAT.TypeSystem.Generic Namespace \[▸ 3086\]](#)**6.12.5.2.5 NamespaceCollection.T..Item Property****Overload List**

	Name	Description
	Item.Int32. [▸ 3130]	Gets or sets the element at the specified index.
	Item.String. [▸ 3131]	Gets or sets the element at the specified index.

Reference[NamespaceCollection.T. Class \[▸ 3126\]](#)[TwinCAT.TypeSystem.Generic Namespace \[▸ 3086\]](#)**6.12.5.2.5.1 NamespaceCollection.T..Item Property (Int32)**

Gets or sets the element at the specified index.

Namespace: [TwinCAT.TypeSystem.Generic \[▸ 3086\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public INamespace<T> this[
    int index
] { get; set; }
```

Parameters

index	Type: System.Int32 The index.
-------	----------------------------------

Return ValueType: [INamespace \[▸ 3102\].T \[▸ 3126\]](#).**Implements**

IList.T..Item.Int32.

Exceptions

Exception	Condition
NotSupportedException	

Reference

[NamespaceCollection.T. Class \[▶ 3126\]](#)

[Item Overload \[▶ 3130\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.5.2.5.2 NamespaceCollection.T..Item Property (String)

Gets or sets the element at the specified index.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public INamespace<T> this[
    string str
] { get; }
```

Parameters

str	Type: System.String The STR.
-----	---------------------------------

Return Value

Type: [INamespace \[▶ 3102\].T \[▶ 3126\]](#).

Reference

[NamespaceCollection.T. Class \[▶ 3126\]](#)







[Item Overload \[▶ 3130\]](#)














[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.5.3 NamespaceCollection.T. Methods

The [NamespaceCollection.T. \[▶ 3126\]](#) generic type exposes the following members.

Methods

	Name	Description
	Add [▶ 3132]	Adds an item to the ICollection.T..
	Clear [▶ 3133]	Removes all items from the ICollection.T..
	Contains [▶ 3133]	Determines whether the ICollection.T. contains a specific value.
	ContainsNamespace [▶ 3134]	Determines whether the specified name contains namespace.
	CopyTo [▶ 3134]	Copies to.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)

	Name	Description
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 3135]	Returns an enumerator that iterates through the collection.
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 3135]	Determines the index of a specific item in the IList.T..
	Insert [▶ 3136]	Inserts an item to the IList.T. at the specified index.
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 3136]	Removes the first occurrence of a specific object from the ICollection.T..
	RemoveAt [▶ 3137]	Removes the IList.T. item at the specified index.
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetNamespace [▶ 3137]	Tries to get the namespace object
	TryGetType [▶ 3138]	Tries to get the specified type.
	TryGetTypeByFullName [▶ 3139]	Tries to get the data type by full name.

Reference

[NamespaceCollection.T. Class](#) [▶ 3126]

[TwinCAT.TypeSystem.Generic Namespace](#) [▶ 3086]

6.12.5.3.1 NamespaceCollection.T..Add Method

Adds an item to the ICollection.T..

Namespace: [TwinCAT.TypeSystem.Generic](#) [▶ 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Add(
    INamespace<T> item
)
```

Parameters

item	Type: TwinCAT.TypeSystem.Generic.INamespace [▶ 3102].T [▶ 3126]. The object to add to the ICollection.T..
------	--

Implements

ICollection.T..Add(T)

Reference

[NamespaceCollection.T. Class \[► 3126\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.5.3.2 NamespaceCollection.T..Clear Method

Removes all items from the ICollection.T..

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void Clear()
```

Implements

ICollection.T..Clear.

Reference

[NamespaceCollection.T. Class \[► 3126\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.5.3.3 NamespaceCollection.T..Contains Method

Determines whether the ICollection.T. contains a specific value.

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(  
    INamespace<T> item  
)
```

Parameters

item	Type: TwinCAT.TypeSystem.Generic.INamespace [► 3102].T [► 3126] . The object to locate in the ICollection.T..
------	--

Return Value

Type: Boolean
true if item is found in the ICollection.T.; otherwise, false.

Implements

ICollection.T..Contains(T)

Reference

[NamespaceCollection.T. Class \[► 3126\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.5.3.4 NamespaceCollection.T..ContainsNamespace Method

Determines whether the specified name contains namespace.

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool ContainsNamespace(
    string name
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: Boolean

true if the specified name contains namespace; otherwise, false.

Reference

[NamespaceCollection.T. Class \[► 3126\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.5.3.5 NamespaceCollection.T..CopyTo Method

Copies to.

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void CopyTo(
    INamespace<T>[] array,
    int arrayIndex
)
```

Parameters

array	Type: .TwinCAT.TypeSystem.Generic.INamespace [► 3102].T [► 3126] .. The array.
arrayIndex	Type: System.Int32 Index of the array.

Implements

[ICollection.T..CopyTo\(.T., Int32\)](#)

Reference

[NamespaceCollection.T. Class \[► 3126\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.5.3.6 NamespaceCollection.T..GetEnumerator Method

Returns an enumerator that iterates through the collection.

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IEnumerator<INamespace<T>> GetEnumerator()
```

Return Value

Type: [IEnumerator.INamespace \[► 3102\].T \[► 3126\]](#)..

A [IEnumerator.T.](#) that can be used to iterate through the collection.

Implements

[IEnumerable.T..GetEnumerator.](#)

Reference

[NamespaceCollection.T. Class \[► 3126\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.5.3.7 NamespaceCollection.T..IndexOf Method

Determines the index of a specific item in the [IList.T.](#)

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int IndexOf(  
    INamespace<T> item  
)
```

Parameters

item	Type: TwinCAT.TypeSystem.Generic.INamespace [► 3102].T [► 3126] . The object to locate in the IList.T.
------	---

Return Value

Type: [Int32](#)

The index of item if found in the list; otherwise, -1.

Implements

[IList.T..IndexOf\(T\)](#)

Reference

[NamespaceCollection.T. Class](#) [[▶ 3126](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.5.3.8 NamespaceCollection.T..Insert Method

Inserts an item to the [IList.T.](#) at the specified index.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public void Insert(
    int index,
    INamespace<T> item
)
```

Parameters

index	Type: System.Int32 The zero-based index at which item should be inserted.
item	Type: TwinCAT.TypeSystem.Generic.INamespace [▶ 3102]. T [▶ 3126]. The object to insert into the IList.T.

Implements

[IList.T..Insert\(Int32, T\)](#)

Reference

[NamespaceCollection.T. Class](#) [[▶ 3126](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.5.3.9 NamespaceCollection.T..Remove Method

Removes the first occurrence of a specific object from the [ICollection.T.](#)

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public bool Remove(
    INamespace<T> item
)
```


Parameters

item	Type: TwinCAT.TypeSystem.Generic.INamespace [▶ 3102]. T [▶ 3126]. The object to remove from the ICollection.T. .
------	---

Return Value

Type: Boolean

true if item was successfully removed from the [ICollection.T.](#); otherwise, false. This method also returns false if item is not found in the original [ICollection.T.](#).

Implements

[ICollection.T.](#).[Remove\(T\)](#)

Reference

[NamespaceCollection.T. Class](#) [[▶ 3126](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.5.3.10 NamespaceCollection.T..RemoveAt Method

Removes the [IList.T.](#) item at the specified index.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void RemoveAt(  
    int index  
)
```

Parameters

index	Type: System.Int32 The zero-based index of the item to remove.
-------	---

Implements

[IList.T.](#).[RemoveAt\(Int32\)](#)

Reference

[NamespaceCollection.T. Class](#) [[▶ 3126](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.5.3.11 NamespaceCollection.T..TryGetNamespace Method

Tries to get the namespace object

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetNamespace(
    string name,
    out INamespace<T>? nspace
)
```

Parameters

name	Type: System.String The name.
nspace	Type: TwinCAT.TypeSystem.Generic.INamespace [▶ 3102]. T [▶ 3126]. The namespace object (out-parameter)

Return Value

Type: Boolean
true if found, false if not contained.

Reference

[NamespaceCollection.T. Class](#) [[▶ 3126](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.5.3.12 NamespaceCollection.T..TryGetType Method

Tries to get the specified type.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetType(
    string typeName,
    out T? dataType
)
```

Parameters

typeName	Type: System.String Data type name
dataType	Type: T [▶ 3126]. The found data type (out-parameter).

Return Value

Type: Boolean
true if found, false if not contained.

Exceptions

Exception	Condition
ArgumentNullException	typeName
ArgumentException	

Reference[NamespaceCollection.T. Class \[► 3126\]](#)[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)**6.12.5.3.13 NamespaceCollection.T..TryGetTypeByFullName Method**

Tries to get the data type by full name.

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public bool TryGetTypeByFullName(
    string fullname,
    out T? dataType
)
```

Parameters

fullname	Type: System.String DataTypes full name.
dataType	Type: T [► 3126]. Found data type (out-parameter).

Return Value

Type: Boolean

true if found, false if not contained.

Reference[NamespaceCollection.T. Class \[► 3126\]](#)[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)**6.12.6 ReadOnlyDataTypeCollection.T. Class**

ReadOnly DataType collection

Inheritance Hierarchy

System.Object

System.Collections.ObjectModel.ReadOnlyCollection.T.

TwinCAT.TypeSystem.Generic.ReadOnlyDataTypeCollection.T.

[TwinCAT.TypeSystem.ReadOnlyDataTypeCollection \[► 2815\]](#)**Namespace:** [TwinCAT.TypeSystem.Generic \[► 3086\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**



```
public class ReadOnlyDataTypeCollection<T> : ReadOnlyCollection<T>,
    IDataValueCollection<T>, IList<T>, ICollection<T>, IEnumerable<T>,
    IEnumerable
where T : class, IDataType
```

Type Parameters





T

The ReadOnlyDataTypeCollection.T. type exposes the following members.











Constructors



	Name	Description
	ReadOnlyDataTypeCollection.T. (DataTypeCollection.T.) [▶ 3141]	Initializes a new instance of the ReadOnlyDataTypeCollection.T. class.
	ReadOnlyDataTypeCollection.T. (ReadOnlyDataTypeCollection.T.) [▶ 3142]	Initializes a new instance of the ReadOnlyDataTypeCollection.T. class.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.T..)
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.T..)
	Item.String. [▶ 3143]	Gets the element with the specified type name.
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.T..)

Methods

	Name	Description
	Contains	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.T..)
	ContainsType [▶ 3144]	Determines whether the specified name contains type.
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.T..)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.T..)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.T..)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)



	Name	Description
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetType [▶ 3145]	Tries to get the Type with the specified name out of the collection.

Reference

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.6.1 ReadOnlyDataTypeCollection.T. Constructor

Overload List

	Name	Description
	ReadOnlyDataTypeCollection.T. (DataTypeCollection.T.) [▶ 3141]	Initializes a new instance of the ReadOnlyDataTypeCollection.T. [▶ 3139] class.
	ReadOnlyDataTypeCollection.T. (ReadOnlyDataTypeCollection.T.) [▶ 3142]	Initializes a new instance of the ReadOnlyDataTypeCollection.T. [▶ 3139] class.

Reference

[ReadOnlyDataTypeCollection.T. Class \[▶ 3139\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.6.1.1 ReadOnlyDataTypeCollection.T. Constructor (DataTypeCollection.T.)

Initializes a new instance of the [ReadOnlyDataTypeCollection.T. \[▶ 3139\]](#) class.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyDataTypeCollection(
    DataTypeCollection<T> coll
)
```

Parameters

coll	Type: TwinCAT.TypeSystem.Generic.DataTypeCollection [▶ 3087].T [▶ 3139] . The collection.
------	--

Reference

[ReadOnlyDataTypeCollection.T. Class \[▶ 3139\]](#)

[ReadOnlyDataTypeCollection.T. Overload \[▶ 3141\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.6.1.2 ReadOnlyDataTypeCollection.T. Constructor (ReadOnlyDataTypeCollection.T.)

Initializes a new instance of the [ReadOnlyDataTypeCollection.T.](#) [[▶ 3139](#)] class.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyDataTypeCollection(
    ReadOnlyDataTypeCollection<T> coll
)
```

Parameters

coll	Type: TwinCAT.TypeSystem.Generic.ReadOnlyDataTypeCollection [▶ 3139].T [▶ 3139]. The coll.
------	---

Reference

[ReadOnlyDataTypeCollection.T. Class](#) [[▶ 3139](#)]





[ReadOnlyDataTypeCollection.T. Overload](#) [[▶ 3141](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.6.2 ReadOnlyDataTypeCollection.T. Properties

The [ReadOnlyDataTypeCollection.T.](#) [[▶ 3139](#)] generic type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.T [▶ 3139]..)
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.T [▶ 3139]..)
	Item.String. [▶ 3143]	Gets the element with the specified type name.
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.T [▶ 3139]..)


Reference


[ReadOnlyDataTypeCollection.T. Class](#) [[▶ 3139](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.6.2.1 ReadOnlyDataTypeCollection.T..Item Property

Overload List

	Name	Description
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.T [▶ 3139]..)

	Name	Description
	Item.String. [▶ 3143]	Gets the element with the specified type name.

Reference

[ReadOnlyDataTypeCollection.T. Class \[▶ 3139\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.6.2.1 ReadOnlyDataTypeCollection.T.Item Property (String)

Gets the element with the specified type name.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T this[
    string name
] { get; }
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: [T \[▶ 3139\]](#)

T.

Implements

[IDataTypeCollection.T.Item.String. \[▶ 2488\]](#)

Reference

[ReadOnlyDataTypeCollection.T. Class \[▶ 3139\]](#)



[Item Overload \[▶ 3142\]](#)











[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.6.3 ReadOnlyDataTypeCollection.T. Methods

The [ReadOnlyDataTypeCollection.T. \[▶ 3139\]](#) generic type exposes the following members.

Methods

	Name	Description
	Contains	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.T [▶ 3139] ..)
	ContainsType [▶ 3144]	Determines whether the specified name contains type.

	Name	Description
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.T [▶ 3139].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.T [▶ 3139].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.T [▶ 3139].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetType [▶ 3145]	Tries to get the Type with the specified name out of the collection.

Reference

[ReadOnlyDataTypeCollection.T. Class \[▶ 3139\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.6.3.1 ReadOnlyDataTypeCollection.T..ContainsType Method

Determines whether the specified name contains type.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool ContainsType(
    string name
)
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: Boolean
true if the specified name contains type; otherwise, false.

Implements

[IDataTypeCollection.T..ContainsType\(String\) \[▶ 2489\]](#)

Reference[ReadOnlyDataTypeCollection.T. Class \[► 3139\]](#)[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)**6.12.6.3.2 ReadOnlyDataTypeCollection.T..TryGetType Method**

Tries to get the Type with the specified name out of the collection.

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public bool TryGetType(
    string name,
    out T? type
)
```

Parameters

name	Type: System.String The name.
type	Type: T [► 3139]. The type.

Return Value

Type: Boolean
true if found

Implements[IDataTypeCollection.T..TryGetType\(String, T.\) \[► 2490\]](#)**Reference**[ReadOnlyDataTypeCollection.T. Class \[► 3139\]](#)[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)**6.12.7 ReadOnlyInstanceCollection.T. Class**

ReadOnly Instance collection

Inheritance Hierarchy

System.Object

System.Collections.ObjectModel.ReadOnlyCollection.T.

TwinCAT.TypeSystem.Generic.ReadOnlyInstanceCollection.T.

TwinCAT.TypeSystem.Generic.ReadOnlySymbolCollection.T. [[► 3161](#)]TwinCAT.TypeSystem.ReadOnlyFieldCollection [[► 2841](#)]TwinCAT.TypeSystem.ReadOnlyMemberCollection [[► 2845](#)]**Namespace:** [TwinCAT.TypeSystem.Generic \[► 3086\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#


```
public class ReadOnlyInstanceCollection<T> : ReadOnlyCollection<T>,
    ICollection<T>, IList<T>, ICollection<T>, IEnumerable<T>,
    IEnumerable
where T : class, IInstance
```

Type Parameters






T

The `ReadOnlyInstanceCollection.T` type exposes the following members.










Constructors








	Name	Description
	<code>ReadOnlyInstanceCollection.T</code> [▶ 3147]	Initializes a new instance of the <code>ReadOnlyInstanceCollection.T</code> class.

Properties

	Name	Description
	<code>Count</code>	Gets the number of elements contained in the <code>ReadOnlyCollection.T</code> instance. (Inherited from <code>ReadOnlyCollection.T</code> .)
	<code>Item.Int32</code>	Gets the element at the specified index. (Inherited from <code>ReadOnlyCollection.T</code> .)
	<code>Item.String</code> [▶ 3148]	Gets the element with the specified instance path.
	<code>Items</code>	Returns the <code>IList.T</code> that the <code>ReadOnlyCollection.T</code> wraps. (Inherited from <code>ReadOnlyCollection.T</code> .)
	<code>Mode</code> [▶ 3149]	Gets the <code>InstanceCollectionMode</code> [▶ 2573].

Methods

	Name	Description
	<code>Contains(T)</code>	Determines whether an element is in the <code>ReadOnlyCollection.T</code> . (Inherited from <code>ReadOnlyCollection.T</code> .)
	<code>Contains(String)</code> [▶ 3150]	Determines whether the <code>ReadOnlyInstanceCollection.T</code> contains an instance with the specified instance path.
	<code>ContainsName</code> [▶ 3151]	Determines whether the specified instance is contained.
	<code>CopyTo</code>	Copies the entire <code>ReadOnlyCollection.T</code> to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from <code>ReadOnlyCollection.T</code> .)
	<code>Equals</code>	Determines whether the specified object is equal to the current object. (Inherited from <code>Object</code> .)
	<code>Finalize</code>	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from <code>Object</code> .)
	<code>GetEnumerator</code>	Returns an enumerator that iterates through the <code>ReadOnlyCollection.T</code> . (Inherited from <code>ReadOnlyCollection.T</code> .)
	<code>GetHashCode</code>	Serves as the default hash function. (Inherited from <code>Object</code> .)
	<code>GetInstance</code> [▶ 3152]	Gets the <code>IInstance</code> [▶ 2549] by instance path.

	Name	Description
	GetInstanceByName [▶ 3152]	Gets the IInstance [▶ 2549] by instance name.
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.T..)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryGetInstance [▶ 3153]	Tries to get the instance with the specified instance path.
	TryGetInstanceByName [▶ 3153]	Tries to get the instance by name.

Reference

[TwinCAT.TypeSystem.Generic Namespace](#) [\[▶ 3086\]](#)

6.12.7.1 [ReadOnlyInstanceCollection.T. Constructor](#)

Initializes a new instance of the [ReadOnlyInstanceCollection.T.](#) [\[▶ 3145\]](#) class.

Namespace: [TwinCAT.TypeSystem.Generic](#) [\[▶ 3086\]](#)

Assembly: [TwinCAT.Ads](#) (in [TwinCAT.Ads.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyInstanceCollection(
    IInstanceCollection<T> coll
)
```

Parameters

coll	Type: TwinCAT.TypeSystem.IInstanceCollection [▶ 2554].T [▶ 3145] . The coll.
------	---

Reference



[ReadOnlyInstanceCollection.T. Class](#) [\[▶ 3145\]](#)




[TwinCAT.TypeSystem.Generic Namespace](#) [\[▶ 3086\]](#)

6.12.7.2 [ReadOnlyInstanceCollection.T. Properties](#)

The [ReadOnlyInstanceCollection.T.](#) [\[▶ 3145\]](#) generic type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.T. [▶ 3145]..)
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.T. [▶ 3145]..)

	Name	Description
	Item.String. [▶ 3148]	Gets the element with the specified instance path.
	Items	Returns the <code>IList.T.</code> that the <code>ReadOnlyCollection.T.</code> wraps. (Inherited from ReadOnlyCollection.T [▶ 3145]..)
	Mode [▶ 3149]	Gets the InstanceCollectionMode [▶ 2573] .



Reference

[ReadOnlyInstanceCollection.T. Class \[▶ 3145\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.7.2.1 ReadOnlyInstanceCollection.T..Item Property

Overload List

	Name	Description
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.T [▶ 3145]..)
	Item.String. [▶ 3148]	Gets the element with the specified instance path.

Reference

[ReadOnlyInstanceCollection.T. Class \[▶ 3145\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.7.2.1.1 ReadOnlyInstanceCollection.T..Item Property (String)

Gets the element with the specified instance path.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T this[
    string instancePath
] { get; }
```

Parameters

instancePath	Type: System.String The instance path.
--------------	---

Return Value

Type: [T \[▶ 3145\]](#)

The instance if contained.

Implements

[IInstanceCollection.T..Item.String. \[▶ 2556\]](#)

Reference

[ReadOnlyInstanceCollection.T. Class \[▶ 3145\]](#)

[Item Overload \[▶ 3148\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.7.2 ReadOnlyInstanceCollection.T..Mode Property

Gets the [InstanceCollectionMode \[▶ 2573\]](#).

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public InstanceCollectionMode Mode { get; }
```

Property Value

Type: [InstanceCollectionMode \[▶ 2573\]](#)

The mode.

Implements

[IInstanceCollection.T..Mode \[▶ 2556\]](#)

Reference







[ReadOnlyInstanceCollection.T. Class \[▶ 3145\]](#)











[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.7.3 ReadOnlyInstanceCollection.T. Methods

The [ReadOnlyInstanceCollection.T. \[▶ 3145\]](#) generic type exposes the following members.

Methods

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.T [▶ 3145] ..)
	Contains(String) [▶ 3150]	Determines whether the ReadOnlyInstanceCollection.T. [▶ 3145] contains an instance with the specified instance path.
	ContainsName [▶ 3151]	Determines whether the specified instance is contained.
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.T [▶ 3145] ..)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)

	Name	Description
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.T [▶ 3145]..)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetInstance [▶ 3152]	Gets the Instance [▶ 2549] by instance path.
	GetInstanceByName [▶ 3152]	Gets the Instance [▶ 2549] by instance name.
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.T [▶ 3145]..)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInstance [▶ 3153]	Tries to get the instance with the specified instance path.
	TryGetInstanceByName [▶ 3153]	Tries to get the instance by name.



Reference

[ReadOnlyInstanceCollection.T. Class \[▶ 3145\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.7.3.1 ReadOnlyInstanceCollection.T..Contains Method

Overload List

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.T [▶ 3145]..)
	Contains(String) [▶ 3150]	Determines whether the ReadOnlyInstanceCollection.T. [▶ 3145] contains an instance with the specified instance path.

Reference

[ReadOnlyInstanceCollection.T. Class \[▶ 3145\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.7.3.1.1 ReadOnlyInstanceCollection.T..Contains Method (String)

Determines whether the [ReadOnlyInstanceCollection.T. \[▶ 3145\]](#) contains an instance with the specified instance path.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Contains(  
    string instancePath  
)
```

Parameters

instancePath	Type: System.String The instance path.
--------------	---

Return Value

Type: Boolean
true if contains the specified instance path; otherwise, false.

Implements

[IInstanceCollection.T..Contains\(String\) \[► 2558\]](#)

Reference

[ReadOnlyInstanceCollection.T. Class \[► 3145\]](#)

[Contains Overload \[► 3150\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.7.3.2 **ReadOnlyInstanceCollection.T..ContainsName Method**

Determines whether the specified instance is contained.

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool ContainsName(  
    string instanceName  
)
```

Parameters

instanceName	Type: System.String Name of the instance.
--------------	--

Return Value

Type: Boolean
true, if instance name is found.

Implements

[IInstanceCollection.T..ContainsName\(String\) \[► 2558\]](#)

Reference

[ReadOnlyInstanceCollection.T. Class \[► 3145\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.7.3.3 **ReadOnlyInstanceCollection.T..GetInstance Method**

Gets the [IInstance \[▶ 2549\]](#) by instance path.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T GetInstance(  
    string instancePath  
)
```

Parameters

instancePath	Type: System.String The instance path.
--------------	---

Return Value

Type: [T \[▶ 3145\]](#)

T.

Implements

[IInstanceCollection.T..GetInstance\(String\) \[▶ 2559\]](#)

Reference

[ReadOnlyInstanceCollection.T. Class \[▶ 3145\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.7.3.4 **ReadOnlyInstanceCollection.T..GetInstanceByName Method**

Gets the [IInstance \[▶ 2549\]](#) by instance name.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IList<T> GetInstanceByName(  
    string instanceName  
)
```

Parameters

instanceName	Type: System.String Name of the instance.
--------------	--

Return Value

Type: [IList.T \[▶ 3145\]](#).

IList<T>.

Implements

[IInstanceCollection.T..GetInstanceByName\(String\) \[► 2559\]](#)

Reference

[ReadOnlyInstanceCollection.T. Class \[► 3145\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.7.3.5 **ReadOnlyInstanceCollection.T..TryGetInstance Method**

Tries to get the instance with the specified instance path.

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetInstance(  
    string instancePath,  
    out T? instance  
)
```

Parameters

instancePath	Type: System.String The instance path.
instance	Type: T [► 3145]. The instance.

Return Value

Type: Boolean
true, if found, false if not contained.

Implements

[IInstanceCollection.T..TryGetInstance\(String, T.\) \[► 2560\]](#)

Reference

[ReadOnlyInstanceCollection.T. Class \[► 3145\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.7.3.6 **ReadOnlyInstanceCollection.T..TryGetInstanceByName Method**

Tries to get the instance by name.

Namespace: [TwinCAT.TypeSystem.Generic \[► 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetInstanceByName(
    string instanceName,
    out IList<T>? symbols
)
```

Parameters

instanceName	Type: System.String Name of the instance.
symbols	Type: System.Collections.Generic.IList.T [▶ 3145]. The found symbols (out-parameter)

Return Value

Type: Boolean
true, if found; false if not contained.

Implements

[IInstanceCollection.T..TryGetInstanceByName\(String, IList.T..\) \[▶ 2560\]](#)

Reference

[ReadOnlyInstanceCollection.T. Class \[▶ 3145\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.8 ReadOnlyNamespaceCollection.T. Class

Read Only namespace collection

Inheritance Hierarchy

System.Object

System.Collections.ObjectModel.ReadOnlyCollection.INamespace [▶ 3102].T..

TwinCAT.TypeSystem.Generic.ReadOnlyNamespaceCollection.T.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#


```
public class ReadOnlyNamespaceCollection<T> : ReadOnlyCollection<INamespace<T>>,
    INamespaceCollection<T>, ICollection<INamespace<T>>, IEnumerable<INamespace<T>>,
    IEnumerable
where T : class, IDataTypeInfo
```

Type Parameters






T

The `ReadOnlyNamespaceCollection.T.` type exposes the following members.















Constructors

	Name	Description
	<code>ReadOnlyNamespaceCollection.T.</code> [▶ 3156]	Initializes a new instance of the <code>ReadOnlyNamespaceCollection.T.</code> class.

Properties

	Name	Description
	<code>AllTypes</code> [▶ 3156]	Gets all types included in all namespaces.
	<code>Count</code>	Gets the number of elements contained in the <code>ReadOnlyCollection.T.</code> instance. (Inherited from <code>ReadOnlyCollection.INamespace</code> [▶ 3102].T...)
	<code>Item.Int32.</code>	Gets the element at the specified index. (Inherited from <code>ReadOnlyCollection.INamespace</code> [▶ 3102].T...)
	<code>Item.String.</code> [▶ 3157]	Gets the element at the specified index.
	<code>Items</code>	Returns the <code>IList.T.</code> that the <code>ReadOnlyCollection.T.</code> wraps. (Inherited from <code>ReadOnlyCollection.INamespace</code> [▶ 3102].T...)

Methods

	Name	Description
	<code>Contains</code>	Determines whether an element is in the <code>ReadOnlyCollection.T..</code> (Inherited from <code>ReadOnlyCollection.INamespace</code> [▶ 3102].T...)
	<code>ContainsNamespace</code> [▶ 3158]	Determines whether this collection contains a namespace with the specified name.
	<code>CopyTo</code>	Copies the entire <code>ReadOnlyCollection.T.</code> to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from <code>ReadOnlyCollection.INamespace</code> [▶ 3102].T...)
	<code>Equals</code>	Determines whether the specified object is equal to the current object. (Inherited from <code>Object.</code>)
	<code>Finalize</code>	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from <code>Object.</code>)
	<code>GetEnumerator</code>	Returns an enumerator that iterates through the <code>ReadOnlyCollection.T..</code> (Inherited from <code>ReadOnlyCollection.INamespace</code> [▶ 3102].T...)
	<code>GetHashCode</code>	Serves as the default hash function. (Inherited from <code>Object.</code>)
	<code>GetType</code>	Gets the Type of the current instance. (Inherited from <code>Object.</code>)
	<code>IndexOf</code>	Searches for the specified object and returns the zero-based index of the first occurrence within the entire <code>ReadOnlyCollection.T..</code> (Inherited from <code>ReadOnlyCollection.INamespace</code> [▶ 3102].T...)
	<code>MemberwiseClone</code>	Creates a shallow copy of the current <code>Object.</code> (Inherited from <code>Object.</code>)
	<code>ToString</code>	Returns a string that represents the current object. (Inherited from <code>Object.</code>)
	<code>TryGetNamespace</code> [▶ 3159]	Tries to get the namespace with the specified name.
	<code>TryGetType</code> [▶ 3160]	Tries to get the specified data type.
	<code>TryGetTypeByFullName</code> [▶ 3160]	Tries to get the specified type (by <code>fullName</code>)

Reference

[TwinCAT.TypeSystem.Generic Namespace](#) [▶ 3086]

6.12.8.1 ReadOnlyNamespaceCollection.T. Constructor

Initializes a new instance of the [ReadOnlyNamespaceCollection.T.](#) [▶ 3154] class.

Namespace: [TwinCAT.TypeSystem.Generic](#) [▶ 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlyNamespaceCollection(
    NamespaceCollection<T> coll
)
```

Parameters

coll	Type: TwinCAT.TypeSystem.Generic.NamespaceCollection [▶ 3126].T [▶ 3154]. The coll.
------	--

Reference






[ReadOnlyNamespaceCollection.T. Class](#) [▶ 3154]

[TwinCAT.TypeSystem.Generic Namespace](#) [▶ 3086]

6.12.8.2 ReadOnlyNamespaceCollection.T. Properties

The [ReadOnlyNamespaceCollection.T.](#) [▶ 3154] generic type exposes the following members.

Properties

	Name	Description
	AllTypes [▶ 3156]	Gets all types included in all namespaces.
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.INamespace [▶ 3102].T [▶ 3154]...)
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.INamespace [▶ 3102].T [▶ 3154]...)
	Item.String. [▶ 3157]	Gets the element at the specified index.
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.INamespace [▶ 3102].T [▶ 3154]...)

Reference

[ReadOnlyNamespaceCollection.T. Class](#) [▶ 3154]

[TwinCAT.TypeSystem.Generic Namespace](#) [▶ 3086]

6.12.8.2.1 ReadOnlyNamespaceCollection.T..AllTypes Property

Gets all types included in all namespaces.

Namespace: [TwinCAT.TypeSystem.Generic](#) [▶ 3086]
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IEnumerable<T> AllTypes { get; }
```

Property Value



Type: [IEnumerable](#) [▶ 2486].
 All types.

Reference

- [ReadOnlyNamespaceCollection.T. Class](#) [▶ 3154]
- [TwinCAT.TypeSystem.Generic Namespace](#) [▶ 3086]

6.12.8.2.2 ReadOnlyNamespaceCollection.T..Item Property

Overload List

	Name	Description
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.INamespace [▶ 3102]. T [▶ 3154]...)
	Item.String. [▶ 3157]	Gets the element at the specified index.

Reference

- [ReadOnlyNamespaceCollection.T. Class](#) [▶ 3154]
- [TwinCAT.TypeSystem.Generic Namespace](#) [▶ 3086]

6.12.8.2.2.1 ReadOnlyNamespaceCollection.T..Item Property (String)

Gets the element at the specified index.

Namespace: [TwinCAT.TypeSystem.Generic](#) [▶ 3086]
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public INamespace<T> this[
    string name
] { get; }
```

Parameters

name	Type: System.String The name.
------	----------------------------------

Return Value

Type: [INamespace](#) [▶ 3102].[T](#) [▶ 3154].

Reference

[ReadOnlyNamespaceCollection.T. Class \[► 3154\]](#)















[Item Overload \[► 3157\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.8.3 ReadOnlyNamespaceCollection.T. Methods

The [ReadOnlyNamespaceCollection.T. \[► 3154\]](#) generic type exposes the following members.

Methods

	Name	Description
	Contains	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.INamespace [► 3102].T [► 3154]...)
	ContainsNamespace [► 3158]	Determines whether this collection contains a namespace with the specified name.
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.INamespace [► 3102].T [► 3154]...)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.INamespace [► 3102].T [► 3154]...)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.INamespace [► 3102].T [► 3154]...)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetNamespace [► 3159]	Tries to get the namespace with the specified name.
	TryGetType [► 3160]	Tries to get the specified data type.
	TryGetTypeByFullName [► 3160]	Tries to get the specified type (by fullName)

Reference

[ReadOnlyNamespaceCollection.T. Class \[► 3154\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[► 3086\]](#)

6.12.8.3.1 ReadOnlyNamespaceCollection.T..ContainsNamespace Method

Determines whether this collection contains a namespace with the specified name.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool ContainsNamespace(
    string name
)
```

Parameters

name	Type: System.String The name of the namespace
------	--

Return Value

Type: Boolean
 true if the namespace is contained; otherwise, false.

Reference

- [ReadOnlyNamespaceCollection.T. Class](#) [[▶ 3154](#)]
- [TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.8.3.2 [ReadOnlyNamespaceCollection.T..TryGetNamespace Method](#)

Tries to get the namespace with the specified name.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetNamespace(
    string name,
    out INamespace<T>? nspace
)
```

Parameters

name	Type: System.String Namespace name.
nspace	Type: TwinCAT.TypeSystem.Generic.INamespace [▶ 3102]. T [▶ 3154]. The found namespace (out-parameter).

Return Value

Type: Boolean
 true if found, false if not contained.

Reference

- [ReadOnlyNamespaceCollection.T. Class](#) [[▶ 3154](#)]
- [TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.8.3.3 ReadOnlyNamespaceCollection.T..TryGetType Method

Tries to get the specified data type.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetType(  
    string typeName,  
    out T? dataType  
)
```

Parameters

typeName	Type: System.String Name of the type.
dataType	Type: T [▶ 3154]. Data Type (out-parameter).

Return Value

Type: Boolean

true if found, false if not contained.

Reference

[ReadOnlyNamespaceCollection.T. Class](#) [[▶ 3154](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.8.3.4 ReadOnlyNamespaceCollection.T..TryGetTypeByFullName Method

Tries to get the specified type (by fullName)

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetTypeByFullName(  
    string fullname,  
    out T? dataType  
)
```

Parameters

fullname	Type: System.String FullName of the data type.
dataType	Type: T [▶ 3154]. Found Data type (out-parameter).

Return Value

Type: Boolean

true if found, false if not contained.

Reference

[ReadOnlyNamespaceCollection.T. Class \[▶ 3154\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.9 ReadOnlySymbolCollection.T. Class

Read only symbol collection.

Inheritance Hierarchy

System.Object
 System.Collections.ObjectModel.ReadOnlyCollection.T.
[TwinCAT.TypeSystem.Generic.ReadOnlyInstanceCollection \[▶ 3145\].T.](#)
[TwinCAT.TypeSystem.Generic.ReadOnlySymbolCollection.T.](#)
[TwinCAT.TypeSystem.ReadOnlySymbolCollection \[▶ 2861\]](#)

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#


```
public class ReadOnlySymbolCollection<T> : ReadOnlyInstanceCollection<T>
where T : class, ISymbol
```

Type Parameters







T

The ReadOnlySymbolCollection.T. type exposes the following members.

















Constructors

	Name	Description
	ReadOnlySymbolCollection.T. [▶ 3162]	Initializes a new instance of the ReadOnlySymbolCollection.T. class.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.T..)
	Empty [▶ 3163]	Returns an empty collection.
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.T..)
	Item.String. [▶ 3148]	Gets the element with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.T..)
	Mode [▶ 3149]	Gets the InstanceCollectionMode [▶ 2573] . (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)

Methods

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.T..)
	<u>Contains(String)</u> [▶ 3150]	Determines whether the <u>ReadOnlyInstanceCollection.T.</u> [▶ 3145] contains an instance with the specified instance path. (Inherited from <u>ReadOnlyInstanceCollection.T.</u> [▶ 3145].)
	<u>ContainsName</u> [▶ 3151]	Determines whether the specified instance is contained. (Inherited from <u>ReadOnlyInstanceCollection.T.</u> [▶ 3145].)
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.T..)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.T..)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	<u>GetInstance</u> [▶ 3152]	Gets the <u>IInstance</u> [▶ 2549] by instance path. (Inherited from <u>ReadOnlyInstanceCollection.T.</u> [▶ 3145].)
	<u>GetInstanceByName</u> [▶ 3152]	Gets the <u>IInstance</u> [▶ 2549] by instance name. (Inherited from <u>ReadOnlyInstanceCollection.T.</u> [▶ 3145].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.T..)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	<u>TryGetInstance</u> [▶ 3153]	Tries to get the instance with the specified instance path. (Inherited from <u>ReadOnlyInstanceCollection.T.</u> [▶ 3145].)
	<u>TryGetInstanceByName</u> [▶ 3153]	Tries to get the instance by name. (Inherited from <u>ReadOnlyInstanceCollection.T.</u> [▶ 3145].)

Reference

TwinCAT.TypeSystem.Generic Namespace [▶ 3086]

6.12.9.1 ReadOnlySymbolCollection.T. Constructor

Initializes a new instance of the ReadOnlySymbolCollection.T. [▶ 3161] class.

Namespace: TwinCAT.TypeSystem.Generic [▶ 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlySymbolCollection(
    ICollection<T> coll
)
```

Parameters

coll	Type: TwinCAT.TypeSystem.IInstanceCollection [▶ 2554].T [▶ 3161]. The coll.
------	--

Reference







[ReadOnlySymbolCollection.T. Class](#) [[▶ 3161](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.9.2 ReadOnlySymbolCollection.T. Properties

The [ReadOnlySymbolCollection.T.](#) [[▶ 3161](#)] generic type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements contained in the ReadOnlyCollection.T. instance. (Inherited from ReadOnlyCollection.T [▶ 3161].)
	Empty [▶ 3163]	Returns an empty collection.
	Item.Int32.	Gets the element at the specified index. (Inherited from ReadOnlyCollection.T [▶ 3161].)
	Item.String. [▶ 3148]	Gets the element with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	Items	Returns the IList.T. that the ReadOnlyCollection.T. wraps. (Inherited from ReadOnlyCollection.T [▶ 3161].)
	Mode [▶ 3149]	Gets the InstanceCollectionMode [▶ 2573]. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)

Reference

[ReadOnlySymbolCollection.T. Class](#) [[▶ 3161](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.9.2.1 ReadOnlySymbolCollection.T..Empty Property

Returns an empty collection.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ReadOnlySymbolCollection<T> Empty { get; }
```

Return Value

Type: [ReadOnlySymbolCollection](#) [[▶ 3161](#)].T [[▶ 3161](#)].
 ReadOnlySymbolCollection<T>.

Reference

















[ReadOnlySymbolCollection.T. Class](#) [[▶ 3161](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.9.3 ReadOnlySymbolCollection.T. Methods

The [ReadOnlySymbolCollection.T.](#) [[▶ 3161](#)] generic type exposes the following members.

Methods

	Name	Description
	Contains(T)	Determines whether an element is in the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.T [▶ 3161]..)
	Contains(String) [▶ 3150]	Determines whether the ReadOnlyInstanceCollection.T. [▶ 3145] contains an instance with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	ContainsName [▶ 3151]	Determines whether the specified instance is contained. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	CopyTo	Copies the entire ReadOnlyCollection.T. to a compatible one-dimensional Array, starting at the specified index of the target array. (Inherited from ReadOnlyCollection.T [▶ 3161]..)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetEnumerator	Returns an enumerator that iterates through the ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.T [▶ 3161]..)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetInstance [▶ 3152]	Gets the IInstance [▶ 2549] by instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	GetInstanceByName [▶ 3152]	Gets the IInstance [▶ 2549] by instance name. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire ReadOnlyCollection.T.. (Inherited from ReadOnlyCollection.T [▶ 3161]..)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TryGetInstance [▶ 3153]	Tries to get the instance with the specified instance path. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)
	TryGetInstanceByName [▶ 3153]	Tries to get the instance by name. (Inherited from ReadOnlyInstanceCollection.T. [▶ 3145].)

Reference

[ReadOnlySymbolCollection.T. Class \[▸ 3161\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▸ 3086\]](#)

6.12.9.3.1 ReadOnlySymbolCollection.T..Empty Method

Returns an empty collection.

Namespace: [TwinCAT.TypeSystem.Generic \[▸ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public static ReadOnlySymbolCollection<T> Empty()
```

Return Value

Type: [ReadOnlySymbolCollection \[▸ 3161\].T \[▸ 3161\]](#).

[ReadOnlySymbolCollection<T>](#).

Reference

[ReadOnlySymbolCollection.T. Class \[▸ 3161\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▸ 3086\]](#)

6.12.10 SymbolCollection.T. Class

Interface represents a collection of [ISymbol \[▸ 2691\]](#) objects.

Inheritance Hierarchy

System.Object

[TwinCAT.TypeSystem.Generic.InstanceCollection \[▸ 3104\].T.](#)

[TwinCAT.TypeSystem.Generic.SymbolCollection.T.](#)

[TwinCAT.TypeSystem.SymbolCollection \[▸ 2928\]](#)

Namespace: [TwinCAT.TypeSystem.Generic \[▸ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#









```
public class SymbolCollection<T> : InstanceCollection<T>,
    ISymbolCollection<T>, IInstanceCollection<T>, IList<T>, ICollection<T>,
    IEnumerable<T>, IEnumerable
where T : class, ISymbol
```

Type Parameters















T












The [SymbolCollection.T.](#) type exposes the following members.

Properties

	Name	Description
	Count [▶ 3107]	Gets the collection count. (Inherited from InstanceCollection.T. [▶ 3104].)
	Empty [▶ 3168]	Creates an Empty SymbolCollection.T.
	InnerList [▶ 3108]	Gets the List of instances. (Inherited from InstanceCollection.T. [▶ 3104].)
	InnerPathDict [▶ 3108]	The Path dictionary (Inherited from InstanceCollection.T. [▶ 3104].)
	IsReadOnly [▶ 3109]	Gets a value indicating whether this instance is read only. (Inherited from InstanceCollection.T. [▶ 3104].)
	Item.Int32. [▶ 3109]	Gets or sets the Instance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	Item.String. [▶ 3110]	Gets the Instance [▶ 2549] with the specified instance path. (Inherited from InstanceCollection.T. [▶ 3104].)
	Mode [▶ 3111]	The mode this InstanceCollection.T. [▶ 3104] is working in. (Inherited from InstanceCollection.T. [▶ 3104].)

Methods

	Name	Description
	Add [▶ 3112]	Adds the specified item. (Inherited from InstanceCollection.T. [▶ 3104].)
	AddRange [▶ 3113]	Adds the specified items to this collection. (Inherited from InstanceCollection.T. [▶ 3104].)
	AsReadOnly [▶ 3169]	Returns a Read only version of this collection (shallow copy).
	Clear [▶ 3114]	Clears this instance. (Inherited from InstanceCollection.T. [▶ 3104].)
	Clone [▶ 3170]	Clones this instance.
	Contains(T) [▶ 3115]	Determines whether this collection contains the specified Instance [▶ 2549] (Inherited from InstanceCollection.T. [▶ 3104].)
	Contains(String) [▶ 3114]	Determines whether this collection contains an Instance [▶ 2549] with the specified InstanceName / InstancePath (Inherited from InstanceCollection.T. [▶ 3104].)
	ContainsName [▶ 3116]	Determines whether the specified instance name contains name. (Inherited from InstanceCollection.T. [▶ 3104].)
	CopyTo [▶ 3116]	Copies this InstanceCollection.T. [▶ 3104] to the specified array. (Inherited from InstanceCollection.T. [▶ 3104].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetEnumerator [▶ 3117]	Gets the enumerator. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetInstance [▶ 3117]	Gets the Instance [▶ 2549] by instance path. (Inherited from InstanceCollection.T. [▶ 3104].)

	Name	Description
	GetInstanceByName [▶ 3118]	Gets the name of the instance by. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 3119]	Determines the index of the specified IInstance [▶ 2549]. (Inherited from InstanceCollection.T. [▶ 3104].)
	Insert [▶ 3119]	Inserts the specified IInstance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 3120]	Removes the specified item. (Inherited from InstanceCollection.T. [▶ 3104].)
	RemoveAt [▶ 3121]	Removes the IInstance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInstance [▶ 3121]	Tries to get the IInstance [▶ 2549]. of the specified path. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetInstanceByName [▶ 3122]	Tries to get Instances by name. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetInstances [▶ 3170]	Try to get instances with predicate function









Reference

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.10.1 SymbolCollection.T. Properties

The [SymbolCollection.T.](#) [[▶ 3165](#)] generic type exposes the following members.

Properties

	Name	Description
	Count [▶ 3107]	Gets the collection count. (Inherited from InstanceCollection.T. [▶ 3104].)
	Empty [▶ 3168]	Creates an Empty SymbolCollection.T. [▶ 3165]
	InnerList [▶ 3108]	Gets the IList of instances. (Inherited from InstanceCollection.T. [▶ 3104].)
	InnerPathDict [▶ 3108]	The Path dictionary (Inherited from InstanceCollection.T. [▶ 3104].)
	IsReadOnly [▶ 3109]	Gets a value indicating whether this instance is read only. (Inherited from InstanceCollection.T. [▶ 3104].)
	Item.Int32. [▶ 3109]	Gets or sets the IInstance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	Item.String. [▶ 3110]	Gets the IInstance [▶ 2549] with the specified instance path. (Inherited from InstanceCollection.T. [▶ 3104].)
	Mode [▶ 3111]	The mode this InstanceCollection.T. [▶ 3104] is working in. (Inherited from InstanceCollection.T. [▶ 3104].)

Reference

[SymbolCollection.T. Class](#) [▶ 3165]

[TwinCAT.TypeSystem.Generic Namespace](#) [▶ 3086]

6.12.10.1.1 SymbolCollection.T..Empty Property

Creates an Empty [SymbolCollection.T.](#) [▶ 3165]

Namespace: [TwinCAT.TypeSystem.Generic](#) [▶ 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static SymbolCollection<T> Empty { get; }
```

Return Value

Type: [SymbolCollection](#) [▶ 3165].[T](#) [▶ 3165].
SymbolCollection<T>.

Reference











[SymbolCollection.T. Class](#) [▶ 3165]
















[TwinCAT.TypeSystem.Generic Namespace](#) [▶ 3086]

6.12.10.2 SymbolCollection.T. Methods

The [SymbolCollection.T.](#) [▶ 3165] generic type exposes the following members.

Methods

	Name	Description
	Add [▶ 3112]	Adds the specified item. (Inherited from InstanceCollection.T. [▶ 3104].)
	AddRange [▶ 3113]	Adds the specified items to this collection. (Inherited from InstanceCollection.T. [▶ 3104].)
	AsReadOnly [▶ 3169]	Returns a Read only version of this collection (shallow copy).
	Clear [▶ 3114]	Clears this instance. (Inherited from InstanceCollection.T. [▶ 3104].)
	Clone [▶ 3170]	Clones this instance.
	Contains(T) [▶ 3115]	Determines whether this collection contains the specified Instance [▶ 2549] (Inherited from InstanceCollection.T. [▶ 3104].)
	Contains(String) [▶ 3114]	Determines whether this collection contains an Instance [▶ 2549] with the specified InstanceName / InstancePath (Inherited from InstanceCollection.T. [▶ 3104].)
	ContainsName [▶ 3116]	Determines whether the specified instance name contains name. (Inherited from InstanceCollection.T. [▶ 3104].)
	CopyTo [▶ 3116]	Copies this InstanceCollection.T. [▶ 3104] to the specified array. (Inherited from InstanceCollection.T. [▶ 3104].)
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)

	Name	Description
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 3117]	Gets the enumerator. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetInstance [▶ 3117]	Gets the Instance [▶ 2549] by instance path. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetInstanceByName [▶ 3118]	Gets the name of the instance by. (Inherited from InstanceCollection.T. [▶ 3104].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf [▶ 3119]	Determines the index of the specified Instance [▶ 2549]. (Inherited from InstanceCollection.T. [▶ 3104].)
	Insert [▶ 3119]	Inserts the specified Instance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	Remove [▶ 3120]	Removes the specified item. (Inherited from InstanceCollection.T. [▶ 3104].)
	RemoveAt [▶ 3121]	Removes the Instance [▶ 2549] at the specified index. (Inherited from InstanceCollection.T. [▶ 3104].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)
	TryGetInstance [▶ 3121]	Tries to get the Instance [▶ 2549]. of the specified path. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetInstanceByName [▶ 3122]	Tries to get Instances by name. (Inherited from InstanceCollection.T. [▶ 3104].)
	TryGetInstances [▶ 3170]	Try to get instances with predicate function

Reference

[SymbolCollection.T. Class](#) [▶ 3165]

[TwinCAT.TypeSystem.Generic Namespace](#) [▶ 3086]

6.12.10.2.1 SymbolCollection.T..AsReadOnly Method

Returns a Read only version of this collection (shallow copy).

Namespace: [TwinCAT.TypeSystem.Generic](#) [▶ 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ReadOnlySymbolCollection<T> AsReadOnly()
```

Return Value

Type: [ReadOnlySymbolCollection](#) [▶ 3161].T [▶ 3165].
[ReadOnlySymbolCollection<T>](#).

Reference

[SymbolCollection.T. Class](#) [► 3165]

[TwinCAT.TypeSystem.Generic Namespace](#) [► 3086]

6.12.10.2.2 SymbolCollection.T..Clone Method

Clones this instance.

Namespace: [TwinCAT.TypeSystem.Generic](#) [► 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolCollection<T> Clone()
```

Return Value

Type: [SymbolCollection](#) [► 3165].[T](#) [► 3165].

SymbolCollection<T>.

Reference

[SymbolCollection.T. Class](#) [► 3165]

[TwinCAT.TypeSystem.Generic Namespace](#) [► 3086]

6.12.10.2.3 SymbolCollection.T..Empty Method

Creates an Empty [SymbolCollection.T.](#) [► 3165]

Namespace: [TwinCAT.TypeSystem.Generic](#) [► 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version:

5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
public static SymbolCollection<T> Empty()
```

Return Value

Type: [SymbolCollection](#) [► 3165].[T](#) [► 3165].

SymbolCollection<T>.

Reference

[SymbolCollection.T. Class](#) [► 3165]

[TwinCAT.TypeSystem.Generic Namespace](#) [► 3086]

6.12.10.2.4 SymbolCollection.T..TryGetInstances Method

Try to get instances with predicate function

Namespace: [TwinCAT.TypeSystem.Generic](#) [► 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool TryGetInstances(
    Func<T, bool> predicate,
    bool recurse,
    out IList<T>? instances
)
```

Parameters

predicate	Type: System.Func.T [▶ 3165], Boolean. The predicate function
recurse	Type: System.Boolean if set to true the symbol hierarchy will be searched recursively.
instances	Type: System.Collections.Generic.IList.T [▶ 3165]. The instances.

Return Value

Type: Boolean
true if the Instance(s) is/are found, false otherwise.

Reference

[SymbolCollection.T. Class \[▶ 3165\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.11 SymbolIterationMask Enumeration

Mask Flagset to specify filters for [SymbolIterator.T. \[▶ 3172\]](#).

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.95+cf8c3f5

Syntax

C#

```
[FlagsAttribute]
public enum SymbolIterationMask
```

Members

	Member name	Value	Description
	None	0	Uninitialized / None
	Structures	1	Iterates over Subelements of Structs
	Arrays	2	Iterates over Elements of Arrays
	Unions	4	Iterates over Subelements of Unions
	Pointer	8	Iterates over Pointer SubElements
	References	16	Iterates over References
	All	31	Iterates over All Complex/Combined types

Reference

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.12 SymbolIterator.T. Class

Iterator class for enumerations of [Symbols](#) [[▶ 2691](#)].

Inheritance Hierarchy

System.Object

TwinCAT.TypeSystem.Generic.SymbolIterator.T.

[TwinCAT.Ads.TypeSystem.SymbolIterator](#) [[▶ 1916](#)]

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#






```
public class SymbolIterator<T> : IEnumerable<T>,
    IEnumerable
where T : class, ISymbol
```

Type Parameters


T	Concrete ISymbol [▶ 2691] type.
---	---


The SymbolIterator.T. type exposes the following members.

Constructors








	Name	Description
	SymbolIterator.T. (ICollection . T.) [▶ 3174]	Initializes a new instance of the SymbolIterator.T. class.
	SymbolIterator.T. (IEnumerable . T. , Boolean) [▶ 3175]	Initializes a new instance of the SymbolIterator.T. class.
	SymbolIterator.T. (ICollection . T. , Func . T. , Boolean .) [▶ 3175]	Initializes a new instance of the SymbolIterator.T. class.
	SymbolIterator.T. (IEnumerable . T. , Boolean , Func . T. , Boolean .) [▶ 3176]	Initializes a new instance of the SymbolIterator.T. class.
	SymbolIterator.T. (IEnumerable . T. , Boolean , SymbolIterationMas k , Func . T. , Boolean ., Func . T. , Boolean .) [▶ 3177]	Initializes a new instance of the SymbolIterator.T. class.

Properties

	Name	Description
	Mask [▶ 3178]	Gets or sets the SymbolIterationMask [▶ 3019]

	Name	Description
	SymbolRecursionDetection [▶ 3178]	Gets or sets a value indicating whether the iterator checks for Symbol recursions (true by default).

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator [▶ 3179]	Gets the enumerator that enumerates through a collection
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Remarks

This iterator class can be used to iterate over collections of symbol trees (root symbols + sub symbols). By constructor the user can choose if the iterator works recursively within the symbol tree and optionally a filter function to select only specific symbols (predicate).

Examples

The following example shows how to determine, browse and filter symbols.

Browsing and filtering Symbols

```
using (AdsClient client = new AdsClient())
{
    CancellationToken cancel = CancellationToken.None;

    uint valueToRead = 0;
    uint valueToWrite = 42;

    client.Connect(AmsNetId.Local, 851);

    // Load all Symbols + DataTypes
    ISymbolLoader loader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.Default);

    ResultSymbols resultSymbols = await loader.GetSymbolsAsync(cancel);

    if (resultSymbols.Succeeded)
    {
        Symbol symbol = (Symbol)resultSymbols.Symbols["MAIN.nCounter"];

        // Works for ALL Primitive 'ANY TYPES' Symbols
        ResultWriteAccess resultWrite = await symbol.WriteValueAsync(valueToWrite, cancel);
        ResultReadValueAccess resultRead = await symbol.ReadValueAsync(cancel);

        if (resultRead.Succeeded)
            valueToRead = (uint)resultRead.Value;

        // Simple filtering of Symbols
        Regex filterExpression = new Regex(pattern: @"^MAIN.*"); // Everything that starts with "MAIN"

        // FilterFunction that filters for the InstancePath
        Func<ISymbol, bool> filter = s => filterExpression.IsMatch(s.InstancePath);
        SymbolIterator iterator = new SymbolIterator(symbols: resultSymbols.Symbols, recurse: true, selector: filter);
    }
}
```

```

foreach (ISymbol filteredSymbol in iterator)
{
    Console.WriteLine(filteredSymbol.InstancePath);
}
}

```






Reference

[TwinCAT.TypeSystem.Generic Namespace](#) [► 3086]

[System.Collections.Generic.IEnumerable.T.](#)

6.12.12.1 SymbolIterator.T. Constructor

Overload List

	Name	Description
	SymbolIterator.T. (ICollection.T.) [► 3174]	Initializes a new instance of the SymbolIterator.T. [► 3172] class.
	SymbolIterator.T. (IEnumerable.T., Boolean) [► 3175]	Initializes a new instance of the SymbolIterator.T. [► 3172] class.
	SymbolIterator.T. (ICollection.T., Func.T, Boolean.) [► 3175]	Initializes a new instance of the SymbolIterator.T. [► 3172] class.
	SymbolIterator.T. (IEnumerable.T., Boolean, Func.T, Boolean.) [► 3176]	Initializes a new instance of the SymbolIterator.T. [► 3172] class.
	SymbolIterator.T. (IEnumerable.T., Boolean, SymbolIterationMask, Func.T, Boolean., Func.T, Boolean.) [► 3177]	Initializes a new instance of the SymbolIterator.T. [► 3172] class.

Reference

[SymbolIterator.T. Class](#) [► 3172]

[TwinCAT.TypeSystem.Generic Namespace](#) [► 3086]

6.12.12.1.1 SymbolIterator.T. Constructor (ICollection.T.)

Initializes a new instance of the [SymbolIterator.T.](#) [► 3172] class.

Namespace: [TwinCAT.TypeSystem.Generic](#) [► 3086]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolIterator(
    ICollection<T> symbols
)
```

Parameters

symbols	Type: TwinCAT.TypeSystem.InstanceCollection [▶ 2554]. T [▶ 3172]. The root symbols.
---------	--

Reference

[SymbolIterator.T. Class](#) [[▶ 3172](#)]

[SymbolIterator.T. Overload](#) [[▶ 3174](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.12.1.2 SymbolIterator.T. Constructor (IEnumerable.T., Boolean)

Initializes a new instance of the [SymbolIterator.T.](#) [[▶ 3172](#)] class.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolIterator(
    IEnumerable<T> symbols,
    bool recurse
)
```

Parameters

symbols	Type: System.Collections.Generic.IEnumerable.T [▶ 3172]. The root collection
recurse	Type: System.Boolean if set to true, the iterator works recursively over all subsymbols.

Reference

[SymbolIterator.T. Class](#) [[▶ 3172](#)]

[SymbolIterator.T. Overload](#) [[▶ 3174](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.12.1.3 SymbolIterator.T. Constructor (InstanceCollection.T., Func.T, Boolean.)

Initializes a new instance of the [SymbolIterator.T.](#) [[▶ 3172](#)] class.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolIterator(
    IEnumerable<T> symbols,
    Func<T, bool>? selector
)
```

Parameters

symbols	Type: TwinCAT.TypeSystem.IInstanceCollection [▶ 2554]. T [▶ 3172]. The root symbols.
selector	Type: System.Func.T [▶ 3172], Boolean. Predicate function indicating that symbols are visible for the iteration. The default value null returns all symbols (of the specified mask).

Exceptions

Exception	Condition
NotSupportedException	

Reference

[SymbolIterator.T. Class](#) [[▶ 3172](#)]

[SymbolIterator.T. Overload](#) [[▶ 3174](#)]

[TwinCAT.TypeSystem.Generic Namespace](#) [[▶ 3086](#)]

6.12.12.1.4 SymbolIterator.T. Constructor (IEnumerable.T., Boolean, Func.T, Boolean.)

Initializes a new instance of the [SymbolIterator.T.](#) [[▶ 3172](#)] class.

Namespace: [TwinCAT.TypeSystem.Generic](#) [[▶ 3086](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolIterator(
    IEnumerable<T> symbols,
    bool recurse,
    Func<T, bool>? selector
)
```

Parameters

symbols	Type: System.Collections.Generic.IEnumerable.T [▶ 3172]. Input collection (root objects).
recurse	Type: System.Boolean if set to true, the iterator works recursively over all subsymbols.
selector	Type: System.Func.T [▶ 3172], Boolean. Predicate function indicating that symbols are visible for the iteration. The default value null returns all symbols (of the specified mask).

Reference

[SymbolIterator.T. Class](#) [[▶ 3172](#)]

[SymbolIterator.T. Overload \[▶ 3174\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.12.1.5 SymbolIterator.T. Constructor (IEnumerable.T., Boolean, SymbolIterationMask, Func.T, Boolean., Func.T, Boolean.)

Initializes a new instance of the [SymbolIterator.T. \[▶ 3172\]](#) class.

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolIterator(
    IEnumerable<T> symbols,
    bool recurse,
    SymbolIterationMask mask,
    Func<T, bool>? selector,
    Func<T, bool>? areChildsIterated
)
```

Parameters

symbols	Type: System.Collections.Generic.IEnumerable.T [▶ 3172] . Input collection (root objects).
recurse	Type: System.Boolean if set to true, the iterator works recursively over all subsymbols.
mask	Type: TwinCAT.TypeSystem.SymbolIterationMask [▶ 3019] Specifies a mask over the symbols, that filters out specific symbol categories. The default is All [▶ 3019] and all symbols are shown.
selector	Type: System.Func.T [▶ 3172] , Boolean . Predicate function indicating that symbols are visible for the iteration. The default value null returns all symbols (of the specified mask).
areChildsIterated	Type: System.Func.T [▶ 3172] , Boolean . Predicate function indicating that child of the actual symbol should be iterated (in recurse mode). The default value iterates all child (of the specified mask).

Reference

[SymbolIterator.T. Class \[▶ 3172\]](#)



[SymbolIterator.T. Overload \[▶ 3174\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.12.2 SymbolIterator.T. Properties

The [SymbolIterator.T. \[▶ 3172\]](#) generic type exposes the following members.

Properties

	Name	Description
	Mask [▶ 3178]	Gets or sets the SymbolIterationMask [▶ 3019]
	SymbolRecursionDetection [▶ 3178]	Gets or sets a value indicating whether the iterator checks for Symbol recursions (true by default).

Reference

[SymbolIterator.T. Class \[▶ 3172\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.12.2.1 SymbolIterator.T..Mask Property

Gets or sets the [SymbolIterationMask \[▶ 3019\]](#)

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public SymbolIterationMask Mask { get; set; }
```

Property Value

Type: [SymbolIterationMask \[▶ 3019\]](#)

The mask.

Remarks

This property can be used for prefiltering the iterator without using a predicate function.

Reference

[SymbolIterator.T. Class \[▶ 3172\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.12.2.2 SymbolIterator.T..SymbolRecursionDetection Property

Gets or sets a value indicating whether the iterator checks for Symbol recursions (true by default).

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool SymbolRecursionDetection { get; set; }
```

Property Value

Type: Boolean

true if recursion checking, false switched off check.

Reference








[SymbolIterator.T. Class \[▶ 3172\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.12.3 SymbolIterator.T. Methods

The [SymbolIterator.T. \[▶ 3172\]](#) generic type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	<u>GetEnumerator</u> [▶ 3179]	Gets the enumerator that enumerates through a collection
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[SymbolIterator.T. Class \[▶ 3172\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)

6.12.12.3.1 SymbolIterator.T..GetEnumerator Method

Gets the enumerator that enumerates through a collection

Namespace: [TwinCAT.TypeSystem.Generic \[▶ 3086\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IEnumerator<T> GetEnumerator()
```

Return Value

Type: [IEnumerator.T \[▶ 3172\]](#).

A [IEnumerator.T](#). that can be used to iterate through the collection.

Implements

[IEnumerable.T..GetEnumerator](#).

Reference











[SymbolIterator.T. Class \[▶ 3172\]](#)

[TwinCAT.TypeSystem.Generic Namespace \[▶ 3086\]](#)



6.13 TwinCAT.ValueAccess Namespace

Namespace for the common (non ADS dependent) value access.





Classes

	Class	Description
	CannotAccessValueException [▶ 3181]	Class CannotAccessValueException. This class cannot be inherited. Implements the SymbolException [▶ 2933]
	ResultAccess [▶ 3197]	Result class for an asynchronous access operation.
	ResultHandleAccess [▶ 3224]	Asynchronous read access result returning an variable handle. Implements the ResultAccess [▶ 3197]
	ResultReadDynamicValueAccess [▶ 3204]	Asynchronous read access result returning an Dynamic value object (IDynamicValue [▶ 2504]. Implements the ResultReadValueAccess.T. [▶ 3211]
	ResultReadRawAccess [▶ 3206]	Asynchronous read access result object, reading raw (byte[]) data into memory locations. Implements the ResultReadValueAccess.T. [▶ 3211]
	ResultReadValueAccess [▶ 3209]	Asynchronous read access result returning an untyped Value object type 'object'. Implements the ResultAccess [▶ 3197]
	ResultReadValueAccess.T. [▶ 3211]	Result object of an asynchronous read of a specific value of type T. Implements the ResultAccess [▶ 3197]
	ResultReadValueAccess2.I, V. [▶ 3226]	Result object of an asynchronous read. Implements the ResultAccess [▶ 3197]
	ResultRpcMethodAccess [▶ 3213]	Asynchronous Invoke RPC Method result class. Implements the ResultReadValueAccess [▶ 3209]
	ResultWriteAccess [▶ 3217]	Asynchronous write access result class.

Interfaces

	Interface	Description
	IAccessorRawValue [▶ 3184]	Helper Interface to access Symbol Values as byte Arrays
	IAccessorValueFactory [▶ 3192]	Factory interfaces for Accessor implementations.

Enumerations

	Enumeration	Description
	SymbolNotificationTypes [▶ 3222]	Specifies the Notification type of ADS Notifications
	ValueAccessMode [▶ 3228]	Enum ValueAccessMethod
	ValueCreationModes [▶ 3223]	Creation mode for Values
	ValueUpdateMode [▶ 3229]	Value Update Mode.

6.13.1 CannotAccessValueException Class

Class CannotAccessValueException. This class cannot be inherited. Implements the [SymbolException](#) [[▶ 2933](#)]

Inheritance Hierarchy

System.Object
 System.Exception
 [TwinCAT.AdsException](#) [[▶ 61](#)]
 [TwinCAT.TypeSystem.SymbolException](#) [[▶ 2933](#)]
 TwinCAT.ValueAccess.CannotAccessValueException

Namespace: [TwinCAT.ValueAccess](#) [[▶ 3179](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229



Syntax

C#











```
[SerializableAttribute]
public sealed class CannotAccessValueException : SymbolException
```

The CannotAccessValueException type exposes the following members.







Constructors

	Name	Description
	CannotAccessValueException . [▶ 3182]	Initializes a new instance of the CannotAccessValueException class.
	CannotAccessValueException(ISymbol) [▶ 3183]	Initializes a new instance of the CannotAccessValueException class.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	InstancePath [▶ 2944]	Gets the instance path. (Inherited from SymbolException [▶ 2933].)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	Symbol [▶ 2944]	Gets the symbol. (Inherited from SymbolException [▶ 2933].)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Methods



	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData [▶ 2945]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from SymbolException [▶ 2933] .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Reference

[TwinCAT.ValueAccess Namespace](#) [\[▶ 3179\]](#)

[TwinCAT.TypeSystem.SymbolException](#) [\[▶ 2933\]](#)

6.13.1.1 CannotAccessValueException Constructor**Overload List**

	Name	Description
	CannotAccessValueException . [▶ 3182]	Initializes a new instance of the CannotAccessValueException [▶ 3181] class.
	CannotAccessValueException(ISymbol) [▶ 3183]	Initializes a new instance of the CannotAccessValueException [▶ 3181] class.

Reference

[CannotAccessValueException Class](#) [\[▶ 3181\]](#)

[TwinCAT.ValueAccess Namespace](#) [\[▶ 3179\]](#)

6.13.1.1.1 CannotAccessValueException Constructor

Initializes a new instance of the [CannotAccessValueException](#) [\[▶ 3181\]](#) class.

Namespace: [TwinCAT.ValueAccess](#) [\[▶ 3179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public CannotAccessValueException()
```

Reference

[CannotAccessValueException Class](#) [\[▶ 3181\]](#)

[CannotAccessValueException Overload \[▶ 3182\]](#)

[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

6.13.1.1.2 CannotAccessValueException Constructor (ISymbol)

Initializes a new instance of the [CannotAccessValueException \[▶ 3181\]](#) class.

Namespace: [TwinCAT.ValueAccess \[▶ 3179\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public CannotAccessValueException(
    ISymbol symbol
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol.
--------	--

Reference

[CannotAccessValueException Class \[▶ 3181\]](#)











[CannotAccessValueException Overload \[▶ 3182\]](#)

[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

6.13.1.2 CannotAccessValueException Properties

The [CannotAccessValueException \[▶ 3181\]](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	InstancePath [▶ 2944]	Gets the instance path. (Inherited from SymbolException [▶ 2933] .)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	Symbol [▶ 2944]	Gets the symbol. (Inherited from SymbolException [▶ 2933] .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Reference







[CannotAccessValueException Class \[▶ 3181\]](#)

[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

6.13.1.3 CannotAccessValueException Methods

The [CannotAccessValueException \[▶ 3181\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData [▶ 2945]	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from SymbolException [▶ 2933] .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Reference

[CannotAccessValueException Class \[▶ 3181\]](#)

[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

6.13.2 IAccessorRawValue Interface

Helper Interface to access Symbol Values as byte Arrays

Namespace: [TwinCAT.ValueAccess \[▶ 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax

C#









```
public interface IAccessorRawValue
```

The IAccessorRawValue type exposes the following members.

Properties

	Name	Description
	DefaultValueEncoding [▶ 3185]	Gets the default value encoding.
	ValueFactory [▶ 3186]	Gets the value factory

Methods

	Name	Description
	ReadArrayElementRawAsync [▶ 3187]	Reads the array element value as bytes asynchronously.
	ReadRawAsync [▶ 3188]	Read a Symbol value asynchronously as bytes .
	TryReadArrayElementRaw [▶ 3188]	Reads an array element value as bytes.
	TryReadRaw [▶ 3189]	Read a Symbol value as bytes.
	TryWriteArrayElementRaw [▶ 3189]	Writes an array element value from raw memory asynchronously to the ADS Device.
	TryWriteRaw [▶ 3190]	Writes the symbol value from source memory location to the ADS Device.
	WriteArrayElementRawAsync [▶ 3191]	Writes an array element value from raw memory asynchronously to the ADS Device.
	WriteRawAsync [▶ 3191]	Writes the symbol value asynchronously from source memory location to the ADS Device.



Reference

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.2.1 IAccessorRawValue Properties

The [IAccessorRawValue](#) [[▶ 3184](#)] type exposes the following members.

Properties

	Name	Description
	DefaultValueEncoding [▶ 3185]	Gets the default value encoding.
	ValueFactory [▶ 3186]	Gets the value factory

Reference

[IAccessorRawValue Interface](#) [[▶ 3184](#)]

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.2.1.1 IAccessorRawValue.DefaultValueEncoding Property

Gets the default value encoding.

Namespace: [TwinCAT.ValueAccess](#) [[▶ 3179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Encoding DefaultValueEncoding { get; }
```

Property Value

Type: Encoding

The default value encoding.

Reference

[IAccessorRawValue Interface \[▸ 3184\]](#)

[TwinCAT.ValueAccess Namespace \[▸ 3179\]](#)

6.13.2.1.2 IAccessorRawValue.ValueFactory Property

Gets the value factory

Namespace: [TwinCAT.ValueAccess \[▸ 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IAccessorValueFactory ValueFactory { get; }
```

Property Value

Type: [IAccessorValueFactory \[▸ 3192\]](#)

The value factory or null if only Raw Values only.

Reference





[IAccessorRawValue Interface \[▸ 3184\]](#)





[TwinCAT.ValueAccess Namespace \[▸ 3179\]](#)

6.13.2.2 IAccessorRawValue Methods

The [IAccessorRawValue \[▸ 3184\]](#) type exposes the following members.

Methods

	Name	Description
	ReadArrayElementRawAsync [▸ 3187]	Reads the array element value as bytes asynchronously.
	ReadRawAsync [▸ 3188]	Read a Symbol value asynchronously as bytes .
	TryReadArrayElementRaw [▸ 3188]	Reads an array element value as bytes.
	TryReadRaw [▸ 3189]	Read a Symbol value as bytes.

	Name	Description
	TryWriteArrayElementRaw [▶ 3189]	Writes an array element value from raw memory asynchronously to the ADS Device.
	TryWriteRaw [▶ 3190]	Writes the symbol value from source memory location to the ADS Device.
	WriteArrayElementRawAsync [▶ 3191]	Writes an array element value from raw memory asynchronously to the ADS Device.
	WriteRawAsync [▶ 3191]	Writes the symbol value asynchronously from source memory location to the ADS Device.

Reference

[IAccessorRawValue Interface](#) [[▶ 3184](#)]

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.2.2.1 IAccessorRawValue.ReadArrayElementRawAsync Method

Reads the array element value as bytes asynchronously.

Namespace: [TwinCAT.ValueAccess](#) [[▶ 3179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultReadRawAccess> ReadArrayElementRawAsync(
    IArrayInstance arrayInstance,
    int[] indices,
    Memory<byte> readBuffer,
    CancellationToken cancel
)
```

Parameters

arrayInstance	Type: TwinCAT.TypeSystem.IArrayInstance [▶ 2453] The array instance.
indices	Type: <code>.System.Int32</code> . The indices, which specify the element to read.
readBuffer	Type: <code>System.Memory.Byte</code> . Read buffer where to read the data
cancel	Type: <code>System.Threading.CancellationToken</code> The cancellation token.

Return Value

Type: [Task.ResultReadRawAccess](#) [[▶ 3206](#)].

A task that represents the asynchronous 'ReadArrayElementRaw' operation. The [ResultReadRawAccess](#) [[▶ 3206](#)] result contains the ([Value](#) [[▶ 3213](#)]) and the [ErrorCode](#) [[▶ 3202](#)] after execution.

Reference

[IAccessorRawValue Interface](#) [[▶ 3184](#)]

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.2.2 IAccessorRawValue.ReadRawAsync Method

Read a Symbol value asynchronously as bytes .

Namespace: [TwinCAT.ValueAccess](#) [[▶ 3179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultReadRawAccess> ReadRawAsync (
    ISymbol symbolInstance,
    Memory<byte> readBuffer,
    CancellationToken cancel
)
```

Parameters

symbolInstance	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol instance.
readBuffer	Type: System.Memory.Byte. Read buffer where to read the value.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultReadRawAccess](#) [[▶ 3206](#)].

A task that represents the asynchronous 'ReadRaw' operation. The [ResultReadRawAccess](#) [[▶ 3206](#)] result contains the ([Value](#) [[▶ 3213](#)]) and the [ErrorCode](#) [[▶ 3202](#)] after execution.

Reference

[IAccessorRawValue Interface](#) [[▶ 3184](#)]

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.2.3 IAccessorRawValue.TryReadArrayElementRaw Method

Reads an array element value as bytes.

Namespace: [TwinCAT.ValueAccess](#) [[▶ 3179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int TryReadArrayElementRaw (
    IArrayInstance arrayInstance,
    int[] indices,
    Memory<byte> readBuffer,
    out DateTimeOffset? timeStamp
)
```

Parameters

arrayInstance	Type: TwinCAT.TypeSystem.IArrayInstance [▶ 2453] The array instance.
indices	Type: .System.Int32. The indices specify which element to read.

readBuffer	Type: System.Memory.Byte. ReadBuffer where to read the value.
timeStamp	Type: System.Nullable.DateTimeOffset.. The readtime snapshot

Return Value

Type: Int32
Error code. 0 represents succeed.

Reference

- [IAccessorRawValue Interface \[▶ 3184\]](#)
- [TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

6.13.2.2.4 IAccessorRawValue.TryReadRaw Method

Read a Symbol value as bytes.

Namespace: [TwinCAT.ValueAccess \[▶ 3179\]](#)
Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int TryReadRaw(
    ISymbol symbolInstance,
    Memory<byte> readBuffer,
    out DateTimeOffset? timeStamp
)
```

Parameters

symbolInstance	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol instance.
readBuffer	Type: System.Memory.Byte. Read buffer where to read the value.
timeStamp	Type: System.Nullable.DateTimeOffset.. The readtime snapshot

Return Value

Type: Int32
Error code. 0 represents succeed.

Reference

- [IAccessorRawValue Interface \[▶ 3184\]](#)
- [TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

6.13.2.2.5 IAccessorRawValue.TryWriteArrayElementRaw Method

Writes an array element value from raw memory asynchronously to the ADS Device.

Namespace: [TwinCAT.ValueAccess \[▶ 3179\]](#)
Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int TryWriteArrayElementRaw(
    IArrayInstance arrayInstance,
    int[] indices,
    ReadOnlyMemory<byte> writeBuffer,
    out DateTimeOffset? timeStamp
)
```

Parameters

arrayInstance	Type: TwinCAT.TypeSystem.IArrayInstance [► 2453] The array instance.
indices	Type: .System.Int32 . The indices of the array element.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer containing the data to write.
timeStamp	Type: System.Nullable.DateTimeOffset.. Write time / timestamp

Return Value

Type: [Int32](#)
Error code. 0 represents succeed.

Reference

[IAccessorRawValue Interface](#) [► 3184]

[TwinCAT.ValueAccess Namespace](#) [► 3179]

6.13.2.2.6 IAccessorRawValue.TryWriteRaw Method

Writes the symbol value from source memory location to the ADS Device.

Namespace: [TwinCAT.ValueAccess](#) [► 3179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
int TryWriteRaw(
    ISymbol symbolInstance,
    ReadOnlyMemory<byte> writeBuffer,
    out DateTimeOffset? timeStamp
)
```

Parameters

symbolInstance	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol instance.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer containing the data to write.
timeStamp	Type: System.Nullable.DateTimeOffset.. The write timestamp.

Return Value

Type: [Int32](#)
Error code. 0 represents succeed.

Reference

[IAccessorRawValue Interface \[▶ 3184\]](#)

[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

6.13.2.2.7 IAccessorRawValue.WriteArrayElementRawAsync Method

Writes an array element value from raw memory asynchronously to the ADS Device.

Namespace: [TwinCAT.ValueAccess \[▶ 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultWriteAccess> WriteArrayElementRawAsync (
    IArrayInstance arrayInstance,
    int[] indices,
    ReadOnlyMemory<byte> writeBuffer,
    CancellationToken cancel
)
```

Parameters

arrayInstance	Type: TwinCAT.TypeSystem.IArrayInstance [▶ 2453] The array instance.
indices	Type: .System.Int32 . The indices of the array element.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer containing the data to write (element value).
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWriteAccess \[▶ 3217\]](#).

A task that represents the asynchronous 'WriteRaw' operation. The [ResultWriteAccess \[▶ 3217\]](#) result contains the the [ErrorCode \[▶ 3202\]](#) after execution.

Reference

[IAccessorRawValue Interface \[▶ 3184\]](#)

[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

6.13.2.2.8 IAccessorRawValue.WriteRawAsync Method

Writes the symbol value asynchronously from source memory location to the ADS Device.

Namespace: [TwinCAT.ValueAccess \[▶ 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Task<ResultWriteAccess> WriteRawAsync (
    ISymbol symbolInstance,
    ReadOnlyMemory<byte> writeBuffer,
    CancellationToken cancel
)
```

Parameters

symbolInstance	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol instance.
writeBuffer	Type: System.ReadOnlyMemory.Byte . The write buffer containing the data to write.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [Task.ResultWriteAccess](#) [▶ 3217].

A task that represents the asynchronous 'WriteRaw' operation. The [ResultWriteAccess](#) [▶ 3217] result contains the the [ErrorCode](#) [▶ 3202] after execution.

Reference

[IAccessorRawValue Interface](#) [▶ 3184]

[TwinCAT.ValueAccess Namespace](#) [▶ 3179]

6.13.3 IAccessorValueFactory Interface

Factory interfaces for Accessor implementations.

Namespace: [TwinCAT.ValueAccess](#) [▶ 3179]



Assembly: [TwinCAT.Ads.Abstractions](#) (in [TwinCAT.Ads.Abstractions.dll](#)) Version: 6.0.328+39e3229

Syntax**C#**

```
public interface IAccessorValueFactory
```

The [IAccessorValueFactory](#) type exposes the following members.

Methods

	Name	Description
	CreatePrimitiveValue [▶ 3193]	Creates a primitive value, independent of any settings.
	CreateValue [▶ 3194]	Creates the symbols value from raw memory data.



Remarks

This interface is used by a custom Accessor class to create Value objects from memory representations.

Reference

[TwinCAT.ValueAccess Namespace](#) [▶ 3179]



Also see about this

-  [IAccessorValueFactory.CreateValue Method \(ISymbol, ReadOnlySpan.Byte., DateTimeOffset\)](#) [▶ 3196]
-  [IAccessorValueFactory.CreateValue Method \(ISymbol, ReadOnlySpan.Byte., IValue, DateTimeOffset\)](#) [▶ 3196]

6.13.3.1 IAccessorValueFactory Methods

The [IAccessorValueFactory](#) [▶ 3192] type exposes the following members.

Methods



	Name	Description
	CreatePrimitiveValue [▶ 3193]	Creates a primitive value, independent of any settings.
	CreateValue [▶ 3194]	Creates the symbols value from raw memory data.

Reference

[IAccessorValueFactory Interface](#) [▶ 3192]

[TwinCAT.ValueAccess Namespace](#) [▶ 3179]

Also see about this

-  [IAccessorValueFactory.CreateValue Method \(ISymbol, ReadOnlySpan.Byte., DateTimeOffset\)](#) [▶ 3196]
-  [IAccessorValueFactory.CreateValue Method \(ISymbol, ReadOnlySpan.Byte., IValue, DateTimeOffset\)](#) [▶ 3196]

6.13.3.1.1 IAccessorValueFactory.CreatePrimitiveValue Method

Creates a primitive value, independent of any settings.

Namespace: [TwinCAT.ValueAccess](#) [▶ 3179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object CreatePrimitiveValue(
    ISymbol symbol,
    Memory<byte> sourceData,
    IValue? parent,
    DateTimeOffset timeStamp
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691] The symbol.
sourceData	Type: System.Memory.Byte. The source data.
parent	Type: TwinCAT.TypeSystem.IValue [▶ 2745] The parent.
timeStamp	Type: System.DateTimeOffset The time stamp.

Return Value

Type: Object
A primitive value.

Reference

[IAccessorValueFactory Interface](#) [▶ 3192]

[TwinCAT.ValueAccess Namespace \[► 3179\]](#)

6.13.3.1.2 IAccessorValueFactory.CreateValue Method

Creates the symbols value from raw memory data.

Namespace: [TwinCAT.ValueAccess \[► 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
Object CreateValue(
    ISymbol symbol,
    Memory<byte> sourceData,
    IValue? parent,
    DateTimeOffset timeStamp
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol.
sourceData	Type: System.Memory.Byte. The data.
parent	Type: TwinCAT.TypeSystem.IValue [► 2745] The parent value
timeStamp	Type: System.DateTimeOffset The time stamp.

Return Value

Type: Object
System.Object.

Remarks

The parent argument is used to organize values in hierarchies, equally to the corresponding symbol/instance trees.

Reference

[IAccessorValueFactory Interface \[► 3192\]](#)

[TwinCAT.ValueAccess Namespace \[► 3179\]](#)

6.13.3.1.2.1 IAccessorValueFactory.CreateValue Method (ISymbol, ReadOnlyMemory<byte>, Void)

Namespace: [TwinCAT.ValueAccess \[► 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax

C#

```
Object CreateValue(
    ISymbol symbol,
    ReadOnlyMemory sourceData,
    void timeStamp
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691]
sourceData	Type: ReadOnlyMemory
timeStamp	Type: System.Void

Return Value

Type: [Object](#)

Reference

[IAccessorValueFactory Interface](#) [[▶ 3192](#)]

[CreateValue Overload](#) [[▶ 3194](#)]

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.3.1.2.2 IAccessorValueFactory.CreateValue Method (ISymbol, ReadOnlyMemory`1, Void, Byte)

Namespace: [TwinCAT.ValueAccess](#) [[▶ 3179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 5.0.294+Branch.releases-5.0.Sha.90bb9a1b43b6095934fddca3e72bc0ea15da1c14

Syntax**C#**

```
Object CreateValue(  
    ISymbol symbol,  
    ReadOnlyMemory sourceData,  
    void parent,  
    byte timeStamp  
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▶ 2691]
sourceData	Type: ReadOnlyMemory
parent	Type: System.Void
timeStamp	Type: System.Byte

Return Value

Type: [Object](#)

Reference

[IAccessorValueFactory Interface](#) [[▶ 3192](#)]

[CreateValue Overload](#) [[▶ 3194](#)]

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.3.1.2.3 IAccessorValueFactory.CreateValue Method (ISymbol, ReadOnlySpan.Byte., DateTimeOffset)

Creates the symbols value from raw memory data

Namespace: [TwinCAT.ValueAccess](#) [► 3179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
Object CreateValue(  
    ISymbol symbol,  
    ReadOnlySpan<byte> sourceData,  
    DateTimeOffset timeStamp  
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol.
sourceData	Type: System.ReadOnlySpan.Byte. Memory data.
timeStamp	Type: System.DateTimeOffset The timestamp of the memory data.

Return Value

Type: [Object](#)

The unmarshalled value object.

Reference

[IAccessorValueFactory Interface](#) [► 3192]

[CreateValue Overload](#) [► 3194]

[TwinCAT.ValueAccess Namespace](#) [► 3179]

6.13.3.1.2.4 IAccessorValueFactory.CreateValue Method (ISymbol, ReadOnlySpan.Byte., IValue, DateTimeOffset)

Creates the symbols value from raw memory data.

Namespace: [TwinCAT.ValueAccess](#) [► 3179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
Object CreateValue(  
    ISymbol symbol,  
    ReadOnlySpan<byte> sourceData,  
    IValue parent,  
    DateTimeOffset timeStamp  
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The symbol.
--------	--

sourceData Type: [System.ReadOnlySpan<Byte>](#).
The data.

parent Type: [TwinCAT.TypeSystem.IValue](#) [[▶ 2745](#)]
The parent value

timeStamp Type: [System.DateTimeOffset](#)
The time stamp.

Return Value

Type: [Object](#)
System.Object.

Remarks

The parent argument is used to organize values in hierarchies, equally to the corresponding symbol/instance trees.

Reference

- [IAccessorValueFactory Interface](#) [[▶ 3192](#)]
- [CreateValue Overload](#) [[▶ 3194](#)]
- [TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.4 ResultAccess Class

Result class for an asynchronous access operation.

Inheritance Hierarchy

- System.Object
- TwinCAT.ValueAccess.ResultAccess
- [TwinCAT.ValueAccess.ResultHandleAccess](#) [[▶ 3224](#)]
- [TwinCAT.ValueAccess.ResultReadValueAccess.T.](#) [[▶ 3211](#)]
- [TwinCAT.ValueAccess.ResultWriteAccess](#) [[▶ 3217](#)]

Namespace: [TwinCAT.ValueAccess](#) [[▶ 3179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax

C#






```
public class ResultAccess
```

The ResultAccess type exposes the following members.








Constructors

	Name	Description
	ResultAccess(Int32, UInt32) [▶ 3200]	Initializes a new instance of the ResultAccess class.
	ResultAccess(Int32, DateTimeOffset, UInt32) [▶ 3200]	Initializes a new instance of the ResultReadValueAccess [▶ 3209] struct.

Properties

	Name	Description
	DateTime [▶ 3201]	The Timestamp / the date time of the communication access.
	ErrorCode [▶ 3202]	The ErrorCode of the communication access
	Failed [▶ 3202]	Gets a value indicating whether this ResultAccess represents a failed access.
	Invokeld [▶ 3203]	Gets the invoke identifier (optionally) or 0
	Succeeded [▶ 3202]	Gets a value indicating whether this ResultAccess represents a succeeded access.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [▶ 3204]	Sets the error to this ResultAccess object.
	ToString	Returns a string that represents the current object. (Inherited from Object.)



Remarks

This result is independent of any used protocol and is used by the different protocol providers (ADS, OPC, IOT etc.)



Reference

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

Also see about this

-  [ResultAccess Constructor \(Int32\)](#) [[▶ 3199](#)]
-  [ResultAccess Constructor \(Int32, DateTimeOffset\)](#) [[▶ 3199](#)]

6.13.4.1 ResultAccess Constructor**Overload List**

	Name	Description
	ResultAccess(Int32, UInt32) [▶ 3200]	Initializes a new instance of the ResultAccess [▶ 3197] class.
	ResultAccess(Int32, DateTimeOffset, UInt32) [▶ 3200]	Initializes a new instance of the ResultReadValueAccess [▶ 3209] struct.

Reference

[ResultAccess Class](#) [► 3197]

[TwinCAT.ValueAccess Namespace](#) [► 3179]

Also see about this

- [ResultAccess Constructor \(Int32\)](#) [► 3199]
- [ResultAccess Constructor \(Int32, DateTimeOffset\)](#) [► 3199]

6.13.4.1.1 ResultAccess Constructor (Int32)

Initializes a new instance of the [ResultAccess](#) [► 3197] class.

Namespace: [TwinCAT.ValueAccess](#) [► 3179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public ResultAccess(  
    int errorResult  
)
```

Parameters

errorResult	Type: System.Int32 The result code of the communication access..
-------------	---

Reference

[ResultAccess Class](#) [► 3197]

[ResultAccess Overload](#) [► 3198]

[TwinCAT.ValueAccess Namespace](#) [► 3179]

6.13.4.1.2 ResultAccess Constructor (Int32, DateTimeOffset)

Initializes a new instance of the [ResultReadValueAccess](#) [► 3209] struct.

Namespace: [TwinCAT.ValueAccess](#) [► 3179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public ResultAccess(  
    int errorResult,  
    DateTimeOffset time  
)
```

Parameters

errorResult	Type: System.Int32 The result code of the operation.
time	Type: System.DateTimeOffset The timestamp of the operation.

Reference[ResultAccess Class \[► 3197\]](#)[ResultAccess Overload \[► 3198\]](#)[TwinCAT.ValueAccess Namespace \[► 3179\]](#)**6.13.4.1.3 ResultAccess Constructor (Int32, UInt32)**Initializes a new instance of the [ResultAccess \[► 3197\]](#) class.**Namespace:** [TwinCAT.ValueAccess \[► 3179\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public ResultAccess(
    int errorResult,
    uint invokeId
)
```

Parameters

errorResult	Type: System.Int32 The result code of the communication access..
invokeld	Type: System.UInt32 The ADS request invoke identifier.

Reference[ResultAccess Class \[► 3197\]](#)[ResultAccess Overload \[► 3198\]](#)[TwinCAT.ValueAccess Namespace \[► 3179\]](#)**6.13.4.1.4 ResultAccess Constructor (Int32, DateTimeOffset, UInt32)**Initializes a new instance of the [ResultReadValueAccess \[► 3209\]](#) struct.**Namespace:** [TwinCAT.ValueAccess \[► 3179\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public ResultAccess(
    int errorResult,
    DateTimeOffset time,
    uint invokeId
)
```

Parameters

errorResult	Type: System.Int32 The result code of the operation.
time	Type: System.DateTimeOffset The timestamp of the operation.

invokeld	Type: System.UInt32 The ADS request invoke identifier.
----------	---

Reference

[ResultAccess Class \[▶ 3197\]](#)






[ResultAccess Overload \[▶ 3198\]](#)

[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

6.13.4.2 ResultAccess Properties

The [ResultAccess \[▶ 3197\]](#) type exposes the following members.

Properties

	Name	Description
	DateTime [▶ 3201]	The Timestamp / the date time of the communication access.
	ErrorCode [▶ 3202]	The ErrorCode of the communication access
	Failed [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a failed access.
	Invokeld [▶ 3203]	Gets the invoke identifier (optionally) or 0
	Succeeded [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a succeeded access.

Reference

[ResultAccess Class \[▶ 3197\]](#)

[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

6.13.4.2.1 ResultAccess.DateTime Property

The Timestamp / the date time of the communication access.

Namespace: [TwinCAT.ValueAccess \[▶ 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public DateTimeOffset DateTime { get; }
```

Property Value

Type: DateTimeOffset

Reference

[ResultAccess Class \[▶ 3197\]](#)

[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

6.13.4.2.2 ResultAccess.ErrorCode Property

The ErrorCode of the communication access

Namespace: [TwinCAT.ValueAccess](#) [[▶ 3179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int ErrorCode { get; }
```

Property Value

Type: Int32

Remarks

A value of '0' represents success.

Reference

[ResultAccess Class](#) [[▶ 3197](#)]

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.4.2.3 ResultAccess.Failed Property

Gets a value indicating whether this [ResultAccess](#) [[▶ 3197](#)] represents a failed access.

Namespace: [TwinCAT.ValueAccess](#) [[▶ 3179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Failed { get; }
```

Property Value

Type: Boolean
true if failed; otherwise, false.

Reference

[ResultAccess Class](#) [[▶ 3197](#)]

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.4.2.4 ResultAccess.Succeeded Property

Gets a value indicating whether this [ResultAccess](#) [[▶ 3197](#)] represents a succeeded access.

Namespace: [TwinCAT.ValueAccess](#) [[▶ 3179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool Succeeded { get; }
```

Property Value

Type: Boolean
true if succeeded; otherwise, false.

Reference

[ResultAccess Class \[▸ 3197\]](#)

[TwinCAT.ValueAccess Namespace \[▸ 3179\]](#)

6.13.4.2.5 ResultAccess.InvokeId Property

Gets the invoke identifier (optionally) or 0

Namespace: [TwinCAT.ValueAccess \[▸ 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint InvokeId { get; }
```

Property Value

Type: UInt32
The invoke identifier.

Reference








[ResultAccess Class \[▸ 3197\]](#)

[TwinCAT.ValueAccess Namespace \[▸ 3179\]](#)

6.13.4.3 ResultAccess Methods

The [ResultAccess \[▸ 3197\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [▸ 3204]	Sets the error to this ResultAccess [▸ 3197] object.
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[ResultAccess Class \[▸ 3197\]](#)

[TwinCAT.ValueAccess Namespace \[► 3179\]](#)

6.13.4.3.1 ResultAccess.SetError Method

Sets the error to this [ResultAccess \[► 3197\]](#) object.

Namespace: [TwinCAT.ValueAccess \[► 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public void SetError(
    int errorCode
)
```

Parameters

errorCode	Type: System.Int32 The error code.
-----------	---------------------------------------

Reference

[ResultAccess Class \[► 3197\]](#)

[TwinCAT.ValueAccess Namespace \[► 3179\]](#)

6.13.5 ResultReadDynamicValueAccess Class

Asynchronous read access result returning an Dynamic value object ([IDynamicValue \[► 2504\]](#)). Implements the [ResultReadValueAccess.T. \[► 3211\]](#)

Inheritance Hierarchy

System.Object

[TwinCAT.ValueAccess.ResultAccess \[► 3197\]](#)

[TwinCAT.ValueAccess.ResultReadValueAccess \[► 3211\].IDynamicValue \[► 2504\]](#).

[TwinCAT.ValueAccess.ResultReadDynamicValueAccess](#)

Namespace: [TwinCAT.ValueAccess \[► 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229




Syntax

C#

```
public class ResultReadDynamicValueAccess : ResultReadValueAccess<IDynamicValue>
```

The [ResultReadDynamicValueAccess](#) type exposes the following members.

Properties

	Name	Description
	DateTime [► 3201]	The Timestamp / the date time of the communication access. (Inherited from ResultAccess [► 3197] .)
	ErrorCode [► 3202]	The ErrorCode of the communication access (Inherited from ResultAccess [► 3197] .)
	Failed [► 3202]	Gets a value indicating whether this ResultAccess [► 3197] represents a failed access. (Inherited from ResultAccess [► 3197] .)

	Name	Description
	Invokeld [▶ 3203]	Gets the invoke identifier (optionally) or 0 (Inherited from ResultAccess [▶ 3197].)
	Succeeded [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a succeeded access. (Inherited from ResultAccess [▶ 3197].)
	Value [▶ 3213]	The value (Inherited from ResultReadValueAccess.T. [▶ 3211].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [▶ 3204]	Sets the error to this ResultAccess [▶ 3197] object. (Inherited from ResultAccess [▶ 3197].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

[TwinCAT.ValueAccess.ResultReadValueAccess.T. \[▶ 3211\]](#)

6.13.5.1 ResultReadDynamicValueAccess Properties

The [ResultReadDynamicValueAccess \[▶ 3204\]](#) type exposes the following members.

Properties

	Name	Description
	DateTime [▶ 3201]	The Timestamp / the date time of the communication access. (Inherited from ResultAccess [▶ 3197].)
	ErrorCode [▶ 3202]	The ErrorCode of the communication access (Inherited from ResultAccess [▶ 3197].)
	Failed [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a failed access. (Inherited from ResultAccess [▶ 3197].)
	Invokeld [▶ 3203]	Gets the invoke identifier (optionally) or 0 (Inherited from ResultAccess [▶ 3197].)
	Succeeded [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a succeeded access. (Inherited from ResultAccess [▶ 3197].)
	Value [▶ 3213]	The value (Inherited from ResultReadValueAccess.T. [▶ 3211].)

Reference








[ResultReadDynamicValueAccess Class \[▶ 3204\]](#)

[TwinCAT.ValueAccess Namespace \[► 3179\]](#)

6.13.5.2 ResultReadDynamicValueAccess Methods

The [ResultReadDynamicValueAccess \[► 3204\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [► 3204]	Sets the error to this ResultAccess [► 3197] object. (Inherited from ResultAccess [► 3197] .)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[ResultReadDynamicValueAccess Class \[► 3204\]](#)

[TwinCAT.ValueAccess Namespace \[► 3179\]](#)

6.13.6 ResultReadRawAccess Class

Asynchronous read access result object, reading raw (byte[]) data into memory locations. Implements the [ResultReadValueAccess.T. \[► 3211\]](#)

Inheritance Hierarchy

System.Object

[TwinCAT.ValueAccess.ResultAccess \[► 3197\]](#)

[TwinCAT.ValueAccess.ResultReadValueAccess \[► 3211\].Memory.Byte..](#)

[TwinCAT.ValueAccess.ResultReadRawAccess](#)

Namespace: [TwinCAT.ValueAccess \[► 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax






C#

```
public class ResultReadRawAccess : ResultReadValueAccess<Memory<byte>>
```








The [ResultReadRawAccess](#) type exposes the following members.

Properties

	Name	Description
	DateTime [► 3201]	The Timestamp / the date time of the communication access. (Inherited from ResultAccess [► 3197] .)
	Empty [► 3208]	Gets the empty ResultAnyValue [► 1129] object.

	Name	Description
S		
	ErrorCode [▶ 3202]	The ErrorCode of the communication access (Inherited from ResultAccess [▶ 3197].)
	Failed [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a failed access. (Inherited from ResultAccess [▶ 3197].)
	Invokeld [▶ 3203]	Gets the invoke identifier (optionally) or 0 (Inherited from ResultAccess [▶ 3197].)
	Succeeded [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a succeeded access. (Inherited from ResultAccess [▶ 3197].)
	Value [▶ 3213]	The value (Inherited from ResultReadValueAccess.T. [▶ 3211].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [▶ 3204]	Sets the error to this ResultAccess [▶ 3197] object. (Inherited from ResultAccess [▶ 3197].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference





[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]




[TwinCAT.ValueAccess.ResultReadValueAccess.T.](#) [[▶ 3211](#)]

6.13.6.1 ResultReadRawAccess Properties

The [ResultReadRawAccess](#) [[▶ 3206](#)] type exposes the following members.

Properties

	Name	Description
	DateTime [▶ 3201]	The Timestamp / the date time of the communication access. (Inherited from ResultAccess [▶ 3197].)
	Empty [▶ 3208]	Gets the empty ResultAnyValue [▶ 1129] object.
S		
	ErrorCode [▶ 3202]	The ErrorCode of the communication access (Inherited from ResultAccess [▶ 3197].)
	Failed [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a failed access. (Inherited from ResultAccess [▶ 3197].)

	Name	Description
	InvokeId [▶ 3203]	Gets the invoke identifier (optionally) or 0 (Inherited from ResultAccess [▶ 3197].)
	Succeeded [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a succeeded access. (Inherited from ResultAccess [▶ 3197].)
	Value [▶ 3213]	The value (Inherited from ResultReadValueAccess.T. [▶ 3211].)

Reference

[ResultReadRawAccess Class](#) [[▶ 3206](#)]

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.6.1.1 ResultReadRawAccess.Empty Property

Gets the empty [ResultAnyValue](#) [[▶ 1129](#)] object.

Namespace: [TwinCAT.ValueAccess](#) [[▶ 3179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultReadRawAccess Empty { get; }
```

Property Value

Type: [ResultReadRawAccess](#) [[▶ 3206](#)]

The empty / unprocessed result.

Reference







[ResultReadRawAccess Class](#) [[▶ 3206](#)]


[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.6.2 ResultReadRawAccess Methods

The [ResultReadRawAccess](#) [[▶ 3206](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	SetError [▶ 3204]	Sets the error to this ResultAccess [▶ 3197] object. (Inherited from ResultAccess [▶ 3197].)

	Name	Description
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[ResultReadRawAccess Class \[▶ 3206\]](#)

[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

6.13.7 ResultReadValueAccess Class

Asynchronous read access result returning an untyped Value object type 'object'. Implements the [ResultAccess \[▶ 3197\]](#)

Inheritance Hierarchy

System.Object

[TwinCAT.ValueAccess.ResultAccess \[▶ 3197\]](#)

[TwinCAT.ValueAccess.ResultReadValueAccess \[▶ 3211\].Object.](#)

[TwinCAT.ValueAccess.ResultReadValueAccess](#)

[TwinCAT.ValueAccess.ResultRpcMethodAccess \[▶ 3213\]](#)

Namespace: [TwinCAT.ValueAccess \[▶ 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229







Syntax

C#



```
public class ResultReadValueAccess : ResultReadValueAccess<Object>
```






The ResultReadValueAccess type exposes the following members.

Properties

	Name	Description
	DateTime [▶ 3201]	The Timestamp / the date time of the communication access. (Inherited from ResultAccess [▶ 3197].)
	ErrorCode [▶ 3202]	The ErrorCode of the communication access (Inherited from ResultAccess [▶ 3197].)
	Failed [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a failed access. (Inherited from ResultAccess [▶ 3197].)
	Invokeld [▶ 3203]	Gets the invoke identifier (optionally) or 0 (Inherited from ResultAccess [▶ 3197].)
	Succeeded [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a succeeded access. (Inherited from ResultAccess [▶ 3197].)
	Value [▶ 3213]	The value (Inherited from ResultReadValueAccess.T. [▶ 3211].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)

	Name	Description
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [▶ 3204]	Sets the error to this ResultAccess [▶ 3197] object. (Inherited from ResultAccess [▶ 3197].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference







[TwinCAT.ValueAccess Namespace \[\[▶ 3179\]\(#\)\]](#)

[TwinCAT.ValueAccess.ResultAccess \[\[▶ 3197\]\(#\)\]](#)

6.13.7.1 ResultReadValueAccess Properties

The [ResultReadValueAccess \[\[▶ 3209\]\(#\)\]](#) type exposes the following members.

Properties

	Name	Description
	DateTime [▶ 3201]	The Timestamp / the date time of the communication access. (Inherited from ResultAccess [▶ 3197] .)
	ErrorCode [▶ 3202]	The ErrorCode of the communication access (Inherited from ResultAccess [▶ 3197] .)
	Failed [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a failed access. (Inherited from ResultAccess [▶ 3197] .)
	Invokeld [▶ 3203]	Gets the invoke identifier (optionally) or 0 (Inherited from ResultAccess [▶ 3197] .)
	Succeeded [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a succeeded access. (Inherited from ResultAccess [▶ 3197] .)
	Value [▶ 3213]	The value (Inherited from ResultReadValueAccess.T. [▶ 3211] .)

Reference



[ResultReadValueAccess Class \[\[▶ 3209\]\(#\)\]](#)






[TwinCAT.ValueAccess Namespace \[\[▶ 3179\]\(#\)\]](#)

6.13.7.2 ResultReadValueAccess Methods

The [ResultReadValueAccess \[\[▶ 3209\]\(#\)\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)

	Name	Description
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [▶ 3204]	Sets the error to this ResultAccess [▶ 3197] object. (Inherited from ResultAccess [▶ 3197] .)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[ResultReadValueAccess Class \[▶ 3209\]](#)

[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

6.13.8 ResultReadValueAccess.T. Class

Result object of an asynchronous read of a specific value of type T. Implements the [ResultAccess \[▶ 3197\]](#)

Inheritance Hierarchy

System.Object

[TwinCAT.ValueAccess.ResultAccess \[▶ 3197\]](#)

[TwinCAT.ValueAccess.ResultReadValueAccess.T.](#)

[TwinCAT.ValueAccess.ResultReadDynamicValueAccess \[▶ 3204\]](#)

[TwinCAT.ValueAccess.ResultReadRawAccess \[▶ 3206\]](#)

[TwinCAT.ValueAccess.ResultReadValueAccess \[▶ 3209\]](#)

[TwinCAT.ValueAccess.ResultReadValueAccess2.I, V. \[▶ 3226\]](#)

Namespace: [TwinCAT.ValueAccess \[▶ 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#





```
public class ResultReadValueAccess<T> : ResultAccess
```



Type Parameters

T	The type of the value.
---	------------------------








The ResultReadValueAccess.T. type exposes the following members.

Properties

	Name	Description
	DateTime [▶ 3201]	The Timestamp / the date time of the communication access. (Inherited from ResultAccess [▶ 3197] .)
	ErrorCode [▶ 3202]	The ErrorCode of the communication access (Inherited from ResultAccess [▶ 3197] .)
	Failed [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a failed access. (Inherited from ResultAccess [▶ 3197] .)
	Invokeld [▶ 3203]	Gets the invoke identifier (optionally) or 0 (Inherited from ResultAccess [▶ 3197] .)

	Name	Description
	Succeeded [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a succeeded access. (Inherited from ResultAccess [▶ 3197].)
	Value [▶ 3213]	The value

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [▶ 3204]	Sets the error to this ResultAccess [▶ 3197] object. (Inherited from ResultAccess [▶ 3197].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference







[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

[TwinCAT.ValueAccess.ResultAccess](#) [[▶ 3197](#)]

6.13.8.1 ResultReadValueAccess.T. Properties

The [ResultReadValueAccess.T.](#) [[▶ 3211](#)] generic type exposes the following members.

Properties

	Name	Description
	DateTime [▶ 3201]	The Timestamp / the date time of the communication access. (Inherited from ResultAccess [▶ 3197].)
	ErrorCode [▶ 3202]	The ErrorCode of the communication access (Inherited from ResultAccess [▶ 3197].)
	Failed [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a failed access. (Inherited from ResultAccess [▶ 3197].)
	InvokeId [▶ 3203]	Gets the invoke identifier (optionally) or 0 (Inherited from ResultAccess [▶ 3197].)
	Succeeded [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a succeeded access. (Inherited from ResultAccess [▶ 3197].)
	Value [▶ 3213]	The value

Reference

[ResultReadValueAccess.T. Class](#) [[▶ 3211](#)]

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.8.1.1 ResultReadValueAccess.T.Value Property

The value

Namespace: [TwinCAT.ValueAccess](#) [[▶ 3179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public T Value { get; }
```

Property Value

Type: [T](#) [[▶ 3211](#)]

Reference








[ResultReadValueAccess.T. Class](#) [[▶ 3211](#)]

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.8.2 ResultReadValueAccess.T. Methods

The [ResultReadValueAccess.T.](#) [[▶ 3211](#)] generic type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	setError [▶ 3204]	Sets the error to this ResultAccess [▶ 3197] object. (Inherited from ResultAccess [▶ 3197].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[ResultReadValueAccess.T. Class](#) [[▶ 3211](#)]

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.9 ResultRpcMethodAccess Class

Asynchronous Invoke RPC Method result class. Implements the [ResultReadValueAccess](#) [[▶ 3209](#)]

Inheritance Hierarchy

System.Object

[TwinCAT.ValueAccess.ResultAccess](#) [▶ 3197]

[TwinCAT.ValueAccess.ResultReadValueAccess](#) [▶ 3211].Object.

[TwinCAT.ValueAccess.ResultReadValueAccess](#) [▶ 3209]

[TwinCAT.ValueAccess.ResultRpcMethodAccess](#)

Namespace: [TwinCAT.ValueAccess](#) [▶ 3179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229


Syntax

C#









```
public class ResultRpcMethodAccess : ResultReadValueAccess
```

The ResultRpcMethodAccess type exposes the following members.





Constructors




	Name	Description
	ResultRpcMethodAccess [▶ 3215]	Initializes a new instance of the ResultRpcMethod [▶ 1176] struct.

Properties

	Name	Description
	DateTime [▶ 3201]	The Timestamp / the date time of the communication access. (Inherited from ResultAccess [▶ 3197].)
	ErrorCode [▶ 3202]	The ErrorCode of the communication access (Inherited from ResultAccess [▶ 3197].)
	Failed [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a failed access. (Inherited from ResultAccess [▶ 3197].)
	Invokeld [▶ 3203]	Gets the invoke identifier (optionally) or 0 (Inherited from ResultAccess [▶ 3197].)
	OutParameters [▶ 3216]	Gets the out parameters.
	ReturnValue [▶ 3216]	Gets the return value of the RpcMethod (optionally).
	Succeeded [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a succeeded access. (Inherited from ResultAccess [▶ 3197].)
	Value [▶ 3213]	The value (Inherited from ResultReadValueAccess.T [▶ 3211].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)

	Name	Description
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	setError [▶ 3204]	Sets the error to this ResultAccess [▶ 3197] object. (Inherited from ResultAccess [▶ 3197].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[TwinCAT.ValueAccess Namespace \[\[▶ 3179\]\(#\)\]](#)

[TwinCAT.ValueAccess.ResultReadValueAccess \[\[▶ 3209\]\(#\)\]](#)

6.13.9.1 ResultRpcMethodAccess Constructor

Initializes a new instance of the [ResultRpcMethod \[\[▶ 1176\]\(#\)\]](#) struct.

Namespace: [TwinCAT.ValueAccess \[\[▶ 3179\]\(#\)\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ResultRpcMethodAccess (
    Object? returnValue,
    Object[]? outParameters,
    int errorCode,
    DateTimeOffset timeStamp,
    uint invokeId
)
```

Parameters

returnValue	Type: System.Object The value.
outParameters	Type: .System.Object. The out parameters.
errorCode	Type: System.Int32 The error code.
timeStamp	Type: System.DateTimeOffset The time stamp.
invokeld	Type: System.UInt32 The ADS request invoke identifier.

Reference









[ResultRpcMethodAccess Class \[\[▶ 3213\]\(#\)\]](#)

[TwinCAT.ValueAccess Namespace \[\[▶ 3179\]\(#\)\]](#)

6.13.9.2 ResultRpcMethodAccess Properties

The [ResultRpcMethodAccess \[\[▶ 3213\]\(#\)\]](#) type exposes the following members.

Properties

	Name	Description
	DateTime [▶ 3201]	The Timestamp / the date time of the communication access. (Inherited from ResultAccess [▶ 3197].)
	ErrorCode [▶ 3202]	The ErrorCode of the communication access (Inherited from ResultAccess [▶ 3197].)
	Failed [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a failed access. (Inherited from ResultAccess [▶ 3197].)
	Invokeld [▶ 3203]	Gets the invoke identifier (optionally) or 0 (Inherited from ResultAccess [▶ 3197].)
	OutParameters [▶ 3216]	Gets the out parameters.
	ReturnValue [▶ 3216]	Gets the return value of the RpcMethod (optionally).
	Succeeded [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a succeeded access. (Inherited from ResultAccess [▶ 3197].)
	Value [▶ 3213]	The value (Inherited from ResultReadValueAccess.T. [▶ 3211].)

Reference

[ResultRpcMethodAccess Class](#) [[▶ 3213](#)]

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.9.2.1 ResultRpcMethodAccess.OutParameters Property

Gets the out parameters.

Namespace: [TwinCAT.ValueAccess](#) [[▶ 3179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public Object[]? OutParameters { get; }
```

Property Value

Type: .Object.
The out parameters.

Reference

[ResultRpcMethodAccess Class](#) [[▶ 3213](#)]

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.9.2.2 ResultRpcMethodAccess.ReturnValue Property

Gets the return value of the RpcMethod (optionally).

Namespace: [TwinCAT.ValueAccess](#) [[▶ 3179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Object? ReturnValue { get; }
```

Property Value

Type: Object
The return value or NULL

Reference








[ResultRpcMethodAccess Class \[► 3213\]](#)

[TwinCAT.ValueAccess Namespace \[► 3179\]](#)

6.13.9.3 ResultRpcMethodAccess Methods

The [ResultRpcMethodAccess \[► 3213\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [► 3204]	Sets the error to this ResultAccess [► 3197] object. (Inherited from ResultAccess [► 3197] .)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[ResultRpcMethodAccess Class \[► 3213\]](#)

[TwinCAT.ValueAccess Namespace \[► 3179\]](#)

6.13.10 ResultWriteAccess Class

Asynchronous write access result class.

Inheritance Hierarchy

System.Object

[TwinCAT.ValueAccess.ResultAccess \[► 3197\]](#)

[TwinCAT.ValueAccess.ResultWriteAccess](#)

Namespace: [TwinCAT.ValueAccess \[► 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229



Syntax

C#







```
public class ResultWriteAccess : ResultAccess
```

The ResultWriteAccess type exposes the following members.








Constructors

	Name	Description
	ResultWriteAccess(Int32, UInt32) [► 3220]	Initializes a new instance of the ResultWriteAccess class.
	ResultWriteAccess(Int32, DateTimeOffset, UInt32) [► 3221]	Initializes a new instance of the ResultWriteAccess struct.

Properties

	Name	Description
	DateTime [► 3201]	The Timestamp / the date time of the communication access. (Inherited from ResultAccess [► 3197] .)
	Empty [► 3222]	Gets the empty ResultWriteAccess object.
	ErrorCode [► 3202]	The ErrorCode of the communication access (Inherited from ResultAccess [► 3197] .)
	Failed [► 3202]	Gets a value indicating whether this ResultAccess [► 3197] represents a failed access. (Inherited from ResultAccess [► 3197] .)
	Invokeld [► 3203]	Gets the invoke identifier (optionally) or 0 (Inherited from ResultAccess [► 3197] .)
	Succeeded [► 3202]	Gets a value indicating whether this ResultAccess [► 3197] represents a succeeded access. (Inherited from ResultAccess [► 3197] .)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [► 3204]	Sets the error to this ResultAccess [► 3197] object. (Inherited from ResultAccess [► 3197] .)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference



[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

Also see about this

- [ResultWriteAccess Constructor \(Int32\) \[▶ 3219\]](#)
- [ResultWriteAccess Constructor \(Int32, DateTimeOffset\) \[▶ 3220\]](#)

6.13.10.1 ResultWriteAccess Constructor

Overload List

	Name	Description
	ResultWriteAccess(Int32, UInt32) [▶ 3220]	Initializes a new instance of the ResultWriteAccess [▶ 3217] class.
	ResultWriteAccess(Int32, DateTimeOffset, UInt32) [▶ 3221]	Initializes a new instance of the ResultWriteAccess [▶ 3217] struct.

Reference

[ResultWriteAccess Class \[▶ 3217\]](#)

[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

Also see about this

- [ResultWriteAccess Constructor \(Int32\) \[▶ 3219\]](#)
- [ResultWriteAccess Constructor \(Int32, DateTimeOffset\) \[▶ 3220\]](#)

6.13.10.1.1 ResultWriteAccess Constructor (Int32)

Initializes a new instance of the [ResultWriteAccess \[▶ 3217\]](#) class.

Namespace: [TwinCAT.ValueAccess \[▶ 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public ResultWriteAccess(
    int errorResult
)
```

Parameters

errorResult Type: [System.Int32](#)
 The result code of the communication access..

Reference

[ResultWriteAccess Class \[▶ 3217\]](#)

[ResultWriteAccess Overload \[▶ 3219\]](#)

[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

6.13.10.1.2 ResultWriteAccess Constructor (Int32, DateTimeOffset)

Initializes a new instance of the [ResultWriteAccess](#) [▸ 3217] struct.

Namespace: [TwinCAT.ValueAccess](#) [▸ 3179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public ResultWriteAccess(
    int errorResult,
    DateTimeOffset timeStamp
)
```

Parameters

errorResult	Type: System.Int32 The error result.
timeStamp	Type: System.DateTimeOffset The time stamp.

Reference

[ResultWriteAccess Class](#) [▸ 3217]

[ResultWriteAccess Overload](#) [▸ 3219]

[TwinCAT.ValueAccess Namespace](#) [▸ 3179]

6.13.10.1.3 ResultWriteAccess Constructor (Int32, UInt32)

Initializes a new instance of the [ResultWriteAccess](#) [▸ 3217] class.

Namespace: [TwinCAT.ValueAccess](#) [▸ 3179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ResultWriteAccess(
    int errorResult,
    uint invokeId
)
```

Parameters

errorResult	Type: System.Int32 The result code of the communication access.
invokeld	Type: System.UInt32 The ADS request invoke identifier.

Reference

[ResultWriteAccess Class](#) [▸ 3217]

[ResultWriteAccess Overload](#) [▸ 3219]

[TwinCAT.ValueAccess Namespace](#) [▸ 3179]

6.13.10.1.4 ResultWriteAccess Constructor (Int32, DateTimeOffset, UInt32)

Initializes a new instance of the [ResultWriteAccess](#) [▶ 3217] struct.

Namespace: [TwinCAT.ValueAccess](#) [▶ 3179]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ResultWriteAccess(
    int errorResult,
    DateTimeOffset timeStamp,
    uint invokeId
)
```

Parameters

errorResult	Type: System.Int32 The error result.
timeStamp	Type: System.DateTimeOffset The time stamp.
invokeld	Type: System.UInt32 The ADS request invoke identifier.

Reference

[ResultWriteAccess Class](#) [▶ 3217]







[ResultWriteAccess Overload](#) [▶ 3219]

[TwinCAT.ValueAccess Namespace](#) [▶ 3179]

6.13.10.2 ResultWriteAccess Properties

The [ResultWriteAccess](#) [▶ 3217] type exposes the following members.

Properties

	Name	Description
	DateTime [▶ 3201]	The Timestamp / the date time of the communication access. (Inherited from ResultAccess [▶ 3197].)
	Empty [▶ 3222]	Gets the empty ResultWriteAccess [▶ 3217] object.
	ErrorCode [▶ 3202]	The ErrorCode of the communication access (Inherited from ResultAccess [▶ 3197].)
	Failed [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a failed access. (Inherited from ResultAccess [▶ 3197].)
	Invokeld [▶ 3203]	Gets the invoke identifier (optionally) or 0 (Inherited from ResultAccess [▶ 3197].)
	Succeeded [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a succeeded access. (Inherited from ResultAccess [▶ 3197].)

Reference

[ResultWriteAccess Class](#) [▶ 3217]

[TwinCAT.ValueAccess Namespace \[► 3179\]](#)

6.13.10.2.1 ResultWriteAccess.Empty Property

Gets the empty [ResultWriteAccess \[► 3217\]](#) object.

Namespace: [TwinCAT.ValueAccess \[► 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static ResultWriteAccess Empty { get; }
```

Property Value

Type: [ResultWriteAccess \[► 3217\]](#)

The empty / unprocessed result.

Reference








[ResultWriteAccess Class \[► 3217\]](#)

[TwinCAT.ValueAccess Namespace \[► 3179\]](#)

6.13.10.3 ResultWriteAccess Methods

The [ResultWriteAccess \[► 3217\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [► 3204]	Sets the error to this ResultAccess [► 3197] object. (Inherited from ResultAccess [► 3197] .)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[ResultWriteAccess Class \[► 3217\]](#)

[TwinCAT.ValueAccess Namespace \[► 3179\]](#)

6.13.11 SymbolNotificationTypes Enumeration

Specifies the Notification type of ADS Notifications

Namespace: [TwinCAT.ValueAccess \[▸ 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
[FlagsAttribute]
public enum SymbolNotificationTypes
```

Members

	Member name	Value	Description
	None	0	None / Uninitialized
	Value	1	ValueChanged notifications
	RawValue	2	RawValueChanged notifications
	Both	3	ValueChanged + RawValueChanged notifications

Reference

[TwinCAT.ValueAccess Namespace \[▸ 3179\]](#)

6.13.12 ValueCreationModes Enumeration

Creation mode for Values

Namespace: [TwinCAT.ValueAccess \[▸ 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
[FlagsAttribute]
public enum ValueCreationModes
```

Members

	Member name	Value	Description
	None	0	None / Uninitialized Mask
	Primitives	1	Convert to .NET Primitives, wherever possible.
	Enums	2	Use IEnumValue on EnumTypes instead of .NET Primitives
	FullDynamics	4	Wraps all Primitives also in IValue Objects
	PlcOpenTypes	8	Use PlcOpen Times (TIME, LTIME, DT, TOD, DATETIME) instead of .NET Primitives DateTime and TimeSpan
	Default	1	Default settings for the value creation mode (Translate to Primitives)

Remarks

This setting is used by the ValueFactory/ to create Read values. In default primitive mode all values will be transferred to Primitive .NET Symbols if possible. E.g PlcOpen.TIME --> TimeSpan, IEnumValue --> .NET Primitives. They won't be wrapped into

Reference

[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

6.13.13 ResultHandleAccess Class

Asynchronous read access result returning an variable handle. Implements the [ResultAccess \[▶ 3197\]](#)

Inheritance Hierarchy

System.Object

[TwinCAT.ValueAccess.ResultAccess \[▶ 3197\]](#)

[TwinCAT.ValueAccess.ResultHandleAccess](#)

Namespace: [TwinCAT.ValueAccess \[▶ 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public class ResultHandleAccess : ResultAccess
```

The ResultHandleAccess type exposes the following members.

Properties

Name	Description
DateTime [▶ 3201]	The Timestamp / the date time of the communication access. (Inherited from ResultAccess [▶ 3197] .)
ErrorCode [▶ 3202]	The ErrorCode of the communication access (Inherited from ResultAccess [▶ 3197] .)
Failed [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a failed access. (Inherited from ResultAccess [▶ 3197] .)
Handle [▶ 3225]	Gets the handle.
Invokeld [▶ 3203]	Gets the invoke identifier (optionally) or 0 (Inherited from ResultAccess [▶ 3197] .)
Succeeded [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a succeeded access. (Inherited from ResultAccess [▶ 3197] .)

Methods

Name	Description
Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
GetHashCode	Serves as the default hash function. (Inherited from Object.)
GetType	Gets the Type of the current instance. (Inherited from Object.)
MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
SetError [▶ 3204]	Sets the error to this ResultAccess [▶ 3197] object. (Inherited from ResultAccess [▶ 3197] .)
ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[TwinCAT.ValueAccess Namespace \[► 3179\]](#)

[TwinCAT.ValueAccess.ResultAccess \[► 3197\]](#)

6.13.13.1 ResultHandleAccess Properties

The [ResultHandleAccess \[► 3224\]](#) type exposes the following members.

Properties

	Name	Description
	DateTime [► 3201]	The Timestamp / the date time of the communication access. (Inherited from ResultAccess [► 3197] .)
	ErrorCode [► 3202]	The ErrorCode of the communication access (Inherited from ResultAccess [► 3197] .)
	Failed [► 3202]	Gets a value indicating whether this ResultAccess [► 3197] represents a failed access. (Inherited from ResultAccess [► 3197] .)
	Handle [► 3225]	Gets the handle.
	Invokeld [► 3203]	Gets the invoke identifier (optionally) or 0 (Inherited from ResultAccess [► 3197] .)
	Succeeded [► 3202]	Gets a value indicating whether this ResultAccess [► 3197] represents a succeeded access. (Inherited from ResultAccess [► 3197] .)

Reference

[ResultHandleAccess Class \[► 3224\]](#)

[TwinCAT.ValueAccess Namespace \[► 3179\]](#)

6.13.13.1.1 ResultHandleAccess.Handle Property

Gets the handle.

Namespace: [TwinCAT.ValueAccess \[► 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint Handle { get; }
```

Property Value

Type: UInt32
The handle.

Reference

[ResultHandleAccess Class \[► 3224\]](#)

[TwinCAT.ValueAccess Namespace \[► 3179\]](#)

6.13.13.2 ResultHandleAccess Methods

The [ResultHandleAccess \[► 3224\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	setError [▶ 3204]	Sets the error to this ResultAccess [▶ 3197] object. (Inherited from ResultAccess [▶ 3197].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[ResultHandleAccess Class](#) [[▶ 3224](#)]

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.14 ResultReadValueAccess2.I, V. Class

Result object of an asynchronous read. Implements the [ResultAccess](#) [[▶ 3197](#)]

Inheritance Hierarchy

System.Object

[TwinCAT.ValueAccess.ResultAccess](#) [[▶ 3197](#)]

[TwinCAT.ValueAccess.ResultReadValueAccess](#) [[▶ 3211](#)].V.

[TwinCAT.ValueAccess.ResultReadValueAccess2.I, V.](#)

Namespace: [TwinCAT.ValueAccess](#) [[▶ 3179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public class ResultReadValueAccess2<I, V> : ResultReadValueAccess<V>
```

Type Parameters

I	The SymbolID / AddressInformation bound to the value.
V	The type of the read Value.

The ResultReadValueAccess2.I, V. type exposes the following members.

Properties

	Name	Description
	DateTime [▶ 3201]	The Timestamp / the date time of the communication access. (Inherited from ResultAccess [▶ 3197].)
	ErrorCode [▶ 3202]	The ErrorCode of the communication access (Inherited from ResultAccess [▶ 3197].)
	Failed [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a failed access. (Inherited from ResultAccess [▶ 3197].)

	Name	Description
	Invokeld [▶ 3203]	Gets the invoke identifier (optionally) or 0 (Inherited from ResultAccess [▶ 3197].)
	Source [▶ 3228]	Gets the source of the value (symbolic specifier/Address).
	Succeeded [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a succeeded access. (Inherited from ResultAccess [▶ 3197].)
	Value [▶ 3213]	The value (Inherited from ResultReadValueAccess.T. [▶ 3211].)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	SetError [▶ 3204]	Sets the error to this ResultAccess [▶ 3197] object. (Inherited from ResultAccess [▶ 3197].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

[TwinCAT.ValueAccess.ResultAccess \[▶ 3197\]](#)

6.13.14.1 ResultReadValueAccess2.I, V. Properties

The [ResultReadValueAccess2.I, V. \[▶ 3226\]](#) generic type exposes the following members.

Properties

	Name	Description
	DateTime [▶ 3201]	The Timestamp / the date time of the communication access. (Inherited from ResultAccess [▶ 3197].)
	ErrorCode [▶ 3202]	The ErrorCode of the communication access (Inherited from ResultAccess [▶ 3197].)
	Failed [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a failed access. (Inherited from ResultAccess [▶ 3197].)
	Invokeld [▶ 3203]	Gets the invoke identifier (optionally) or 0 (Inherited from ResultAccess [▶ 3197].)
	Source [▶ 3228]	Gets the source of the value (symbolic specifier/Address).
	Succeeded [▶ 3202]	Gets a value indicating whether this ResultAccess [▶ 3197] represents a succeeded access. (Inherited from ResultAccess [▶ 3197].)
	Value [▶ 3213]	The value (Inherited from ResultReadValueAccess.T. [▶ 3211].)

Reference

[ResultReadValueAccess2.I, V. Class \[▶ 3226\]](#)

[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

6.13.14.1 ResultReadValueAccess2.I, V..Source Property

Gets the source of the value (symbolic specifier/Address).

Namespace: [TwinCAT.ValueAccess](#) [[▶ 3179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public I Source { get; }
```

Property Value

Type: [I](#) [[▶ 3226](#)]

The source.

Reference

[ResultReadValueAccess2.I, V. Class](#) [[▶ 3226](#)]

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.14.2 ResultReadValueAccess2.I, V. Methods

The [ResultReadValueAccess2.I, V.](#) [[▶ 3226](#)] generic type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	setError [▶ 3204]	Sets the error to this ResultAccess [▶ 3197] object. (Inherited from ResultAccess [▶ 3197].)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Reference

[ResultReadValueAccess2.I, V. Class](#) [[▶ 3226](#)]

[TwinCAT.ValueAccess Namespace](#) [[▶ 3179](#)]

6.13.15 ValueAccessMode Enumeration

Enum ValueAccessMethod

Namespace: [TwinCAT.ValueAccess](#) [[▶ 3179](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum ValueAccessMode
```

Members

	Member name	Value	Description
	None	0	None / Uninitialized
	IndexGroupOffset	1	Value access via Index Group and Offset Only
	SymbolicByHandle	2	Symbolic access via Instance Path only.
	IndexGroupOffsetPreferred	3	Uses IndexGroup IndexOffset Preferred (and Symbolic for Dereferenced Pointers / References)
	InstancePath	4	Access by symbolic instance path.
	Default	2	The Default access mode (SymbolicByHandle)

Remarks

Mode	Description
None	None/Uninitialized. No Valid mode.
IndexGroupOffset	Communicates over IndexGroup/IndexOffset only. This is the most direct/efficient access into the Process image. The advantage is that, the symbol access is done via 1 ADS round trip. Disadvantages are that not all Symbols can be accessed via IG/IO (e.g. References) and IndexOffsets could be invalid after online changes / PlcProgram downloads. Detection of these events and following invalidation of all changed symbols need to be done within the user application.
SymbolicByHandle	The Symbolic-only mode is the most safe mode to use but needs more time than the IndexGroupOffset. It could need up to 3 ADS round trips (create handle, access value, close handle) but is not influenced by online changes or / plcProgram downloads.
IndexGroupOffsetPreferred	This is a mixed access mode. For symbols, where it is possible it uses the IndexGroup/IndexOffset. For others it chooses the SymbolicByHandle access.
Default	The Default-Mode setting if no other ValueAccessMode is specified. This is set to SymbolicByHandle.

Reference

[TwinCAT.ValueAccess Namespace \[▶ 3179\]](#)

6.13.16 ValueUpdateMode Enumeration

Value Update Mode.

Namespace: [TwinCAT.ValueAccess \[▶ 3179\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public enum ValueUpdateMode
```

Members

	Member name	Value	Description
	None	0	No automatic Value Update / Uninitialized
	Immediately	1	Update Value immediately on property set access.
	Triggered	2	Triggers the ValueUpdate explicitly

Remarks

Mode	Description
Immediately	Writes the values of this DynamicValue instantly when setting its value or the value of its child members/elements.
Triggered	Caches internally the value of this DynamicValue until the DynamicValue.Write method is called. This reduces ADS roundtrips, if one or more member/element values should be changed. Furthermore the write on the destination system happens consistently in one ADS Write operation, which could be important for dependent properties/members/elements.

Reference

[TwinCAT.ValueAccess Namespace \[► 3179\]](#)

6.14 TwinCAT.Ads.Server.TypeSystem Namespace

Classes

	Class	Description
	DataArea [► 3230]	Class DataArea.
	ServerSymbolFactory [► 3242]	Class ServerSymbolFactory.

Interfaces

	Interface	Description
	IAddSymbolicInformation [► 3234]	Interface IAddSymbolicInformation

6.14.1 DataArea Class

Class DataArea.

Inheritance Hierarchy

[System.Object](#)

[TwinCAT.Ads.Server.TypeSystem.DataArea](#)

Namespace: [TwinCAT.Ads.Server.TypeSystem \[► 3230\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public class DataArea
```

The DataArea type exposes the following members.

Constructors

	Name	Description
	DataArea [▶ 3231]	Initializes a new instance of the DataArea class.

Properties

	Name	Description
	IndexGroup [▶ 3232]	Gets the index group of the DataArea.
	IndexOffset [▶ 3233]	Gets the index offset of the DataArea.
	Name [▶ 3233]	Gets the name of the DataArea.
	Size [▶ 3234]	Gets the size of the DataArea.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Remarks

The DataArea class organizes the ProcessImage in different memory areas to address them independently via their IndexGroup.

Reference

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.1.1 DataArea Constructor

Initializes a new instance of the [DataArea](#) [[▶ 3230](#)] class.

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[▶ 3230](#)]

Assembly: [TwinCAT.Ads.SymbolicServer](#) (in [TwinCAT.Ads.SymbolicServer.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public DataArea(
    string name,
    uint indexGroup,
```

```

    uint indexOffset,
    uint size
)

```

Parameters

name	Type: System.String The name.
indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.
size	Type: System.UInt32 The size.

Exceptions

Exception	Condition
ArgumentNullException	name

Reference

[DataArea Class](#) [[▶ 3230](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.1.2 DataArea Properties

The [DataArea](#) [[▶ 3230](#)] type exposes the following members.

Properties

	Name	Description
	IndexGroup [▶ 3232]	Gets the index group of the DataArea [▶ 3230].
	IndexOffset [▶ 3233]	Gets the index offset of the DataArea [▶ 3230].
	Name [▶ 3233]	Gets the name of the DataArea [▶ 3230].
	Size [▶ 3234]	Gets the size of the DataArea [▶ 3230].

Reference

[DataArea Class](#) [[▶ 3230](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.1.2.1 DataArea.IndexGroup Property

Gets the index group of the [DataArea](#) [[▶ 3230](#)].

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[▶ 3230](#)]

Assembly: [TwinCAT.Ads.SymbolicServer](#) (in [TwinCAT.Ads.SymbolicServer.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public uint IndexGroup { get; }
```


Property Value

Type: [UInt32](#)
The index group.

Reference

[DataArea Class](#) [[▶ 3230](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.1.2.2 DataArea.IndexOffset Property

Gets the index offset of the [DataArea](#) [[▶ 3230](#)].

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[▶ 3230](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public uint IndexOffset { get; }
```

Property Value

Type: [UInt32](#)
The index offset.

Reference

[DataArea Class](#) [[▶ 3230](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.1.2.3 DataArea.Name Property

Gets the name of the [DataArea](#) [[▶ 3230](#)].

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[▶ 3230](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Name { get; }
```

Property Value

Type: [String](#)
The name.

Reference

[DataArea Class](#) [[▶ 3230](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.1.2.4 DataArea.Size Property

Gets the size of the [DataArea](#) [[▶ 3230](#)].

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[▶ 3230](#)]

Assembly: [TwinCAT.Ads.SymbolicServer](#) (in [TwinCAT.Ads.SymbolicServer.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public uint Size { get; }
```

Property Value

Type: [UInt32](#)

The size.

Reference

[DataArea Class](#) [[▶ 3230](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.1.3 DataArea Methods

The [DataArea](#) [[▶ 3230](#)] type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[DataArea Class](#) [[▶ 3230](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.2 IAddSymbolicInformation Interface

Interface [IAddSymbolicInformation](#)

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[▶ 3230](#)]

Assembly: [TwinCAT.Ads.SymbolicServer](#) (in [TwinCAT.Ads.SymbolicServer.dll](#)) Version: 6.0.328+39e3229

Syntax

C#

```
public interface IAddSymbolicInformation
```

The IAddSymbolicInformation type exposes the following members.

Methods

	Name	Description
	AddDataArea [▶ 3236]	Adds a DataArea [▶ 3230]to the symbolic information.
	AddSymbol(String, String, String) [▶ 3237]	Adds and registers a ISymbol [▶ 2691].
	AddSymbol(String, IDataType, DataArea) [▶ 3238]	Adds and registers a symbol
	AddType [▶ 3238]	Adds and registers a DataType [▶ 1721]
	ClearDataAreas [▶ 3239]	Removes all DataAreas.
	ClearSymbols [▶ 3239]	Removes all symbols.
	ClearTypes [▶ 3240]	Removes all DataTypes.
	RemoveDataArea [▶ 3240]	Removes a DataArea [▶ 3230] from the symbolic information.
	RemoveSymbol(ISymbol) [▶ 3241]	Removes the ISymbol [▶ 2691] from the Symbolic information.
	RemoveSymbol(String) [▶ 3241]	Removes the ISymbol [▶ 2691] from the Symbolic information
	RemoveType [▶ 3242]	Removes the DataType [▶ 1721] from the symbolic information.

Remarks

Fluent interface for method chaining (method cascading) [ISymbol](#) [▶ 2691] and [DataType](#) [▶ 1721] information. This helps building up DataTypes, DataAreas and Symbols for the AdsSymbolServer with the help of an [ServerSymbolFactory](#) [▶ 3242].

Reference

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [▶ 3230]

6.14.2.1 IAddSymbolicInformation Methods

The [IAddSymbolicInformation](#) [▶ 3234] type exposes the following members.

Methods

	Name	Description
	AddDataArea [▶ 3236]	Adds a DataArea [▶ 3230]to the symbolic information.
	AddSymbol(String, String, String) [▶ 3237]	Adds and registers a ISymbol [▶ 2691].

	Name	Description
	AddSymbol(String, IDataTypes, DataArea) [▶ 3238]	Adds and registers a symbol
	AddType [▶ 3238]	Adds and registers a DataType [▶ 1721]
	ClearDataAreas [▶ 3239]	Removes all DataAreas.
	ClearSymbols [▶ 3239]	Removes all symbols.
	ClearTypes [▶ 3240]	Removes all DataTypes.
	RemoveDataArea [▶ 3240]	Removes a DataArea [▶ 3230] from the symbolic information.
	RemoveSymbol(ISymbol) [▶ 3241]	Removes the ISymbol [▶ 2691] from the Symbolic information.
	RemoveSymbol(String) [▶ 3241]	Removes the ISymbol [▶ 2691] from the Symbolic information
	RemoveType [▶ 3242]	Removes the DataType [▶ 1721] from the symbolic information.

Reference

[IAddSymbolicInformation Interface](#) [[▶ 3234](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.2.1.1 IAddSymbolicInformation.AddDataArea Method

Adds a [DataArea](#) [[▶ 3230](#)] to the symbolic information.

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[▶ 3230](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IAddSymbolicInformation AddDataArea (
    DataArea area
)
```

Parameters

area	Type: TwinCAT.Ads.Server.TypeSystem.DataArea [▶ 3230] The area.
------	--

Return Value

Type: [IAddSymbolicInformation](#) [[▶ 3234](#)]

The self-referential [IAddSymbolicInformation](#) [[▶ 3234](#)] object to support method chaining/cascading.

Reference

[IAddSymbolicInformation Interface](#) [[▶ 3234](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.2.1.2 IAddSymbolicInformation.AddSymbol Method

Overload List

	Name	Description
	AddSymbol(String, String, String) [▶ 3237]	Adds and registers a ISymbol [▶ 2691].
	AddSymbol(String, IDataType, DataArea) [▶ 3238]	Adds and registers a symbol

Reference

[IAddSymbolicInformation Interface](#) [[▶ 3234](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.2.1.2.1 IAddSymbolicInformation.AddSymbol Method (String, String, String)

Adds and registers a [ISymbol](#) [[▶ 2691](#)].

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[▶ 3230](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IAddSymbolicInformation AddSymbol(
    string instancePath,
    string dataType,
    string dataArea
)
```

Parameters

instancePath	Type: System.String The instancePath.
dataType	Type: System.String The name of the already registered DataType
dataArea	Type: System.String The data area.

Return Value

Type: [IAddSymbolicInformation](#) [[▶ 3234](#)]

The self-referential [IAddSymbolicInformation](#) [[▶ 3234](#)] object to support method chaining/cascading.

Remarks

The address in the DataArea will be calculated automatically (indexGroup, indexOffset).

Reference

[IAddSymbolicInformation Interface](#) [[▶ 3234](#)]

[AddSymbol Overload](#) [[▶ 3237](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.2.1.2 IAddSymbolicInformation.AddSymbol Method (String, IDataType, DataArea)

Adds and registers a symbol

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[▶ 3230](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IAddSymbolicInformation AddSymbol(
    string instancePath,
    IDataType dataType,
    DataArea dataArea
)
```

Parameters

instancePath	Type: System.String The InstancePath (FullPath) to the symbol.
dataType	Type: TwinCAT.TypeSystem.IDataType [▶ 2475] The (already registered) DataType
dataArea	Type: TwinCAT.Ads.Server.TypeSystem.DataArea [▶ 3230] The data area.

Return Value

Type: [IAddSymbolicInformation](#) [[▶ 3234](#)]

The self-referential [IAddSymbolicInformation](#) [[▶ 3234](#)] object to support method chaining/cascading.

Remarks

The address in the DataArea will be calculated automatically (indexGroup, indexOffset).

Reference

[IAddSymbolicInformation Interface](#) [[▶ 3234](#)]

[AddSymbol Overload](#) [[▶ 3237](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.2.1.3 IAddSymbolicInformation.AddType Method

Adds and registers a [DataType](#) [[▶ 1721](#)]

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[▶ 3230](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IAddSymbolicInformation AddType(
    DataType type
)
```

Parameters

type	Type: TwinCAT.Ads.TypeSystem.DataType [1721] The type.
------	---

Return Value

Type: [IAddSymbolicInformation](#) [[3234](#)]

The self-referential [IAddSymbolicInformation](#) [[3234](#)] object to support method chaining/cascading.

Reference

[IAddSymbolicInformation Interface](#) [[3234](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[3230](#)]

6.14.2.1.4 IAddSymbolicInformation.ClearDataAreas Method

Removes all DataAreas.

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[3230](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IAddSymbolicInformation ClearDataAreas()
```

Return Value

Type: [IAddSymbolicInformation](#) [[3234](#)]

The self-referential [IAddSymbolicInformation](#) [[3234](#)] object to support method chaining/cascading.

Reference

[IAddSymbolicInformation Interface](#) [[3234](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[3230](#)]

6.14.2.1.5 IAddSymbolicInformation.ClearSymbols Method

Removes all symbols.

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[3230](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IAddSymbolicInformation ClearSymbols()
```

Return Value

Type: [IAddSymbolicInformation](#) [[3234](#)]

The self-referential [IAddSymbolicInformation](#) [[3234](#)] object to support method chaining/cascading.

Reference

[IAddSymbolicInformation Interface](#) [[3234](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace \[► 3230\]](#)

6.14.2.1.6 IAddSymbolicInformation.ClearTypes Method

Removes all DataTypes.

Namespace: [TwinCAT.Ads.Server.TypeSystem \[► 3230\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IAddSymbolicInformation ClearTypes()
```

Return Value

Type: [IAddSymbolicInformation \[► 3234\]](#)

The self-referential [IAddSymbolicInformation \[► 3234\]](#) object to support method chaining/cascading.

Reference

[IAddSymbolicInformation Interface \[► 3234\]](#)

[TwinCAT.Ads.Server.TypeSystem Namespace \[► 3230\]](#)

6.14.2.1.7 IAddSymbolicInformation.RemoveDataArea Method

Removes a [DataArea \[► 3230\]](#) from the symbolic information.

Namespace: [TwinCAT.Ads.Server.TypeSystem \[► 3230\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IAddSymbolicInformation RemoveDataArea(
    DataArea area
)
```

Parameters

area	Type: TwinCAT.Ads.Server.TypeSystem.DataArea [► 3230] The area.
------	--

Return Value

Type: [IAddSymbolicInformation \[► 3234\]](#)

The self-referential [IAddSymbolicInformation \[► 3234\]](#) object to support method chaining/cascading.

Reference

[IAddSymbolicInformation Interface \[► 3234\]](#)

[TwinCAT.Ads.Server.TypeSystem Namespace \[► 3230\]](#)

6.14.2.1.8 IAddSymbolicInformation.RemoveSymbol Method

Overload List

	Name	Description
	RemoveSymbol(ISymbol) [▸ 3241]	Removes the ISymbol [▸ 2691] from the Symbolic information.
	RemoveSymbol(String) [▸ 3241]	Removes the ISymbol [▸ 2691] from the Symbolic information

Reference

[IAddSymbolicInformation Interface \[▸ 3234\]](#)

[TwinCAT.Ads.Server.TypeSystem Namespace \[▸ 3230\]](#)

6.14.2.1.8.1 IAddSymbolicInformation.RemoveSymbol Method (ISymbol)

Removes the [ISymbol \[▸ 2691\]](#) from the Symbolic information.

Namespace: [TwinCAT.Ads.Server.TypeSystem \[▸ 3230\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IAddSymbolicInformation RemoveSymbol (
    ISymbol symbol
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▸ 2691] The symbol.
--------	--

Return Value

Type: [IAddSymbolicInformation \[▸ 3234\]](#)

The self-referential [IAddSymbolicInformation \[▸ 3234\]](#) object to support method chaining/cascading.

Reference

[IAddSymbolicInformation Interface \[▸ 3234\]](#)

[RemoveSymbol Overload \[▸ 3241\]](#)

[TwinCAT.Ads.Server.TypeSystem Namespace \[▸ 3230\]](#)

6.14.2.1.8.2 IAddSymbolicInformation.RemoveSymbol Method (String)

Removes the [ISymbol \[▸ 2691\]](#) from the Symbolic information

Namespace: [TwinCAT.Ads.Server.TypeSystem \[▸ 3230\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IAddSymbolicInformation RemoveSymbol (
    string instancePath
)
```

Parameters

instancePath	Type: System.String The instance path.
--------------	---

Return Value

Type: [IAddSymbolicInformation](#) [[▶ 3234](#)]

The self-referential [IAddSymbolicInformation](#) [[▶ 3234](#)] object to support method chaining/cascading.

Reference

[IAddSymbolicInformation Interface](#) [[▶ 3234](#)]

[RemoveSymbol Overload](#) [[▶ 3241](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.2.1.9 IAddSymbolicInformation.RemoveType Method

Removes the [DataType](#) [[▶ 1721](#)] from the symbolic information.

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[▶ 3230](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
IAddSymbolicInformation RemoveType (
    DataType type
)
```

Parameters

type	Type: TwinCAT.Ads.TypeSystem.DataType [▶ 1721] The type.
------	---

Return Value

Type: [IAddSymbolicInformation](#) [[▶ 3234](#)]

The self-referential [IAddSymbolicInformation](#) [[▶ 3234](#)] object to support method chaining/cascading.

Reference

[IAddSymbolicInformation Interface](#) [[▶ 3234](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.3 ServerSymbolFactory Class

Class ServerSymbolFactory.

Inheritance Hierarchy

System.Object

TwinCAT.Ads.Server.TypeSystem.ServerSymbolFactory

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [▶ 3230]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public class ServerSymbolFactory : IAddSymbolicInformation
```

The ServerSymbolFactory type exposes the following members.

Constructors

	Name	Description
	ServerSymbolFactory [▶ 3244]	Initializes a new instance of the ServerSymbolFactory class.

Methods

	Name	Description
	AddDataArea [▶ 3245]	Adds a DataArea [▶ 3230] to the symbolic information.
	AddSymbol(String, String, String) [▶ 3246]	Adds and registers a ISymbol [▶ 2691].
	AddSymbol(String, IDataTypes, DataArea) [▶ 3247]	Adds and registers a symbol
	AddType [▶ 3248]	Adds and registers a DataType [▶ 1721]
	ClearDataAreas [▶ 3249]	Removes all DataAreas.
	ClearSymbols [▶ 3249]	Removes all symbols.
	ClearTypes [▶ 3250]	Removes all DataTypes.
	CreateSymbol(String, IDataTypes, ISymbol, Int32) [▶ 3251]	Creates the symbol.
	CreateSymbol(String, String, IDataTypes, UInt32, UInt32) [▶ 3251]	Creates the symbol.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)

	Name	Description
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	RemoveDataArea [▶ 3252]	Removes a DataArea [▶ 3230] from the symbolic information.
	RemoveSymbol(ISymbol) [▶ 3253]	Removes the ISymbol [▶ 2691] from the Symbolic information.
	RemoveSymbol(String) [▶ 3254]	Removes the ISymbol [▶ 2691] from the Symbolic information
	RemoveType [▶ 3254]	Removes the DataType [▶ 1721] from the symbolic information.
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Remarks

Factory class supporting creating DataAreas, DataTypes and Symbols.

Reference

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [\[▶ 3230\]](#)

6.14.3.1 ServerSymbolFactory Constructor

Initializes a new instance of the [ServerSymbolFactory](#) [\[▶ 3242\]](#) class.

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [\[▶ 3230\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ServerSymbolFactory(
    AdsSymbolicServer server
)
```

Parameters

server	Type: TwinCAT.Ads.Server.AdsSymbolicServer [▶ 1468] The server.
--------	--

Reference

[ServerSymbolFactory Class](#) [\[▶ 3242\]](#)

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [\[▶ 3230\]](#)

6.14.3.2 ServerSymbolFactory Methods

The [ServerSymbolFactory](#) [\[▶ 3242\]](#) type exposes the following members.

Methods

	Name	Description
	AddDataArea [▶ 3245]	Adds a DataArea [▶ 3230] to the symbolic information.

	Name	Description
	AddSymbol(String, String, String) [▶ 3246]	Adds and registers a ISymbol [▶ 2691].
	AddSymbol(String, IDataType, DataArea) [▶ 3247]	Adds and registers a symbol
	AddType [▶ 3248]	Adds and registers a DataType [▶ 1721]
	ClearDataAreas [▶ 3249]	Removes all DataAreas .
	ClearSymbols [▶ 3249]	Removes all symbols.
	ClearTypes [▶ 3250]	Removes all DataTypes .
	CreateSymbol(String, IDataType, ISymbol, Int32) [▶ 3251]	Creates the symbol.
	CreateSymbol(String, String, IDataType, UInt32, UInt32) [▶ 3251]	Creates the symbol.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	RemoveDataArea [▶ 3252]	Removes a DataArea [▶ 3230] from the symbolic information.
	RemoveSymbol(ISymbol) [▶ 3253]	Removes the ISymbol [▶ 2691] from the Symbolic information.
	RemoveSymbol(String) [▶ 3254]	Removes the ISymbol [▶ 2691] from the Symbolic information
	RemoveType [▶ 3254]	Removes the DataType [▶ 1721] from the symbolic information.
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Reference

[ServerSymbolFactory Class](#) [[▶ 3242](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.3.2.1 ServerSymbolFactory.AddDataArea Method

Adds a [DataArea](#) [[▶ 3230](#)]to the symbolic information.

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [► 3230]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IAddSymbolicInformation AddDataArea(
    DataArea area
)
```

Parameters

area	Type: TwinCAT.Ads.Server.TypeSystem.DataArea [► 3230] The area.
------	--

Return Value

Type: [IAddSymbolicInformation](#) [► 3234]

The self-referential [IAddSymbolicInformation](#) [► 3234] object to support method chaining/cascading.

Implements

[IAddSymbolicInformation.AddDataArea\(DataArea\)](#) [► 3236]

Reference

[ServerSymbolFactory Class](#) [► 3242]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [► 3230]

6.14.3.2.2 ServerSymbolFactory.AddSymbol Method

Overload List

	Name	Description
	AddSymbol(String, String, String) [► 3246]	Adds and registers a ISymbol [► 2691].
	AddSymbol(String, IDataType, DataArea) [► 3247]	Adds and registers a symbol

Reference

[ServerSymbolFactory Class](#) [► 3242]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [► 3230]

6.14.3.2.2.1 ServerSymbolFactory.AddSymbol Method (String, String, String)

Adds and registers a [ISymbol](#) [► 2691].

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [► 3230]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IAddSymbolicInformation AddSymbol(
    string instancePath,
    string dataType,
    string dataArea
)
```

Parameters

instancePath	Type: System.String The instancePath.
dataType	Type: System.String The name of the already registered DataType
dataArea	Type: System.String The data area.

Return Value

Type: [IAddSymbolicInformation](#) [[▶ 3234](#)]
The self-referential [IAddSymbolicInformation](#) [[▶ 3234](#)] object to support method chaining/cascading.

Implements

[IAddSymbolicInformation.AddSymbol\(String, String, String\)](#) [[▶ 3237](#)]

Exceptions

Exception	Condition
ArgumentOutOfRangeException	dataArea - Cannot find DataArea'{dataArea}'
DataTypeException [▶ 2121]	DataType '{dataType}' not found!

Remarks

The address in the DataArea will be calculated automatically (indexGroup, indexOffset).

Reference

- [ServerSymbolFactory Class](#) [[▶ 3242](#)]
- [AddSymbol Overload](#) [[▶ 3246](#)]
- [TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.3.2.2 ServerSymbolFactory.AddSymbol Method (String, IDatatype, DataArea)

Adds and registers a symbol

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[▶ 3230](#)]
Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IAddSymbolicInformation AddSymbol(
    string instancePath,
    IDataType dataType,
    DataArea dataArea
)
```

Parameters

instancePath	Type: System.String The InstancePath (FullPath) to the symbol.
dataType	Type: TwinCAT.TypeSystem.IDataType [▶ 2475] The (already registered) DataType
dataArea	Type: TwinCAT.Ads.Server.TypeSystem.DataArea [▶ 3230] The data area.

Return Value

Type: [IAddSymbolicInformation](#) [[▶ 3234](#)]

The self-referential [IAddSymbolicInformation](#) [[▶ 3234](#)] object to support method chaining/cascading.

Implements

[IAddSymbolicInformation.AddSymbol\(String, IDataType, DataArea\)](#) [[▶ 3238](#)]

Remarks

The address in the DataArea will be calculated automatically (indexGroup, indexOffset).

Reference

[ServerSymbolFactory Class](#) [[▶ 3242](#)]

[AddSymbol Overload](#) [[▶ 3246](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.3.2.3 ServerSymbolFactory.AddType Method

Adds and registers a [DataType](#) [[▶ 1721](#)]

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[▶ 3230](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IAddSymbolicInformation AddType(
    DataType type
)
```

Parameters

type	Type: TwinCAT.Ads.TypeSystem.DataType [▶ 1721] The type.
------	---

Return Value

Type: [IAddSymbolicInformation](#) [[▶ 3234](#)]

The self-referential [IAddSymbolicInformation](#) [[▶ 3234](#)] object to support method chaining/cascading.

Implements

[IAddSymbolicInformation.AddType\(DataType\)](#) [[▶ 3238](#)]

Reference

[ServerSymbolFactory Class](#) [[▶ 3242](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.3.2.4 ServerSymbolFactory.ClearDataAreas Method

Removes all DataAreas.

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[▶ 3230](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IAddSymbolicInformation ClearDataAreas()
```

Return Value

Type: [IAddSymbolicInformation](#) [[▶ 3234](#)]

The self-referential [IAddSymbolicInformation](#) [[▶ 3234](#)] object to support method chaining/cascading.

Implements

[IAddSymbolicInformation.ClearDataAreas.](#) [[▶ 3239](#)]

Reference

[ServerSymbolFactory Class](#) [[▶ 3242](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.3.2.5 ServerSymbolFactory.ClearSymbols Method

Removes all symbols.

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[▶ 3230](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IAddSymbolicInformation ClearSymbols()
```

Return Value

Type: [IAddSymbolicInformation](#) [[▶ 3234](#)]

The self-referential [IAddSymbolicInformation](#) [[▶ 3234](#)] object to support method chaining/cascading.

Implements

[IAddSymbolicInformation.ClearSymbols. \[▸ 3239\]](#)

Reference

[ServerSymbolFactory Class \[▸ 3242\]](#)

[TwinCAT.Ads.Server.TypeSystem Namespace \[▸ 3230\]](#)

6.14.3.2.6 ServerSymbolFactory.ClearTypes Method

Removes all DataTypes.

Namespace: [TwinCAT.Ads.Server.TypeSystem \[▸ 3230\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public IAddSymbolicInformation ClearTypes ()
```

Return Value

Type: [IAddSymbolicInformation \[▸ 3234\]](#)

The self-referential [IAddSymbolicInformation \[▸ 3234\]](#) object to support method chaining/cascading.

Implements

[IAddSymbolicInformation.ClearTypes. \[▸ 3240\]](#)

Reference

[ServerSymbolFactory Class \[▸ 3242\]](#)

[TwinCAT.Ads.Server.TypeSystem Namespace \[▸ 3230\]](#)

6.14.3.2.7 ServerSymbolFactory.CreateSymbol Method**Overload List**

	Name	Description
	CreateSymbol(String, IDataTypes, ISymbol, Int32) [▸ 3251]	Creates the symbol.
	CreateSymbol(String, String, IDataTypes, UInt32, UInt32) [▸ 3251]	Creates the symbol.

Reference

[ServerSymbolFactory Class \[▸ 3242\]](#)

[TwinCAT.Ads.Server.TypeSystem Namespace \[► 3230\]](#)

6.14.3.2.7.1 ServerSymbolFactory.CreateSymbol Method (String, IDataTypes, ISymbol, Int32)

Creates the symbol.

Namespace: [TwinCAT.Ads.Server.TypeSystem \[► 3230\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Symbol CreateSymbol(
    string instanceName,
    IDataTypes dataType,
    ISymbol parent,
    int fieldOffset
)
```

Parameters

instanceName	Type: System.String Name of the instance.
dataType	Type: TwinCAT.TypeSystem.IDataTypes [► 2475] Type of the data.
parent	Type: TwinCAT.TypeSystem.ISymbol [► 2691] The parent.
fieldOffset	Type: System.Int32 The field offset.

Return Value

Type: [Symbol \[► 1863\]](#)
Symbol.

Exceptions

Exception	Condition
ArgumentNullException	instanceName
ArgumentNullException	dataType
NotSupportedException	

Reference

[ServerSymbolFactory Class \[► 3242\]](#)

[CreateSymbol Overload \[► 3250\]](#)

[TwinCAT.Ads.Server.TypeSystem Namespace \[► 3230\]](#)

6.14.3.2.7.2 ServerSymbolFactory.CreateSymbol Method (String, String, IDataTypes, UInt32, UInt32)

Creates the symbol.

Namespace: [TwinCAT.Ads.Server.TypeSystem \[► 3230\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public Symbol CreateSymbol(
    string instanceName,
    string instancePath,
    IDataType dataType,
    uint indexGroup,
    uint indexOffset
)
```

Parameters

instanceName	Type: System.String Name of the instance.
instancePath	Type: System.String The instance path.
dataType	Type: TwinCAT.TypeSystem.IDataType [▶ 2475] Type of the data.
indexGroup	Type: System.UInt32 The index group.
indexOffset	Type: System.UInt32 The index offset.

Return Value

Type: [Symbol](#) [[▶ 1863](#)]
Symbol.

Exceptions

Exception	Condition
ArgumentNullException	instanceName
ArgumentNullException	dataType
NotSupportedException	Data Type is not supported!

Reference

[ServerSymbolFactory Class](#) [[▶ 3242](#)]

[CreateSymbol Overload](#) [[▶ 3250](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.3.2.8 ServerSymbolFactory.RemoveDataArea Method

Removes a [DataArea](#) [[▶ 3230](#)] from the symbolic information.

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[▶ 3230](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IAddSymbolicInformation RemoveDataArea(
    DataArea area
)
```

Parameters

area	Type: TwinCAT.Ads.Server.TypeSystem.DataArea [▸ 3230] The area.
------	--

Return Value

Type: [IAddSymbolicInformation \[▸ 3234\]](#)
The self-referential [IAddSymbolicInformation \[▸ 3234\]](#) object to support method chaining/cascading.

Implements

[IAddSymbolicInformation.RemoveDataArea\(DataArea\) \[▸ 3240\]](#)

Reference

[ServerSymbolFactory Class \[▸ 3242\]](#)

[TwinCAT.Ads.Server.TypeSystem Namespace \[▸ 3230\]](#)

6.14.3.2.9 ServerSymbolFactory.RemoveSymbol Method

Overload List

	Name	Description
	RemoveSymbol(ISymbol) [▸ 3253]	Removes the ISymbol [▸ 2691] from the Symbolic information.
	RemoveSymbol(String) [▸ 3254]	Removes the ISymbol [▸ 2691] from the Symbolic information

Reference

[ServerSymbolFactory Class \[▸ 3242\]](#)

[TwinCAT.Ads.Server.TypeSystem Namespace \[▸ 3230\]](#)

6.14.3.2.9.1 ServerSymbolFactory.RemoveSymbol Method (ISymbol)

Removes the [ISymbol \[▸ 2691\]](#) from the Symbolic information.

Namespace: [TwinCAT.Ads.Server.TypeSystem \[▸ 3230\]](#)

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IAddSymbolicInformation RemoveSymbol(
    ISymbol symbol
)
```

Parameters

symbol	Type: TwinCAT.TypeSystem.ISymbol [▸ 2691] The symbol.
--------	--

Return Value

Type: [IAddSymbolicInformation](#) [[▶ 3234](#)]

The self-referential [IAddSymbolicInformation](#) [[▶ 3234](#)] object to support method chaining/cascading.

Implements

[IAddSymbolicInformation.RemoveSymbol\(ISymbol\)](#) [[▶ 3241](#)]

Reference

[ServerSymbolFactory Class](#) [[▶ 3242](#)]

[RemoveSymbol Overload](#) [[▶ 3253](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.3.2.9.2 ServerSymbolFactory.RemoveSymbol Method (String)

Removes the [ISymbol](#) [[▶ 2691](#)] from the Symbolic information

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[▶ 3230](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public IAddSymbolicInformation RemoveSymbol(
    string instancePath
)
```

Parameters

instancePath	Type: System.String The instance path.
--------------	---

Return Value

Type: [IAddSymbolicInformation](#) [[▶ 3234](#)]

The self-referential [IAddSymbolicInformation](#) [[▶ 3234](#)] object to support method chaining/cascading.

Implements

[IAddSymbolicInformation.RemoveSymbol\(String\)](#) [[▶ 3241](#)]

Reference

[ServerSymbolFactory Class](#) [[▶ 3242](#)]

[RemoveSymbol Overload](#) [[▶ 3253](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.14.3.2.10 ServerSymbolFactory.RemoveType Method

Removes the [DataType](#) [[▶ 1721](#)] from the symbolic information.

Namespace: [TwinCAT.Ads.Server.TypeSystem](#) [[▶ 3230](#)]

Assembly: TwinCAT.Ads.SymbolicServer (in TwinCAT.Ads.SymbolicServer.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public IAddSymbolicInformation RemoveType(
    DataType type
)
```

Parameters

type	Type: TwinCAT.Ads.TypeSystem.DataType [▶ 1721] The type.
------	---

Return Value

Type: [IAddSymbolicInformation](#) [[▶ 3234](#)]
The self-referential [IAddSymbolicInformation](#) [[▶ 3234](#)] object to support method chaining/cascading.

Implements

[IAddSymbolicInformation.RemoveType\(DataType\)](#) [[▶ 3242](#)]

Reference

[ServerSymbolFactory Class](#) [[▶ 3242](#)]

[TwinCAT.Ads.Server.TypeSystem Namespace](#) [[▶ 3230](#)]

6.15 TwinCAT.Router Namespace

Classes

	Class	Description
	RouterPort [▶ 3255]	Class represents a Router Port
	RouterPortInfo [▶ 3264]	Class RouterPortInfo.

6.15.1 RouterPort Class

Class represents a Router Port

Inheritance Hierarchy

System.Object
[TwinCAT.Router.RouterPortInfo](#) [[▶ 3264](#)]
 TwinCAT.Router.RouterPort

Namespace: [TwinCAT.Router](#) [[▶ 3255](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public class RouterPort : RouterPortInfo
```

The RouterPort type exposes the following members.

Constructors

	Name	Description
	RouterPort . [▶ 3257]	Constructs the RouterPort object
	RouterPort(AmsRegi sterPort) . [▶ 3257]	Constructs the RouterPort object

Properties

	Name	Description
	Attributes [▶ 3258]	Gets additional attributes of the Port
	HasSymbols [▶ 3259]	Gets, whether the Port contains Symbol informations
	IsAdsSecurePort [▶ 3259]	Gets a value indicating whether this instance is ads secure port.
	IsClientPort [▶ 3259]	Gets, whether the port is an Client port
	IsFixedPort [▶ 3260]	Gets, whether the Port is fixed
	IsServerPort [▶ 3260]	Gets, whether the port is an Server port
	IsSynchronized [▶ 3260]	Gets, whether the Port is Synchronized
	Name [▶ 3267]	Gets the Name of the Port (Inherited from RouterPortInfo [▶ 3264].)
	PortID [▶ 3267]	Gets the PortNumber (Inherited from RouterPortInfo [▶ 3264].)
	PortType [▶ 3261]	Gets the type of the Port
	QueueSize [▶ 3261]	Gets the current Queue size

Methods

	Name	Description
	Equals [▶ 3268]	Equals (Inherited from RouterPortInfo [▶ 3264].)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode [▶ 3269]	Gets the HashCode of the Address (Inherited from RouterPortInfo [▶ 3264].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString [▶ 3262]	Returns a String that represents this instance. (Overrides RouterPortInfo.ToString . [▶ 3269].)

Fields

	Name	Description
	fixedPort [▶ 3263]	Fixed port
	port [▶ 3272]	Port Number (Inherited from RouterPortInfo [▶ 3264].)
	portAttributes [▶ 3263]	Port Attributes.
	portName [▶ 3272]	Port name (Inherited from RouterPortInfo [▶ 3264].)
	portQueueSize [▶ 3264]	Port Queue size.
	portType [▶ 3264]	Port Type

Reference

[TwinCAT.Router Namespace \[▶ 3255\]](#)

6.15.1.1 RouterPort Constructor

Overload List

	Name	Description
	RouterPort. [▶ 3257]	Constructs the RouterPort [▶ 3255] object
	RouterPort(AmsRegisterPort) [▶ 3257]	Constructs the RouterPort [▶ 3255] object

Reference

[RouterPort Class \[▶ 3255\]](#)

[TwinCAT.Router Namespace \[▶ 3255\]](#)

6.15.1.1.1 RouterPort Constructor

Constructs the [RouterPort \[▶ 3255\]](#) object

Namespace: [TwinCAT.Router \[▶ 3255\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected RouterPort ()
```

Reference

[RouterPort Class \[▶ 3255\]](#)

[RouterPort Overload \[▶ 3257\]](#)

[TwinCAT.Router Namespace \[▶ 3255\]](#)

6.15.1.1.2 RouterPort Constructor (AmsRegisterPort)

Constructs the [RouterPort \[▶ 3255\]](#) object

Namespace: [TwinCAT.Router \[▶ 3255\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RouterPort (
    AmsRegisterPort intPort
)
```

Parameters

intPort	Type: AmsRegisterPort Internal port representation (Marshal object)
---------	--

Reference[RouterPort Class \[► 3255\]](#)[RouterPort Overload \[► 3257\]](#)[TwinCAT.Router Namespace \[► 3255\]](#)**6.15.1.2 RouterPort Properties**

The [RouterPort \[► 3255\]](#) type exposes the following members.

Properties

	Name	Description
	Attributes [► 3258]	Gets additional attributes of the Port
	HasSymbols [► 3259]	Gets, whether the Port contains Symbol informations
	IsAdsSecurePort [► 3259]	Gets a value indicating whether this instance is ads secure port.
	IsClientPort [► 3259]	Gets, whether the port is an Client port
	IsFixedPort [► 3260]	Gets, whether the Port is fixed
	IsServerPort [► 3260]	Gets, whether the port is an Server port
	IsSynchronized [► 3260]	Gets, whether the Port is Synchronized
	Name [► 3267]	Gets the Name of the Port (Inherited from RouterPortInfo [► 3264].)
	PortID [► 3267]	Gets the PortNumber (Inherited from RouterPortInfo [► 3264].)
	PortType [► 3261]	Gets the type of the Port
	QueueSize [► 3261]	Gets the current Queue size

Reference[RouterPort Class \[► 3255\]](#)[TwinCAT.Router Namespace \[► 3255\]](#)**6.15.1.2.1 RouterPort.Attributes Property**

Gets additional attributes of the Port

Namespace: [TwinCAT.Router \[► 3255\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public AmsPortAttributes Attributes { get; }
```

Property Value

Type: AmsPortAttributes

Reference[RouterPort Class \[► 3255\]](#)[TwinCAT.Router Namespace \[► 3255\]](#)

6.15.1.2.2 RouterPort.HasSymbols Property

Gets, whether the Port contains Symbol informations

Namespace: [TwinCAT.Router](#) [► 3255]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool HasSymbols { get; }
```

Property Value

Type: Boolean

Reference

[RouterPort Class](#) [► 3255]

[TwinCAT.Router Namespace](#) [► 3255]

6.15.1.2.3 RouterPort.IsAdsSecurePort Property

Gets a value indicating whether this instance is ads secure port.

Namespace: [TwinCAT.Router](#) [► 3255]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsAdsSecurePort { get; }
```

Property Value

Type: Boolean

true if this instance is ads secure port; otherwise, false.

Reference

[RouterPort Class](#) [► 3255]

[TwinCAT.Router Namespace](#) [► 3255]

6.15.1.2.4 RouterPort.IsClientPort Property

Gets, whether the port is an Client port

Namespace: [TwinCAT.Router](#) [► 3255]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsClientPort { get; }
```

Property Value

Type: Boolean

Reference

[RouterPort Class \[► 3255\]](#)

[TwinCAT.Router Namespace \[► 3255\]](#)

6.15.1.2.5 RouterPort.IsFixedPort Property

Gets, whether the Port is fixed

Namespace: [TwinCAT.Router \[► 3255\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsFixedPort { get; }
```

Property Value

Type: Boolean

Reference

[RouterPort Class \[► 3255\]](#)

[TwinCAT.Router Namespace \[► 3255\]](#)

6.15.1.2.6 RouterPort.IsServerPort Property

Gets, whether the port is an Server port

Namespace: [TwinCAT.Router \[► 3255\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsServerPort { get; }
```

Property Value

Type: Boolean

Reference

[RouterPort Class \[► 3255\]](#)

[TwinCAT.Router Namespace \[► 3255\]](#)

6.15.1.2.7 RouterPort.IsSynchronized Property

Gets, whether the Port is Synchronized

Namespace: [TwinCAT.Router \[► 3255\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public bool IsSynchronized { get; }
```

Property Value

Type: Boolean

Reference

[RouterPort Class \[► 3255\]](#)

[TwinCAT.Router Namespace \[► 3255\]](#)

6.15.1.2.8 RouterPort.PortType Property

Gets the type of the Port

Namespace: [TwinCAT.Router \[► 3255\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AmsPortType PortType { get; }
```

Property Value

Type: AmsPortType

Reference

[RouterPort Class \[► 3255\]](#)

[TwinCAT.Router Namespace \[► 3255\]](#)

6.15.1.2.9 RouterPort.QueueSize Property

Gets the current Queue size

Namespace: [TwinCAT.Router \[► 3255\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int QueueSize { get; }
```

Property Value

Type: Int32

Reference

[RouterPort Class \[► 3255\]](#)

[TwinCAT.Router Namespace \[► 3255\]](#)

6.15.1.3 RouterPort Methods

The [RouterPort](#) [[▶ 3255](#)] type exposes the following members.

Methods

	Name	Description
	Equals [▶ 3268]	Equals (Inherited from RouterPortInfo [▶ 3264].)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode [▶ 3269]	Gets the GetHashCode of the Address (Inherited from RouterPortInfo [▶ 3264].)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString [▶ 3262]	Returns a String that represents this instance. (Overrides RouterPortInfo.ToString . [▶ 3269].)

Reference

[RouterPort Class](#) [[▶ 3255](#)]

[TwinCAT.Router Namespace](#) [[▶ 3255](#)]

6.15.1.3.1 RouterPort.ToString Method

Returns a String that represents this instance.

Namespace: [TwinCAT.Router](#) [[▶ 3255](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```

Return Value

Type: String

A String that represents this instance.

Reference

[RouterPort Class](#) [[▶ 3255](#)]

[TwinCAT.Router Namespace](#) [[▶ 3255](#)]

6.15.1.4 RouterPort Fields

The [RouterPort](#) [[▶ 3255](#)] type exposes the following members.

Fields

	Name	Description
	fixedPort [▶ 3263]	Fixed port
	port [▶ 3272]	Port Number (Inherited from RouterPortInfo [▶ 3264].)

	Name	Description
	portAttributes [▶ 3263]	Port Attributes.
	portName [▶ 3272]	Port name (Inherited from RouterPortInfo [▶ 3264].)
	portQueueSize [▶ 3264]	Port Queue size.
	portType [▶ 3264]	Port Type

Reference

[RouterPort Class](#) [▶ 3255]

[TwinCAT.Router Namespace](#) [▶ 3255]

6.15.1.4.1 RouterPort.fixedPort Field

Fixed port

Namespace: [TwinCAT.Router](#) [▶ 3255]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected bool fixedPort
```

Field Value

Type: Boolean

Reference

[RouterPort Class](#) [▶ 3255]

[TwinCAT.Router Namespace](#) [▶ 3255]

6.15.1.4.2 RouterPort.portAttributes Field

Port Attributes.

Namespace: [TwinCAT.Router](#) [▶ 3255]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected AmsPortAttributes portAttributes
```

Field Value

Type: AmsPortAttributes

Reference

[RouterPort Class](#) [▶ 3255]

[TwinCAT.Router Namespace](#) [▶ 3255]

6.15.1.4.3 RouterPort.portQueueSize Field

Port Queue size.

Namespace: [TwinCAT.Router](#) [▶ 3255]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected int portQueueSize
```

Field Value

Type: Int32

Reference

[RouterPort Class](#) [▶ 3255]

[TwinCAT.Router Namespace](#) [▶ 3255]

6.15.1.4.4 RouterPort.portType Field

Port Type

Namespace: [TwinCAT.Router](#) [▶ 3255]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected AmsPortType portType
```

Field Value

Type: AmsPortType

Reference

[RouterPort Class](#) [▶ 3255]

[TwinCAT.Router Namespace](#) [▶ 3255]

6.15.2 RouterPortInfo Class

Class RouterPortInfo.

Inheritance Hierarchy

System.Object

 TwinCAT.Router.RouterPortInfo

[TwinCAT.Router.RouterPort](#) [▶ 3255]

Namespace: [TwinCAT.Router](#) [▶ 3255]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public class RouterPortInfo
```


The RouterPortInfo type exposes the following members.

Constructors

	Name	Description
	RouterPortInfo. [▶ 3266]	Initializes a new instance of the RouterPortInfo class.
	RouterPortInfo(Int32, String) [▶ 3266]	Initializes a new instance of the RouterPortInfo class.

Properties

	Name	Description
	Name [▶ 3267]	Gets the Name of the Port
	PortID [▶ 3267]	Gets the PortNumber

Methods

	Name	Description
	Equals [▶ 3268]	Equals (Overrides Object.Equals(Object).)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode [▶ 3269]	Gets the GetHashCode of the Address (Overrides Object.GetHashCode..)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString [▶ 3269]	Returns a String that represents this instance. (Overrides Object.ToString..)
	TryParse [▶ 3269]	Tries to parse the RouterPortInfo object.

Operators

	Name	Description
	Equality [▶ 3270]	Operator==
	Inequality [▶ 3271]	Operator!=

Fields

	Name	Description
	port [▶ 3272]	Port Number
	portName [▶ 3272]	Port name

Reference

[TwinCAT.Router Namespace](#) [\[▶ 3255\]](#)

6.15.2.1 RouterPortInfo Constructor

Overload List

	Name	Description
	RouterPortInfo. [► 3266]	Initializes a new instance of the RouterPortInfo [► 3264] class.
	RouterPortInfo(Int32, String) [► 3266]	Initializes a new instance of the RouterPortInfo [► 3264] class.

Reference

[RouterPortInfo Class \[► 3264\]](#)

[TwinCAT.Router Namespace \[► 3255\]](#)

6.15.2.1.1 RouterPortInfo Constructor

Initializes a new instance of the [RouterPortInfo \[► 3264\]](#) class.

Namespace: [TwinCAT.Router \[► 3255\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected RouterPortInfo()
```

Reference

[RouterPortInfo Class \[► 3264\]](#)

[RouterPortInfo Overload \[► 3266\]](#)

[TwinCAT.Router Namespace \[► 3255\]](#)

6.15.2.1.2 RouterPortInfo Constructor (Int32, String)

Initializes a new instance of the [RouterPortInfo \[► 3264\]](#) class.

Namespace: [TwinCAT.Router \[► 3255\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public RouterPortInfo(
    int port,
    string name
)
```

Parameters

port	Type: System.Int32 The port.
name	Type: System.String The name.

Reference

[RouterPortInfo Class \[► 3264\]](#)

[RouterPortInfo Overload \[► 3266\]](#)

[TwinCAT.Router Namespace \[► 3255\]](#)

6.15.2.2 RouterPortInfo Properties

The [RouterPortInfo \[► 3264\]](#) type exposes the following members.

Properties

	Name	Description
	Name [► 3267]	Gets the Name of the Port
	PortID [► 3267]	Gets the PortNumber

Reference

[RouterPortInfo Class \[► 3264\]](#)

[TwinCAT.Router Namespace \[► 3255\]](#)

6.15.2.2.1 RouterPortInfo.Name Property

Gets the Name of the Port

Namespace: [TwinCAT.Router \[► 3255\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public string Name { get; }
```

Property Value

Type: String

Reference

[RouterPortInfo Class \[► 3264\]](#)

[TwinCAT.Router Namespace \[► 3255\]](#)

6.15.2.2.2 RouterPortInfo.PortID Property

Gets the PortNumber

Namespace: [TwinCAT.Router \[► 3255\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public int PortID { get; }
```

Property Value

Type: Int32

Reference[RouterPortInfo Class \[▶ 3264\]](#)[TwinCAT.Router Namespace \[▶ 3255\]](#)**6.15.2.3 RouterPortInfo Methods**The [RouterPortInfo \[▶ 3264\]](#) type exposes the following members.**Methods**

	Name	Description
	Equals [▶ 3268]	Equals (Overrides Object.Equals(Object).)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetHashCode [▶ 3269]	Gets the GetHashCode of the Address (Overrides Object.GetHashCode..)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString [▶ 3269]	Returns a String that represents this instance. (Overrides Object.ToString..)
	TryParse [▶ 3269]	Tries to parse the RouterPortInfo [▶ 3264] object.

Reference[RouterPortInfo Class \[▶ 3264\]](#)[TwinCAT.Router Namespace \[▶ 3255\]](#)**6.15.2.3.1 RouterPortInfo.Equals Method**

Equals

Namespace: [TwinCAT.Router \[▶ 3255\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public override bool Equals(
    Object? obj
)
```

Parameters

obj	Type: System.Object
-----	---------------------

Return Value

Type: Boolean

Reference

[RouterPortInfo Class \[► 3264\]](#)

[TwinCAT.Router Namespace \[► 3255\]](#)

6.15.2.3.2 RouterPortInfo.GetHashCode Method

Gets the HashCode of the Address

Namespace: [TwinCAT.Router \[► 3255\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override int GetHashCode()
```

Return Value

Type: Int32

Reference

[RouterPortInfo Class \[► 3264\]](#)

[TwinCAT.Router Namespace \[► 3255\]](#)

6.15.2.3.3 RouterPortInfo.ToString Method

Returns a String that represents this instance.

Namespace: [TwinCAT.Router \[► 3255\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public override string ToString()
```

Return Value

Type: String

A String that represents this instance.

Reference

[RouterPortInfo Class \[► 3264\]](#)

[TwinCAT.Router Namespace \[► 3255\]](#)

6.15.2.3.4 RouterPortInfo.TryParse Method

Tries to parse the [RouterPortInfo \[► 3264\]](#) object.

Namespace: [TwinCAT.Router \[► 3255\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool TryParse(
    string value,
    out RouterPortInfo?? info
)
```

Parameters

value	Type: System.String The value.
info	Type: TwinCAT.Router.RouterPortInfo [▸ 3264] . The information.

Return Value

Type: Boolean
true if parsed, false otherwise.

Reference

[RouterPortInfo Class \[▸ 3264\]](#)

[TwinCAT.Router Namespace \[▸ 3255\]](#)

6.15.2.4 RouterPortInfo Operators

The [RouterPortInfo \[▸ 3264\]](#) type exposes the following members.

Operators

	Name	Description
	Equality [▸ 3270]	Operator==
	Inequality [▸ 3271]	Operator!=

Reference

[RouterPortInfo Class \[▸ 3264\]](#)

[TwinCAT.Router Namespace \[▸ 3255\]](#)

6.15.2.4.1 RouterPortInfo.Equality Operator

Operator==

Namespace: [TwinCAT.Router \[▸ 3255\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool operator ==(
    RouterPortInfo? o1,
    RouterPortInfo? o2
)
```

Parameters

o1	Type: TwinCAT.Router.RouterPortInfo [▶ 3264]
o2	Type: TwinCAT.Router.RouterPortInfo [▶ 3264]

Return Value

Type: Boolean

Reference

[RouterPortInfo Class](#) [[▶ 3264](#)]

[TwinCAT.Router Namespace](#) [[▶ 3255](#)]

6.15.2.4.2 RouterPortInfo.Inequality Operator

Operator=

Namespace: [TwinCAT.Router](#) [[▶ 3255](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static bool operator !=(
    RouterPortInfo? o1,
    RouterPortInfo? o2
)
```

Parameters

o1	Type: TwinCAT.Router.RouterPortInfo [▶ 3264]
o2	Type: TwinCAT.Router.RouterPortInfo [▶ 3264]

Return Value

Type: Boolean

Reference

[RouterPortInfo Class](#) [[▶ 3264](#)]

[TwinCAT.Router Namespace](#) [[▶ 3255](#)]

6.15.2.5 RouterPortInfo Fields

The [RouterPortInfo](#) [[▶ 3264](#)] type exposes the following members.

Fields

	Name	Description
	port [▶ 3272]	Port Number

	Name	Description
	portName [▶ 3272]	Port name

Reference

[RouterPortInfo Class](#) [[▶ 3264](#)]

[TwinCAT.Router Namespace](#) [[▶ 3255](#)]

6.15.2.5.1 RouterPortInfo.port Field

Port Number

Namespace: [TwinCAT.Router](#) [[▶ 3255](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected int port
```

Field Value

Type: Int32

Reference

[RouterPortInfo Class](#) [[▶ 3264](#)]

[TwinCAT.Router Namespace](#) [[▶ 3255](#)]

6.15.2.5.2 RouterPortInfo.portName Field

Port name

Namespace: [TwinCAT.Router](#) [[▶ 3255](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
protected string portName
```

Field Value

Type: String

Reference

[RouterPortInfo Class](#) [[▶ 3264](#)]

[TwinCAT.Router Namespace](#) [[▶ 3255](#)]

6.16 TwinCAT.SystemService Namespace

Classes

	Class	Description
	SystemServiceExtension [▶ 3281]	Class SystemServiceExtensions.

Structures

	Structure	Description
	AdsSysServState [▶ 3275]	Class AdsSysServState.

Also see about this

- [SystemServiceIndexGroup Enumeration](#) [[▶ 3273](#)]
- [TargetInfoOffset Enumeration](#) [[▶ 3275](#)]

6.16.1 SystemServiceIndexGroup Enumeration

System Service Index Groups.

Namespace: [TwinCAT.SystemService](#) [[▶ 3272](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public enum SystemServiceIndexGroup
```

Members

	Member name	Value	Description
	OPENCREATE	100	Open Create
	OPENREAD	101	Open read
	OPENWRITE	102	Open write
	CREATEFILE	110	Create file
	CLOSEHANDLE	111	Close handle
	DEVICENAMESBYCLASS	113	Device names by class
	FOPEN	120	File open
	FCLOSE	121	File closeReader
	FREAD	122	File read
	FWRITE	123	File write
	FSEEK	124	File seek
	FTELL	125	File tell
	FGETS	126	File gets
	FPUTS	127	File puts
	FSCANF	128	File scanf
	FPRINTF	129	File printf
	FEOF	130	End of file
	FDELETE	131	File delete
	FRENAME	132	File rename
	FFILEFIND	133	File find

	Member name	Value	Description
	FGETSTATUS	139	SYSTEMSERVICE_FGETSTATUS (134)
	FGETSIZE_BY_HANDLE	135	SYSTEMSERVICE_FGETSIZE_BY_HANDLE (135)
	MKDIR	138	SYSTEMSERVICE_MKDIR (138)
	RMDIR	139	SYSTEMSERVICE_RMDIR (139),
	UdpSocket	140	UDP Socket (SYSTEMSERVICE_UDP_SOCKET)
	UdpDiscovery	141	UDP Discovery (SYSTEMSERVICE_UDP_DISCOVERY, 141)
	CpDiscovery	142	CP Discovery
	XML_CREATEWRITER	150	
	XML_STARTELEMENT	151	
	XML_ENDELEMENT	152	
	XML_ADDATTRIBUTE	153	
	XML_FILLCHARELEMENT	154	
	XML_MULTIWRITE	155	
	XML_CLOSEWRITER	156	
	PRODUCT_VERSION	160	
	ADD_INITCMD	170	
	DEL_INITCMD	171	
	DEL_ALLADDEDINITCMDS	172	
	REG_HKEYLOCALMACHINE	200	Registry Key LOCAL MACHINE
	REG_HKEYCURRENTUSER	201	Registry Key CURRENT USER
	REG_HKEYCLASSESROOT	202	Registry Key CLASSES ROOT
	REG_FLUSHLOCALMACHINE	203	Flush LOCAL MACHINE
	REG_FLUSHCURRENTUSER	204	Flush current user
	REG_FLUSHCLASSESROOT	205	Flush classes root
	SYS_CONTROL	220	Sys Control
	SENDEMAIL	300	Send E-Mail
	TIMESERVICES	400	Timer services
	STARTPROCESS	500	Start process
	CHANGENETID	600	Change net ID
	QUERYNETID	601	Query net ID
	TARGETINFO	700	TargetInfo
	IPHELPERAPI	701	IPHELper API
	IPHOSTNAME	702	IP Hostname
	IPADDRINFO	703	IP AddressStr Info
	ADDREMOTE	801	Add remote Route (801)
	DELREMOTE	802	Delete remote Route
	ENUMREMOTE	803	Enumerate remote
	TEST_CONNECTION	804	Test connection
	SCREENRES	900	Screen resolution

Reference

[TwinCAT.SystemService Namespace \[► 3272\]](#)

6.16.2 TargetInfoOffset Enumeration

Enum TargetInfoOffset

Namespace: [TwinCAT.SystemService \[▶ 3272\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.116+a71ced3

Syntax

C#

```
public enum TargetInfoOffset
```

Members

	Member name	Value	Description
	CompatGlitchLoadTPSState	0	COMPATGLITCH_LOADTPSSTATE = 0, // used for backward compatibility to TC 2.8
	QueryXmlInfo	1	Queries for XML Information (TARGETINFO_QUERYXMLINFO)
	TargetType	2	The target type (TARGETINFO_TARGETTYPE)
	FileVersion	3	The file version (TARGETINFO_FILEVERSION)
	Platform	4	The platform (TARGETINFO_PLATFORM, string like "TwinCAT RT (x86)")
	ProjectGuid	5	The project Guid (TARGETINFO_PROJECTGUID, cbRead >= 32 returns projectGuid + projectVersionGuid)
	ProjectVersionGuid	6	The Project Version Guid (TARGETINFO_PROJECTVERSIONGUID)
	ProjectName	7	The project name (TARGETINFO_PROJECTNAME)
	QueryXmlInfoVerb	8	Queries for the XmlInfoVerb (TARGETINFO_QUERYXMLINFOVERB)
	SelfSignedCertFP	9	The self-signed certification Fingerprint (TARGETINFO_SELFIGNEDCERTFP)

Remarks

IndexOffsets for [TARGETINFO \[▶ 3273\]](#)

Reference

[TwinCAT.SystemService Namespace \[▶ 3272\]](#)

6.16.3 AdsSysServState Structure

Class AdsSysServState.

Namespace: [TwinCAT.SystemService \[▶ 3272\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public struct AdsSysServState
```

The AdsSysServState type exposes the following members.

Methods

	Name	Description
	Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType.)
	GetHashCode	Returns the hash code for this instance. (Inherited from ValueType.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType.)

Fields

	Name	Description
	AdsState [▶ 3277]	The AdsState
	Build [▶ 3277]	The build number of the target system.
	DeviceState [▶ 3278]	The device state
	Flags [▶ 3278]	The System Service flags.
	OsType [▶ 3279]	The Operation System ID of the target system.
	Platform [▶ 3279]	The platform ID of the target system.
	Reserved [▶ 3279]	Reserved data
	RestartIndex [▶ 3280]	The restart index
	Revision [▶ 3280]	The revision of the target system.
	Version [▶ 3280]	The major version of the target system.

Remarks

see TwinCAT.SystemService.SystemServiceIndexGroup.SYSTEMSERVICE_SYSSERV_STATE

Reference

[TwinCAT.SystemService Namespace \[▶ 3272\]](#)

6.16.3.1 AdsSysServState Methods

The [AdsSysServState \[▶ 3275\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType.)
	GetHashCode	Returns the hash code for this instance. (Inherited from ValueType.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType.)

Reference

[AdsSysServState Structure \[► 3275\]](#)

[TwinCAT.SystemService Namespace \[► 3272\]](#)

6.16.3.2 AdsSysServState Fields

The [AdsSysServState \[► 3275\]](#) type exposes the following members.

Fields

	Name	Description
	AdsState [► 3277]	The AdsState
	Build [► 3277]	The build number of the target system.
	DeviceState [► 3278]	The device state
	Flags [► 3278]	The System Service flags.
	OsType [► 3279]	The Operation System ID of the target system.
	Platform [► 3279]	The platform ID of the target system.
	Reserved [► 3279]	Reserved data
	RestartIndex [► 3280]	The restart index
	Revision [► 3280]	The revision of the target system.
	Version [► 3280]	The major version of the target system.

Reference

[AdsSysServState Structure \[► 3275\]](#)

[TwinCAT.SystemService Namespace \[► 3272\]](#)

6.16.3.2.1 AdsSysServState.AdsState Field

The AdsState

Namespace: [TwinCAT.SystemService \[► 3272\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsState AdsState
```

Field Value

Type: [AdsState \[► 729\]](#)

Reference

[AdsSysServState Structure \[► 3275\]](#)

[TwinCAT.SystemService Namespace \[► 3272\]](#)

6.16.3.2.2 AdsSysServState.Build Field

The build number of the target system.

Namespace: [TwinCAT.SystemService](#) [► 3272]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ushort Build
```

Field Value

Type: UInt16

Reference

[AdsSysServState Structure](#) [► 3275]

[TwinCAT.SystemService Namespace](#) [► 3272]

6.16.3.2.3 AdsSysServState.DeviceState Field

The device state

Namespace: [TwinCAT.SystemService](#) [► 3272]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ushort DeviceState
```

Field Value

Type: UInt16

Reference

[AdsSysServState Structure](#) [► 3275]

[TwinCAT.SystemService Namespace](#) [► 3272]

6.16.3.2.4 AdsSysServState.Flags Field

The System Service flags.

Namespace: [TwinCAT.SystemService](#) [► 3272]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public AdsSysServStateFlags Flags
```

Field Value

Type: AdsSysServStateFlags

Reference

[AdsSysServState Structure](#) [► 3275]

[TwinCAT.SystemService Namespace \[► 3272\]](#)

6.16.3.2.5 AdsSysServState.OsType Field

The Operation System ID of the target system.

Namespace: [TwinCAT.SystemService \[► 3272\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public byte OsType
```

Field Value

Type: Byte

Reference

[AdsSysServState Structure \[► 3275\]](#)

[TwinCAT.SystemService Namespace \[► 3272\]](#)

6.16.3.2.6 AdsSysServState.Platform Field

The platform ID of the target system.

Namespace: [TwinCAT.SystemService \[► 3272\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public byte Platform
```

Field Value

Type: Byte

Reference

[AdsSysServState Structure \[► 3275\]](#)

[TwinCAT.SystemService Namespace \[► 3272\]](#)

6.16.3.2.7 AdsSysServState.Reserved Field

Reserved data

Namespace: [TwinCAT.SystemService \[► 3272\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public ushort Reserved
```

Field Value

Type: UInt16

Reference[AdsSysServState Structure \[► 3275\]](#)[TwinCAT.SystemService Namespace \[► 3272\]](#)**6.16.3.2.8 AdsSysServState.RestartIndex Field**

The restart index

Namespace: [TwinCAT.SystemService \[► 3272\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public ushort RestartIndex
```

Field Value

Type: UInt16

Reference[AdsSysServState Structure \[► 3275\]](#)[TwinCAT.SystemService Namespace \[► 3272\]](#)**6.16.3.2.9 AdsSysServState.Revision Field**

The revision of the target system.

Namespace: [TwinCAT.SystemService \[► 3272\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public byte Revision
```

Field Value

Type: Byte

Reference[AdsSysServState Structure \[► 3275\]](#)[TwinCAT.SystemService Namespace \[► 3272\]](#)**6.16.3.2.10 AdsSysServState.Version Field**

The major version of the target system.

Namespace: [TwinCAT.SystemService \[► 3272\]](#)**Assembly:** TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public byte Version
```

Field Value

Type: Byte

Reference

[AdsSysServState Structure \[▸ 3275\]](#)

[TwinCAT.SystemService Namespace \[▸ 3272\]](#)

6.16.4 SystemServiceExtension Class

Class SystemServiceExtensions.

Inheritance Hierarchy

System.Object
 TwinCAT.SystemService.SystemServiceExtension

Namespace: [TwinCAT.SystemService \[▸ 3272\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static class SystemServiceExtension
```

The SystemServiceExtension type exposes the following members.

Methods

	Name	Description
	PollSystemServiceState [▸ 3282]	Polls the state of the system service.
	PollSystemServiceStateAsync [▸ 3283]	Polls the system service state asynchronously
	ReadSysServState [▸ 3284]	Reads the System Service state (AdsSysServState [▸ 3275])
	ReadSysServStateAsync [▸ 3284]	Reads the System Service state (AdsSysServState [▸ 3275]) (asynchronous)
	RestartTwinCATAsync [▸ 3285]	Sends a SystemService Restart to the target system and waits until the system has finished the state changes (SystemService, Port 10000) (asynchronously)
	WaitUntilRestarted [▸ 3286]	Waits until the Restart is detected on the client (SystemService, Port 10000)
	WaitUntilRestartedAsync [▸ 3287]	Waits until the SystemService Restart is detected on the client (SystemService, Port 10000) (asynchronously)

Reference

[TwinCAT.SystemService Namespace \[► 3272\]](#)

6.16.4.1 SystemServiceExtension Methods

The [SystemServiceExtension \[► 3281\]](#) type exposes the following members.

Methods

	Name	Description
	PollSystemServiceState [► 3282]	Polls the state of the system service.
	PollSystemServiceStateAsync [► 3283]	Polls the system service state asynchronously
	ReadSysServState [► 3284]	Reads the System Service state (AdsSysServState [► 3275])
	ReadSysServStateAsync [► 3284]	Reads the System Service state (AdsSysServState [► 3275]) (asynchronous)
	RestartTwinCATAsync [► 3285]	Sends a SystemService Restart to the target system and waits until the system has finished the state changes (SystemService, Port 10000) (asynchronously)
	WaitUntilRestarted [► 3286]	Waits until the Restart is detected on the client (SystemService, Port 10000)
	WaitUntilRestartedAsync [► 3287]	Waits until the SystemService Restart is detected on the client (SystemService, Port 10000) (asynchronously)

Reference

[SystemServiceExtension Class \[► 3281\]](#)

[TwinCAT.SystemService Namespace \[► 3272\]](#)

6.16.4.1.1 SystemServiceExtension.PollSystemServiceState Method

Polls the state of the system service.

Namespace: [TwinCAT.SystemService \[► 3272\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static IObservable<ResultValue<AdsSysServState>> PollSystemServiceState (
    this IAdsConnection connection,
    IObservable<Unit> trigger
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The client (should be an IAdsConnection on port 10000 with FastWriteThrough)
------------	--

trigger	Type: System.IObservable.Unit. The trigger.
---------	--

Return Value

Type: IObservable.ResultValue [▶ 1181].AdsSysServState [▶ 3275]..
IObservable<ResultValue<AdsSysServState>>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [▶ 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[SystemServiceExtension Class](#) [▶ 3281]

[TwinCAT.SystemService Namespace](#) [▶ 3272]

6.16.4.1.2 SystemServiceExtension.PollSystemServiceStateAsync Method

Polls the system service state asynchronously

Namespace: [TwinCAT.SystemService](#) [▶ 3272]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultValue<AdsSysServState>> PollSystemServiceStateAsync (
    this IAdsConnection connection,
    IObservable<Unit> trigger,
    CancellationToken cancel
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The client (should be an IAdsConnection on port 10000 with FastWriteThrough)
trigger	Type: System.IObservable.Unit. The trigger.
cancel	Type: System.Threading.CancellationToken The cancellation token that can be used by other objects or threads to receive notice of cancellation.

Return Value

Type: IObservable.ResultValue [▶ 1181].AdsSysServState [▶ 3275]..
IObservable<ResultValue<AdsSysServState>>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [▶ 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Exceptions

Exception	Condition
ArgumentOutOfRangeException	Client doesn't target port 10000! - client

Reference

[SystemServiceExtension Class \[▶ 3281\]](#)

[TwinCAT.SystemService Namespace \[▶ 3272\]](#)

6.16.4.1.3 SystemServiceExtension.ReadSysServState Method

Reads the System Service state ([AdsSysServState \[▶ 3275\]](#))

Namespace: [TwinCAT.SystemService \[▶ 3272\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static ResultValue<AdsSysServState> ReadSysServState(
    this IAdsConnection connection
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The client (should be an IAdsConnection on port 10000)
------------	--

Return Value

Type: [ResultValue \[▶ 1181\].AdsSysServState \[▶ 3275\]](#).
ResultValue<AdsSysServState>.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection \[▶ 876\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Exceptions

Exception	Condition
ArgumentOutOfRangeException	Service only available on AmsPort 10000 (SystemService). Cannot use this function on the specified client! - client

Reference

[SystemServiceExtension Class \[▶ 3281\]](#)

[TwinCAT.SystemService Namespace \[▶ 3272\]](#)

6.16.4.1.4 SystemServiceExtension.ReadSysServStateAsync Method

Reads the System Service state ([AdsSysServState \[▶ 3275\]](#)) (asynchronous)

Namespace: [TwinCAT.SystemService \[▶ 3272\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static Task<ResultValue<AdsSysServState>> ReadSysServStateAsync(
    this IAdsConnection connection,
    CancellationToken cancel
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▸ 876] The client.
cancel	Type: System.Threading.CancellationToken The cancellation token that can be used by other objects or threads to receive notice of cancellation.

Return Value

Type: [Task.ResultValue \[▸ 1181\].AdsSysServState \[▸ 3275\]](#)..
A Task<ResultValue`1> representing the asynchronous operation.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection \[▸ 876\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Exceptions

Exception	Condition
ArgumentOutOfRangeException	Service only available on AmsPort 10000 (SystemService). Cannot use this function on the specified client! - client

Reference

[SystemServiceExtension Class \[▸ 3281\]](#)

[TwinCAT.SystemService Namespace \[▸ 3272\]](#)

6.16.4.1.5 SystemServiceExtension.RestartTwinCATAsync Method

Sends a SystemService Restart to the target system and waits until the system has finished the state changes (SystemService, Port 10000) (asynchronously)

Namespace: [TwinCAT.SystemService \[▸ 3272\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static Task<ResultWriteControl> RestartTwinCATAsync(
    this IAdsConnection connection,
    AdsStateCommand request,
    TimeSpan pollingRate,
    TimeSpan waitTimeout,
    bool noReinit,
    CancellationToken cancel
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection (should be an IAdsConnection on port 10000 in FastWriteThrough mode)
request	Type: TwinCAT.Ads.AdsStateCommand [▶ 1233] The request/command for (Reconfig/Reset)
pollingRate	Type: System.TimeSpan The polling rate (should be lower than the connection timeout).
waitTimeout	Type: System.TimeSpan The wait timeout.
noReinit	Type: System.Boolean if set to true then this method first checks if the system is already in the requested target state. In that case the method can return immediatly.
cancel	Type: System.Threading.CancellationToken The cancellation token that can be used by other objects or threads to receive notice of cancellation.

Return Value

Type: [Task.ResultWriteControl](#) [[▶ 1236](#)].

A [Task<System.Boolean>](#) representing whether the restart is detected (true), or the timeout occurred (false).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Exceptions

Exception	Condition
ArgumentException	The connection is not conncted to '{AmsPort.SystemService}' - client
ArgumentOutOfRangeException	request

Remarks

Be sure to have the client in FastWriteThrough settings, that the

Reference

[SystemServiceExtension Class](#) [[▶ 3281](#)]

[TwinCAT.SystemService Namespace](#) [[▶ 3272](#)]

6.16.4.1.6 SystemServiceExtension.WaitUntilRestarted Method

Waits until the Restart is detected on the client (SystemService, Port 10000)

Namespace: [TwinCAT.SystemService](#) [[▶ 3272](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax**C#**

```
public static bool WaitUntilRestarted(
    this IAdsConnection connection,
    TimeSpan pollingRate,
```

```

    TimeSpan waitTimeout,
    CancellationToken cancel
)

```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection (should be an IAdsConnection on port 10000)
pollingRate	Type: System.TimeSpan The polling rate (should be lower than the connection timeout)
waitTimeout	Type: System.TimeSpan The wait timeout.
cancel	Type: System.Threading.CancellationToken The cancellation token that can be used by other objects or threads to receive notice of cancellation.

Return Value

Type: Boolean
true if the restart was detected, false if WaitUntilRestarted ran into the wait timeout.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection \[▶ 876\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

- [SystemServiceExtension Class \[▶ 3281\]](#)
- [TwinCAT.SystemService Namespace \[▶ 3272\]](#)

6.16.4.1.7 SystemServiceExtension.WaitUntilRestartedAsync Method

Waits until the SystemService Restart is detected on the client (SystemService, Port 10000) (asynchronously)

Namespace: [TwinCAT.SystemService \[▶ 3272\]](#)
Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public static Task<bool> WaitUntilRestartedAsync (
    this IAdsConnection connection,
    TimeSpan pollingRate,
    TimeSpan waitTimeout,
    CancellationToken cancel
)

```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The client (should be an IAdsConnection on port 10000)
pollingRate	Type: System.TimeSpan The polling rate (should be lower than the connection timeout).
waitTimeout	Type: System.TimeSpan The wait timeout.
cancel	Type: System.Threading.CancellationToken The cancellation token that can be used by other objects or threads to receive notice of cancellation.

Return Value

Type: Task.Boolean.

A Task<System.Boolean> representing whether the restart is detected (true), or the timeout occurred (false).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [AdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[SystemServiceExtension Class](#) [► 3281]

[TwinCAT.SystemService Namespace](#) [► 3272]

6.17 TwinCAT.Tls Namespace

Classes

	Class	Description
	TlsException [► 3288]	Base class for all exceptions thrown by the TwinCAT.Ads component

6.17.1 TlsException Class

Base class for all exceptions thrown by the TwinCAT.Ads component

Inheritance Hierarchy

```

System.Object
  System.Exception
    TwinCAT.Tls.TlsException

```

Namespace: [TwinCAT.Tls](#) [► 3288]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```

[SerializableAttribute]
public class TlsException : Exception

```

The TlsException type exposes the following members.

Constructors

	Name	Description
	TlsException [► 3289]	Initializes a new Instance of the AdsException [► 61] class.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	Error [► 3290]	Gets the TlsError value.

	Name	Description
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)

Reference

[TwinCAT.Tls Namespace \[► 3288\]](#)

6.17.1.1 TlsException Constructor

Initializes a new Instance of the [AdsException \[► 61\]](#) class.

Namespace: [TwinCAT.Tls \[► 3288\]](#)

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public TlsException(
    string message,
    TlsError error
)
```

Parameters

message	Type: System.String A message that describes the error.
error	Type: TlsError The error.

Reference

[TlsException Class](#) [[▶ 3288](#)]

[TwinCAT.Tls Namespace](#) [[▶ 3288](#)]

6.17.1.2 TlsException Properties

The [TlsException](#) [[▶ 3288](#)] type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
	Error [▶ 3290]	Gets the TlsError value.
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

Reference

[TlsException Class](#) [[▶ 3288](#)]

[TwinCAT.Tls Namespace](#) [[▶ 3288](#)]

6.17.1.2.1 TlsException.Error Property

Gets the TlsError value.

Namespace: [TwinCAT.Tls](#) [[▶ 3288](#)]

Assembly: TwinCAT.Ads.Abstractions (in TwinCAT.Ads.Abstractions.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public TlsError Error { get; }
```

Property Value

Type: TlsError
The error.

Reference

[TlsException Class \[► 3288\]](#)

[TwinCAT.Tls Namespace \[► 3288\]](#)

6.17.1.3 TlsException Methods

The [TlsException \[► 3288\]](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
	MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

Reference

[TlsException Class \[► 3288\]](#)

[TwinCAT.Tls Namespace \[► 3288\]](#)

6.17.1.4 TlsException Events

The [TlsException \[► 3288\]](#) type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Obsolete. Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception.)

Reference[TlsException Class \[► 3288\]](#)[TwinCAT.Tls Namespace \[► 3288\]](#)

6.18 TwinCAT.Ads.Extensions Namespace

Classes

	Class	Description
	ConnectionStateExtension [► 3292]	Class ConnectionStateExtension.

6.18.1 ConnectionStateExtension Class

Class ConnectionStateExtension.

Inheritance Hierarchy[System.Object](#)

TwinCAT.Ads.Extensions.ConnectionStateExtension

Namespace: [TwinCAT.Ads.Extensions \[► 3292\]](#)**Assembly:** TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229**Syntax****C#**

```
public static class ConnectionStateExtension
```

The ConnectionStateExtension type exposes the following members.

Methods

	Name	Description
	PollAdsState(IAdsConnection, IObservable.Unit.) [► 3294]	Gets an observable sequence of ResultReadAdsState [► 1146] s via Polling.
	PollAdsState(IAdsConnection, TimeSpan) [► 3295]	Gets an observable sequence of ResultReadAdsState [► 1146] s via Polling.
	PollAdsStateAsync(IAdsConnection, IObservable.Unit., CancellationToken) [► 3296]	Gets an observable sequence of AdsState [► 729] s via Polling.
	PollAdsStateAsync(IAdsConnection,	Gets an observable sequence of AdsState [► 729] s via Polling.

	Name	Description
	TimeSpan, CancellationToken [▶ 3298]	
	SetAdsState [▶ 3299]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes.
	SetAdsStateAsync [▶ 3300]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes.

Reference

[TwinCAT.Ads.Extensions Namespace](#) [▶ 3292]

6.18.1.1 ConnectionStateExtension Methods

The [ConnectionStateExtension](#) [▶ 3292] type exposes the following members.

Methods

	Name	Description
	PollAdsState(IAdsConnection, IObservable.Unit) [▶ 3294]	Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling.
	PollAdsState(IAdsConnection, TimeSpan) [▶ 3295]	Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling.
	PollAdsStateAsync(IAdsConnection, IObservable.Unit, CancellationToken) [▶ 3296]	Gets an observable sequence of AdsState [▶ 729]s via Polling.
	PollAdsStateAsync(IAdsConnection, TimeSpan, CancellationToken) [▶ 3298]	Gets an observable sequence of AdsState [▶ 729]s via Polling.
	SetAdsState [▶ 3299]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes.
	SetAdsStateAsync [▶ 3300]	Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes.

Reference

[ConnectionStateExtension Class](#) [▶ 3292]

[TwinCAT.Ads.Extensions Namespace](#) [▶ 3292]

6.18.1.1.1 ConnectionStateExtension.PollAdsState Method

Overload List

	Name	Description
	PollAdsState(IAdsConnection, IObservable.Unit.) [▶ 3294]	Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling.
	PollAdsState(IAdsConnection, TimeSpan) [▶ 3295]	Gets an observable sequence of ResultReadAdsState [▶ 1146]s via Polling.

Reference

[ConnectionStateExtension Class](#) [▶ 3292]

[TwinCAT.Ads.Extensions Namespace](#) [▶ 3292]

6.18.1.1.1.1 ConnectionStateExtension.PollAdsState Method (IAdsConnection, IObservable.Unit.)

Gets an observable sequence of [ResultReadAdsState](#) [▶ 1146]s via Polling.

Namespace: [TwinCAT.Ads.Extensions](#) [▶ 3292]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultReadAdsState> PollAdsState(
    this IAdsConnection connection,
    IObservable<Unit> trigger
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
trigger	Type: System.IObservable.Unit. The polling trigger

Return Value

Type: [IObservable.ResultReadAdsState](#) [▶ 1146].
[IObservable<AdsState>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [▶ 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Reference

[ConnectionStateExtension Class](#) [▶ 3292]

[PollAdsState Overload](#) [[▶ 3294](#)]

[TwinCAT.Ads.Extensions Namespace](#) [[▶ 3292](#)]

6.18.1.1.1.2 ConnectionStateExtension.PollAdsState Method (IAdsConnection, TimeSpan)

Gets an observable sequence of [ResultReadAdsState](#) [[▶ 1146](#)]s via Polling.

Namespace: [TwinCAT.Ads.Extensions](#) [[▶ 3292](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultReadAdsState> PollAdsState(
    this IAdsConnection connection,
    TimeSpan pollingIntervall
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
pollingIntervall	Type: System.TimeSpan The polling interval.

Return Value

Type: [IObservable.ResultReadAdsState](#) [[▶ 1146](#)].
[IObservable<AdsState>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[▶ 876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

The following sample shows how observe [AdsState](#) [[▶ 729](#)] via polling with the reactive [AdsClientExtensions](#) [[▶ 1215](#)]

Observe changing ADS States with reactive extensions.

```
// To Test the observer, Start/Stop the local PLC
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.DefaultDynamic);

    // Reactive Change Handler
    var valueObserver = Observer.Create<IList<AdsState>>(not =>
    {
        AdsState oldValue = not[0];
        AdsState newValue = not[1];

        Console.WriteLine(string.Format("Changed ADSState from '{0}' --> '{1}!", oldValue, newValue));
    });
};
```

```
// Create a subscription for the AdsState change and buffering 2 Values (for oldValue --
> newValue output).
IDisposable subscription = client.PollAdsState(TimeSpan.FromMilliseconds(200)).Buffer(2, 1).Subscribe(valueObserver);

Console.ReadKey(); // Wait for Key press
subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[ConnectionStateExtension Class](#) [► 3292]

[PollAdsState Overload](#) [► 3294]

[TwinCAT.Ads.Extensions Namespace](#) [► 3292]

6.18.1.1.2 ConnectionStateExtension.PollAdsStateAsync Method

Overload List

	Name	Description
	PollAdsStateAsync(IAdsConnection, IObservable.Unit, Cancellation.Token) [► 3296]	Gets an observable sequence of AdsState [► 729]s via Polling.
	PollAdsStateAsync(IAdsConnection, TimeSpan, Cancellation.Token) [► 3298]	Gets an observable sequence of AdsState [► 729]s via Polling.

Reference

[ConnectionStateExtension Class](#) [► 3292]

[TwinCAT.Ads.Extensions Namespace](#) [► 3292]

6.18.1.1.2.1 ConnectionStateExtension.PollAdsStateAsync Method (IAdsConnection, IObservable.Unit., Cancellation.Token)

Gets an observable sequence of [AdsState](#) [► 729]s via Polling.

Namespace: [TwinCAT.Ads.Extensions](#) [► 3292]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultReadAdsState> PollAdsStateAsync(
    this IAdsConnection connection,
    IObservable<Unit> trigger,
    Cancellation.Token cancel
)
```


Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [► 876] The connection.
trigger	Type: System.IObservable.Unit . The polling trigger
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [IObservable.ResultReadAdsState](#) [► 1146].
[IObservable<AdsState>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [► 876]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

The following sample shows how observe [AdsState](#) [► 729] via polling with the reactive [AdsClientExtensions](#) [► 1215]

Observe changing ADS States with reactive extensions.

```
// To Test the observer, Start/Stop the local PLC
using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.DefaultDynamic);

    // Reactive Change Handler
    var valueObserver = Observer.Create<IList<AdsState>>(not =>
    {
        AdsState oldValue = not[0];
        AdsState newValue = not[1];

        Console.WriteLine(string.Format("Changed ADSState from '{0}' --> '{1}!", oldValue, newValue));
    });

    // Create a subscription for the AdsState change and buffering 2 Values (for oldValue --
    > newValue output).
    IDisposable subscription = client.PollAdsState(TimeSpan.FromMilliseconds(200)).Buffer(2, 1).Subscribe(valueObserver);

    Console.ReadKey(); // Wait for Key press
    subscription.Dispose(); // Dispose the Subscription
}
```

Reference

[ConnectionStateExtension Class](#) [► 3292]

[PollAdsStateAsync Overload](#) [► 3296]

[TwinCAT.Ads.Extensions Namespace](#) [► 3292]

6.18.1.1.2 ConnectionStateExtension.PollAdsStateAsync Method (IAdsConnection, TimeSpan, CancellationToken)

Gets an observable sequence of [AdsState](#) [[729](#)]s via Polling.

Namespace: [TwinCAT.Ads.Extensions](#) [[3292](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static IObservable<ResultReadAdsState> PollAdsStateAsync(
    this IAdsConnection connection,
    TimeSpan pollingInterval,
    CancellationToken cancel
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [876] The connection.
pollingInterval	Type: System.TimeSpan The interval.
cancel	Type: System.Threading.CancellationToken The cancellation token.

Return Value

Type: [IObservable.ResultReadAdsState](#) [[1146](#)].

[IObservable<AdsState>](#).

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection](#) [[876](#)]. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Examples

The following sample shows how observe [AdsState](#) [[729](#)] via polling with the reactive [AdsClientExtensions](#) [[1215](#)]

Observe changing ADS States with reactive extensions.

```
// To Test the observer, Start/Stop the local PLC

using (AdsClient client = new AdsClient())
{
    // Connect to target
    client.Connect(new AmsAddress(AmsNetId.Local, 851));

    // Create Symbol information
    var symbolLoader = SymbolLoaderFactory.Create(client, SymbolLoaderSettings.DefaultDynamic);

    // Reactive Change Handler
    var valueObserver = Observer.Create<IList<AdsState>>(not =>
    {
        AdsState oldValue = not[0];
        AdsState newValue = not[1];

        Console.WriteLine(string.Format("Changed ADSState from '{0}' --> '{1}!", oldValue, newValue));
    });

    // Create a subscription for the AdsState change and buffering 2 Values (for oldValue --
    > newValue output).
```

```

IDisposable subscription = client.PollAdsState(TimeSpan.FromMilliseconds(200)).Buffer(2, 1).Subscribe(valueObserver);

Console.ReadKey(); // Wait for Key press
subscription.Dispose(); // Dispose the Subscription
}

```

Reference

[ConnectionStateExtension Class \[▶ 3292\]](#)

[PollAdsStateAsync Overload \[▶ 3296\]](#)

[TwinCAT.Ads.Extensions Namespace \[▶ 3292\]](#)

6.18.1.1.3 ConnectionStateExtension.SetAdsState Method

Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes.

Namespace: [TwinCAT.Ads.Extensions \[▶ 3292\]](#)

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```

public static ResultWriteControl SetAdsState(
    this IAdsConnection connection,
    AdsStateCommand requested,
    TimeSpan pollingRate,
    TimeSpan waitTimeout,
    bool noReinit,
    bool noWait
)

```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
requested	Type: TwinCAT.Ads.AdsStateCommand [▶ 1233] The requested state / command.
pollingRate	Type: System.TimeSpan The polling rate.
waitTimeout	Type: System.TimeSpan The wait timeout.
noReinit	Type: System.Boolean Checks first for the actual state and decides to return immediatly if the device is already in the expected target state.
noWait	Type: System.Boolean if set to true [no wait].

Return Value

Type: [ResultWriteControl \[▶ 1236\]](#)
ResultWriteControl.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection \[▶ 876\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Exceptions

Exception	Condition
ClientNotConnectedException [▶ 67]	
AdsException [▶ 61]	Polling rate higher than connection timeout!
AdsException [▶ 61]	Cannot set '{address}' to '{requested}'. This would leave the target system unavailable for ADS!

Reference

[ConnectionStateExtension Class](#) [[▶ 3292](#)]

[TwinCAT.Ads.Extensions Namespace](#) [[▶ 3292](#)]

6.18.1.1.4 ConnectionStateExtension.SetAdsStateAsync Method

Sets an AdsState/AdsStateCommand to an AdsServer and waits until the StateChange finishes.

Namespace: [TwinCAT.Ads.Extensions](#) [[▶ 3292](#)]

Assembly: TwinCAT.Ads (in TwinCAT.Ads.dll) Version: 6.0.328+39e3229

Syntax

C#

```
public static Task<ResultWriteControl> SetAdsStateAsync(
    this IAdsConnection connection,
    AdsStateCommand requested,
    TimeSpan pollingRate,
    TimeSpan waitTimeout,
    bool noReinit,
    bool noWait,
    CancellationToken cancel
)
```

Parameters

connection	Type: TwinCAT.Ads.IAdsConnection [▶ 876] The connection.
requested	Type: TwinCAT.Ads.AdsStateCommand [▶ 1233] The requested state / command.
pollingRate	Type: System.TimeSpan The polling rate.
waitTimeout	Type: System.TimeSpan The wait timeout.
noReinit	Type: System.Boolean Checks first for the actual state and decides to return immediately if the device is already in the expected target state.
noWait	Type: System.Boolean if set to true [no wait].
cancel	Type: System.Threading.CancellationToken The cancellation token that can be used by other objects or threads to receive notice of cancellation.

Return Value

Type: [Task.ResultWriteControl](#) [[▶ 1236](#)].
ResultWriteControl.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type [IAdsConnection \[► 876\]](#). When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods \(Visual Basic\)](#) or [Extension Methods \(C# Programming Guide\)](#).

Exceptions

Exception	Condition
ClientNotConnectedException [► 67]	
AdsException [► 61]	Polling rate higher than connection timeout!
AdsException [► 61]	Cannot set '{address}' to '{requested}'. This would leave the target system unavailable for ADS!

Reference

[ConnectionStateExtension Class \[► 3292\]](#)

[TwinCAT.Ads.Extensions Namespace \[► 3292\]](#)

More Information:
www.beckhoff.com/tc1000

Beckhoff Automation GmbH & Co. KG
Hülshorstweg 20
33415 Verl
Germany
Phone: +49 5246 9630
info@beckhoff.com
www.beckhoff.com

