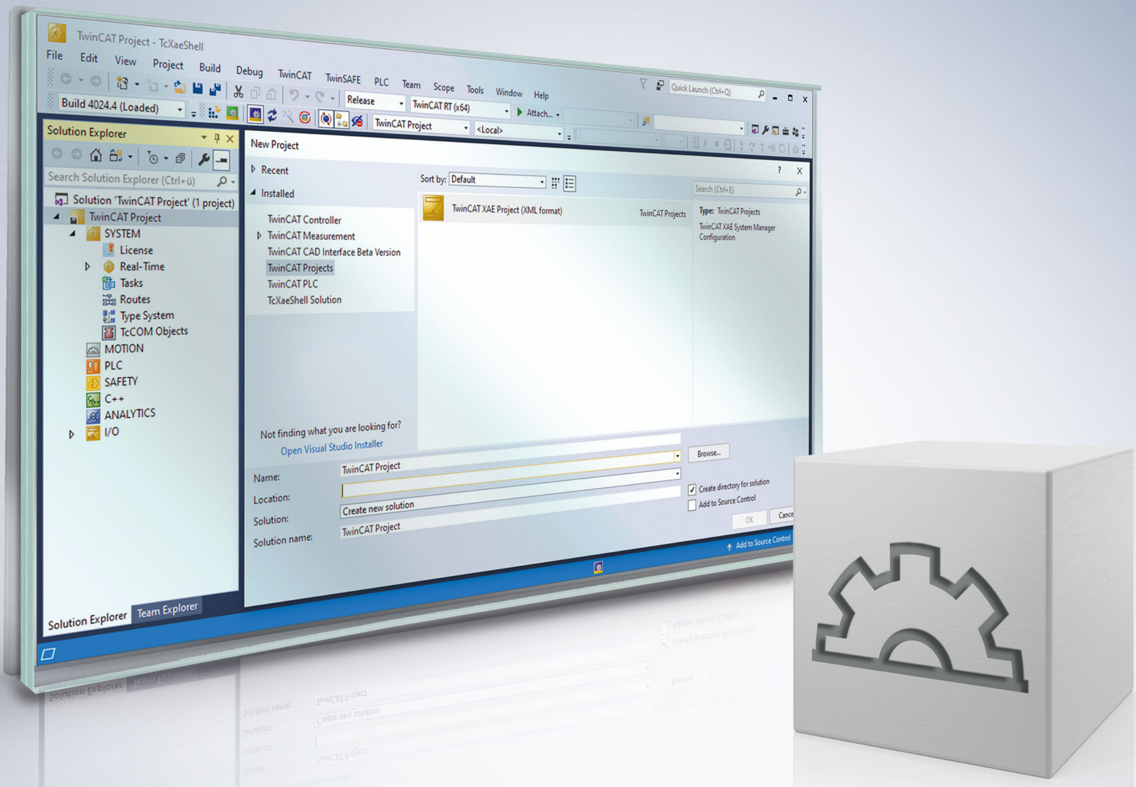
BECKHOFF New Automation Technology

Manual | EN

TE1000

TwinCAT 3 | PLC Library: Tc2_NcDrive



2024-01-04 | Version: 1.2.4

Table of contents

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

Copyright

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

ℹ 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 Overview

The Tc2_NcDrive library should no longer be used in newer projects. Please use the Tc2_MC2_Drive library instead (see documentation TwinCAT 3 PLC Lib Tc2_MC2_Drive).

The Tc2_NcDrive library includes functions and function blocks for SoE drives that access the drive by MC2 axis structure (AXIS_REF).

Drive libraries

The three drive libraries Tc2_Drive, Tc2_NcDrive and Tc2_MC2_Drive were developed for different functional purposes, but are almost identical in their functionality. The function blocks of the libraries Tc2_NcDrive and Tc_MC2_Drive form wrapper function blocks around the function blocks of the Tc2_Drive library.

Drive library	Use	Access to the drive	Remarks
Tc2_Drive See documentation TwinCAT 3 PLC Lib: Tc2_Drive	Use the Tc2_Drive library if you use the drive entirely from the PLC (i.e. without NC).	The drive is accessed via a drive reference. Within the library, the ST_DriveRef structure is used for this with the NetID as a string. For linking purposes, a structure called ST_PlcDriveRef is also provided with the NetID as a byte array. (See Drive reference ST_DriveRef)	If you want to access parameters in the drive for which no special function block has been implemented, use the function blocks FB_SoERead_ByDriveRef and FB_SoEWrite_ByDriveRef. These function blocks are implemented in the PLC Lib Tc2_EtherCAT in the SoE Interface folder.
Tc2_NcDrive See documentation TwinCAT 3 PLC Lib: Tc2_NcDrive	Use the Tc2_NcDrive library if you are using the drive via the NC with the Tc2_Nc libraries.	The drive is accessed via the NC axis structure (NC_TO_PLC). The function blocks of the Tc2_NcDrive library independently determine the access data to the drive (NetID, address and channel number) via the NC axis ID from the NC axis structure.	If you want to access parameters in the drive for which no special function block has been implemented, use the function blocks FB_SoERead and FB_SoEWrite.
Tc2_MC2_Drive See: TwinCAT 3 PLC Lib Tc2_MC2_Drive documentation	Use the Tc2_MC2_Drive library if you are using the drive via the NC with the Tc2_MC2 library.	The drive is accessed via the MC2 axis reference (AXIS_REF). The function blocks of the Tc2_MC2_Drive library independently determine the access data to the drive (NetID, address and channel number) via the NC axis ID from the MC2 axis reference.	If you want to access parameters in the drive for which no special function block has been implemented, use the function blocks FB_SoERead and FB_SoEWrite.

Note the differences when using the drive libraries with AX5000 and Bosch Rexroth IndraDrive CS (see Samples)

Functions

Name	Description
F_GetVersionTcNcDrive [▶ 41]	Reads version information from the PLC library. The function has been replaced by the global structure stLibVersion_Tc2_NcDrive.
F_ConvWordToSTAX5000C1D	Converts the C1D word (S-0-0011) of the AX5000 to an ST_AX5000_C1D structure.

Name	Description
	See: TwinCAT 3 PLC Lib documentation: Tc2_Drive.

Function blocks

Name	Description
FB_SoEReset [▶ 10]	Resets the drive (S-0-0099).
FB_SoEWritePassword [▶ 11]	Sets the drive password (S-0-0267).
FB_SoEReadDiagMessage [▶ 17]	Reads the diagnostic message (S-0-0095).
FB_SoEReadDiagNumber [▶ 18]	Reads the diagnostic number (S-0-0390).
FB_SoEReadDiagNumberList [▶ 20]	Reads the diagnostic number list (up to 30 entries) (S-0-0375)
FB_SoEExecuteCommand [▶ 13]	Executes a command.
FB_SoEWriteCommandControl [▶ 14]	Sets the Command Control.
FB_SoEReadCommandState [▶ 15]	Checks the command status.
FB_SoEReadClassXDiag [▶ 21]	Reads Class 1 diagnosis (S-0-0011) ... Class 3 diagnosis (S-0-0013).
FB_SoERead [▶ 23]	Reads a parameter.
FB_SoEWrite [▶ 24]	Writes a parameter.
FB_SoEReadAmplifierTemperature [▶ 26]	Reads the drive temperature (S-0-0384).
FB_SoEReadMotorTemperature [▶ 28]	Reads the motor temperature (S-0-0383).
FB_SoEReadDcBusCurrent [▶ 29]	Reads the DC bus current (S-0-0381).
FB_SoEReadDcBusVoltage [▶ 30]	Reads the DC bus voltage (S-0-0380).
FB_SoEAX5000ReadActMainVoltage [▶ 34]	Reads the mains voltage (P-0-0200).
FB_SoEAX5000SetMotorCtrlWord [▶ 36]	Sets the Motor Control Word (P-0-0096).
FB_SoEAX5000FirmwareUpdate [▶ 37]	Executes an automatic firmware update for the AX5000.
FB_CoERead [▶ 32]	Reads a parameter.
FB_CoEWrite [▶ 33]	Writes a parameter.

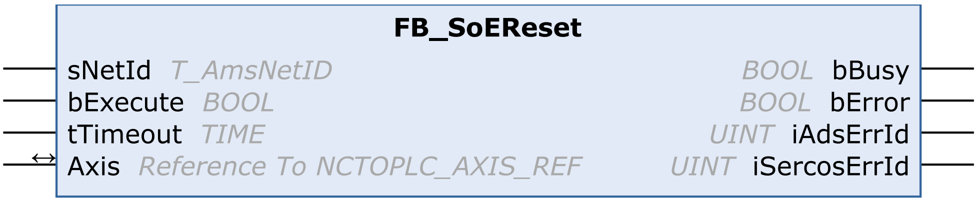
Requests

Component	Version
TwinCAT on the development computer	3.1 Build 4016 or higher
TwinCAT on the Windows CE-Image	3.1 Build 4016 or higher
TwinCAT on the Windows XP-Image	3.1 Build 4016 or higher

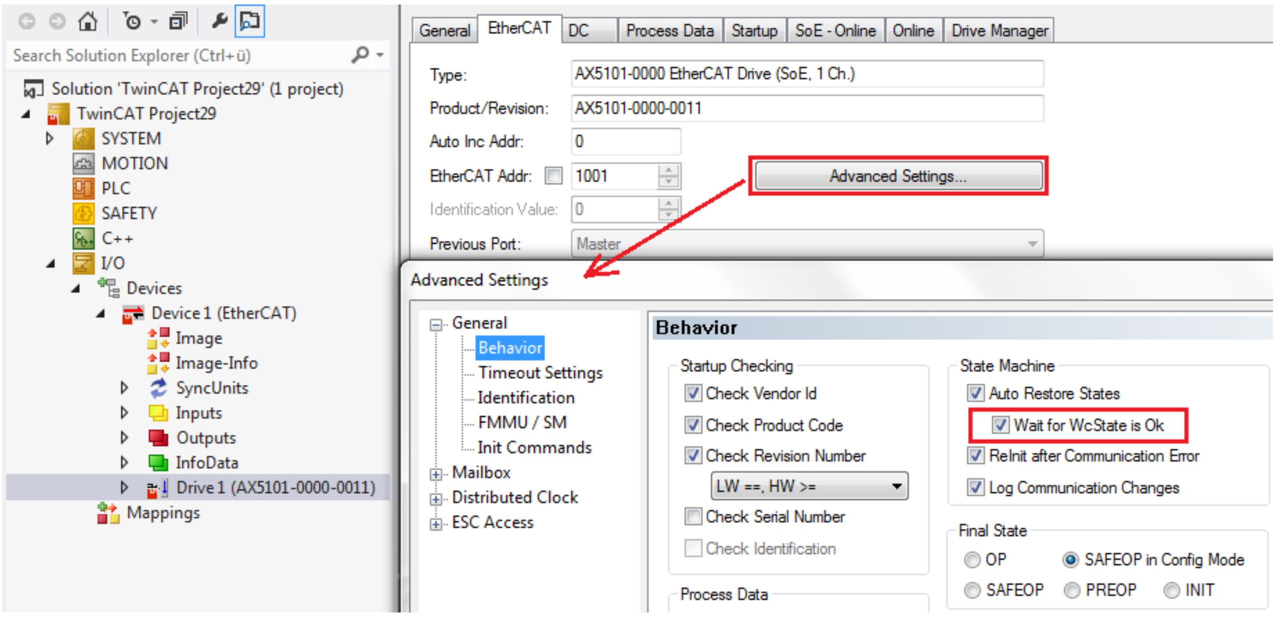
3 Function blocks

3.1 General SoE

3.1.1 FB_SoEReset



The drive (S-0-0099) can be reset with the function block FB_SoEReset. In the case of multiple-channel devices if necessary, both channels will have to perform a reset. The timeout time must be 10 s, as the reset can take up to 10 s depending on the error. The flag "Wait For WcState is OK" must be enabled in the advanced EtherCAT settings for the AX5000.



Type: AX5101-0000 EtherCAT Drive (SoE, 1 Ch.)
Product/Revision: AX5101-0000-0011
Auto Inc Addr: 0
EtherCAT Addr: □ 1001
Identification Value: 0
Previous Port: Master

Behavior

Startup Checking
☑ Check Vendor Id
☑ Check Product Code
☑ Check Revision Number
LW ==, HW >=
□ Check Serial Number
□ Check Identification

State Machine
☑ Auto Restore States
☑ Wait for WcState is Ok
☑ ReInit after Communication Error
☑ Log Communication Changes

Final State
○ OP ● SAFEOP in Config Mode
○ SAFEOP ○ PREOP ○ INIT

Process Data

An NC reset will not be performed. If an NC reset is necessary, it can be executed via the function block MC_Reset from the Tc2_MC2 PLC library.

Inputs

```
VAR_INPUT
    sNetId   : T_AmsNetId := '';
    bExecute : BOOL;
    tTimeout : TIME := DEFAULT_ADS_TIMEOUT;
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).
bExecute	BOOL	The function block is enabled via a positive edge at this input.
tTimeout	TIME	Maximum time allowed for the execution of the function block.

Inputs/outputs

```
VAR_IN_OUT
    Axis : NCTOPLC_AXIS_REF;(* reference to NC axis *)
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of the type NCTOPLC_AXIS_REF

Outputs

```
VAR_OUTPUT
    bBusy        : BOOL;
    bError       : BOOL;
    iAdsErrId    : UINT;
    iSercosErrId : UINT;
END_VAR
```

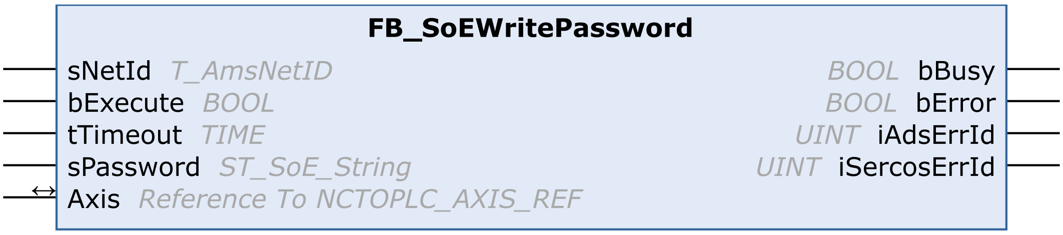
Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
iAdsErrId	UINT	Returns the ADS error code of the last executed command when the bError output is set.
iSercosErrId	UINT	In the case of a set bError output returns the Sercos error of the last executed command.

Sample

```
fbSoEReset : FB_SoEReset_ByDriveRef;
bSoEReset : BOOL;
(* NcAxis *)
NcToPlc AT %I* : NCTOPLC_Axis_REF;

IF bSoEReset THEN
 fbSoEReset(
   Axis := NcToPlc,
   bExecute := TRUE,
    tTimeout := DEFAULT_ADS_TIMEOUT,
 );
 IF NOT fbSoEReset.bBusy THEN
    fbSoEReset(Axis := NcToPlc, bExecute := FALSE);
    bSoEReset := FALSE;
 END_IF
END_IF
```

3.1.2 FB_SoEWritePassword



With the FB_SoEWritePassword function block (S-0-0267) the drive password can be set.

Inputs

```
VAR_INPUT
    sNetId    : T_AmsNetId := '';
    bExecute  : BOOL;
    tTimeout  : TIME := DEFAULT_ADS_TIMEOUT;
    sPassword : ST_SoE_String;
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).
bExecute	BOOL	The function block is enabled via a positive edge at this input.
tTimeout	TIME	Maximum time allowed for the execution of the function block.
sPassword	ST_SoE_String	Password as a Sercos string

Inputs/outputs

```
VAR_IN_OUT
    Axis : NCTOPLC_AXIS_REF;(* reference to NC axis *)
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of the type NCTOPLC_AXIS_REF

Outputs

```
VAR_OUTPUT
    bBusy        : BOOL;
    bError       : BOOL;
    iAdsErrId    : UINT;
    iSercosErrId : UINT;
END_VAR
```

Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
iAdsErrId	UINT	Returns the ADS error code of the last executed command when the bError output is set.
iSercosErrId	UINT	In the case of a set bError output returns the Sercos error of the last executed command.

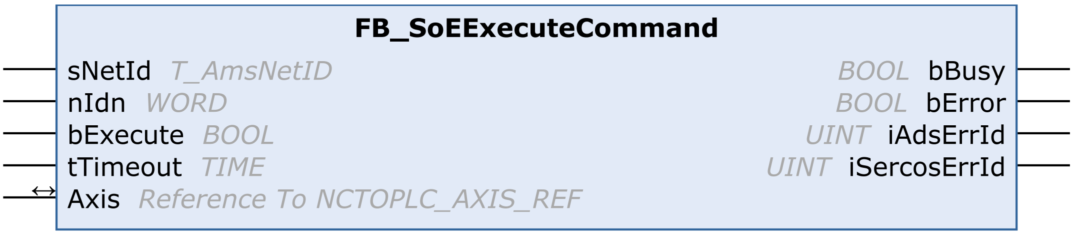
Sample

```
fbWritePassword : FB_SoEWritePassword;
bWritePassword : BOOL;
sPassword : ST_SoE_String;
(* NcAxis *)
NcToPlc AT %I* : NCTOPLC_AXIS_REF;

IF bWritePassword THEN
  fbWritePassword(
    Axis := NcToPlc,
    bExecute := TRUE,
    tTimeout := DEFAULT_ADS_TIMEOUT,
    sPassword := sPassword
  );
  IF NOT fbWritePassword.bBusy THEN
    fbWritePassword(Axis := NcToPlc, bExecute := FALSE);
    bWritePassword := FALSE;
  END_IF
END_IF
```

3.1.3 Function blocks for commands

3.1.3.1 FB_SoEExecuteCommand



With the FB_SoEExecuteCommand function block a command can be executed.

Inputs

```
VAR_INPUT
    sNetId   : T_AmsNetId := '';
    nIdn     : WORD;
    bExecute : BOOL;
    tTimeout : TIME := DEFAULT_ADS_TIMEOUT;
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).
nIdn	WORD	Parameter number to which FB_SoEExecuteCommand refers, e.g. "P_0_IDN + 160" for P-0-0160.
bExecute	BOOL	The function block is enabled via a positive edge at this input.
tTimeout	TIME	Maximum time allowed for the execution of the function block.

Inputs/outputs

```
VAR_IN_OUT
    Axis : NCTOPLC_AXIS_REF;(* reference to NC axis *)
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of the type NCTOPLC_AXIS_REF

Outputs

```
VAR_OUTPUT
    bBusy        : BOOL;
    bError       : BOOL;
    iAdsErrId    : UINT;
    iSercosErrId : UINT;
END_VAR
```

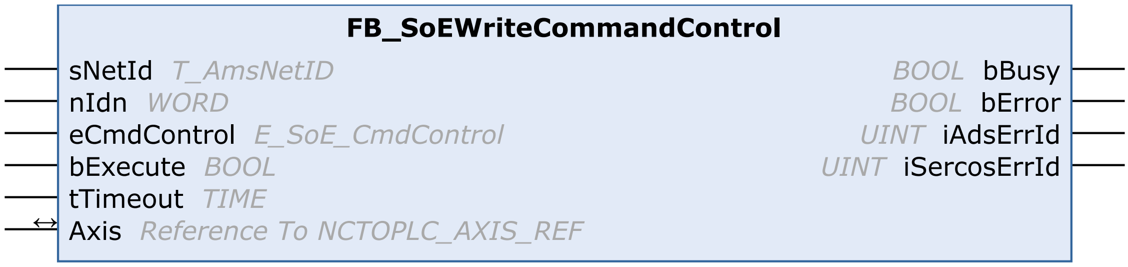
Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
iAdsErrId	UINT	Returns the ADS error code of the last executed command when the bError output is set.
iSercosErrId	UINT	In the case of a set bError output returns the Sercos error of the last executed command.

Sample

```
fbExecuteCommand : FB_SoEExecuteCommand;
bExecuteCommand : BOOL;
nIdn : WORD;
(* NcAxis *)
NcToPlc AT %I* : NCTOPLC_AXIS_REF;

IF bExecuteCommand THEN
 nIdn := P_0_IDN + 160;
 fbExecuteCommand(
    Axis := NcToPlc,
    bExecute := TRUE,
    tTimeout := DEFAULT_ADS_TIMEOUT,
    nIdn := nIdn,
 );
 IF NOT fbExecuteCommand.bBusy THEN
    fbExecuteCommand(Axis := NcToPlc, bExecute := FALSE);
    bExecuteCommand := FALSE;
 END_IF
END_IF
```

3.1.3.2 FB_SoEWriteCommandControl



With the FB_SoEWriteCommandControl function block a command can be prepared, started or canceled.

Inputs

```
VAR_INPUT
    sNetId      : T_AmsNetId := '';
    nIdn        : WORD;
    eCmdControl : E_SoE_CmdControl;
    bExecute    : BOOL;
    tTimeout    : TIME := DEFAULT_ADS_TIMEOUT;
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).
nIdn	WORD	Parameter number to which FB_SoEExecuteCommand refers, e.g. "P_0_IDN + 160" for P-0-0160.
eCmdControl	E_SoE_CmdControl	Indicates, if a command should be prepared (eSoE_CmdControl_Set := 1), executed (eSoE_CmdControl_SetAndEnable := 3) or aborted (eSoE_CmdControl_Cancel := 0).
bExecute	BOOL	The function block is enabled via a positive edge at this input.
tTimeout	TIME	Maximum time allowed for the execution of the function block.

Inputs/outputs

```
VAR_IN_OUT
    Axis : NCTOPLC_AXIS_REF;(* reference to NC axis *)
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of the type NCTOPLC_AXIS_REF

Outputs

```
VAR_OUTPUT
    bBusy        : BOOL;
    bError       : BOOL;
    iAdsErrId    : UINT;
    iSercosErrId : UINT;
END_VAR
```

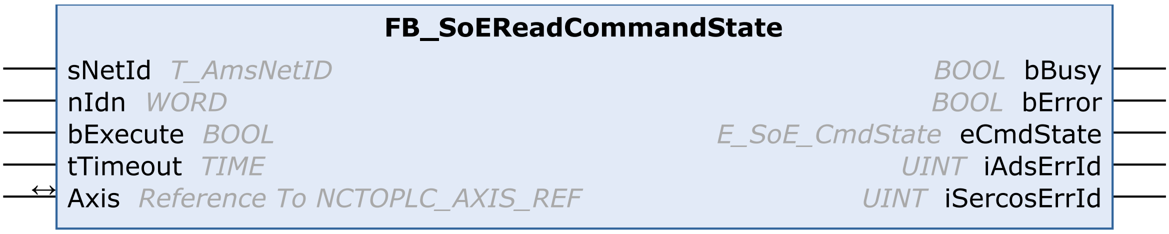
Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
iAdsErrId	UINT	Returns the ADS error code of the last executed command when the bError output is set.
iSercosErrId	UINT	In the case of a set bError output returns the Sercos error of the last executed command.

Sample

```
fbWriteCommandControl : FB_SoEWriteCommandControl;
bWriteCommandControl : BOOL;
nIdn : WORD;
eCmdControl : E_SoE_CmdControl;
(* NcAxis *)
NcToPlc AT %I* : NCTOPLC_AXIS_REF;

IF bWriteCommandControl THEN
 nIdn := P_0_IDN + 160;
 fbWriteCommandControl(
    Axis := NcToPlc,
    bExecute := TRUE,
    tTimeout := DEFAULT_ADS_TIMEOUT,
    nIdn := nIdn,
    eCmdControl := eCmdControl
 );
 IF NOT fbWriteCommandControl.bBusy THEN
    fbWriteCommandControl(Axis := NcToPlc, bExecute := FALSE);
    bWriteCommandControl := FALSE;
 END_IF
END_IF
```

3.1.3.3 FB_SoEReadCommandState



With the FB_SoEReadCommandState function block the command execution can be checked.

Inputs

```
VAR_INPUT
    sNetId   : T_AmsNetId := '';
    nIdn     : WORD;
    bExecute : BOOL;
    tTimeout : TIME := DEFAULT_ADS_TIMEOUT;
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).
nIdn	WORD	Parameter number to which FB_SoEExecuteCommand refers, e.g. "P_0_IDN + 160" for P-0-0160.
bExecute	BOOL	The function block is enabled via a positive edge at this input.
tTimeout	TIME	Maximum time allowed for the execution of the function block.

Inputs/outputs

```
VAR_IN_OUT
    Axis : NCTOPLC_AXIS_REF;(* reference to NC axis *)
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of the type NCTOPLC_AXIS_REF

Outputs

```
VAR_OUTPUT
    bBusy         : BOOL;
    bError        : BOOL;
    eCmdState     : E_SoE_CmdState;
    iAdsErrId     : UINT;
    iSercosErrId  : UINT;
END_VAR
```

Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
eCmdState	E_SoE_CmdState	Returns the command state (see below).
iAdsErrId	UINT	Returns the ADS error code of the last executed command when the bError output is set.
iSercosErrId	UINT	In the case of a set bError output returns the Sercos error of the last executed command.

```
eSoE_CmdState_NotSet = 0
- kein Kommando aktiv

eSoE_CmdState_Set = 1
- Kommando gesetzt (vorbereitet) aber (noch) nicht ausgeführt

eSoE_CmdState_Executed = 2
- Kommando wurde ausgeführt

eSoE_CmdState_SetEnabledExecuted = 3
- Kommando gesetzt (vorbereitet) und ausgeführt

eSoE_CmdState_SetAndInterrupted = 5
- Kommando wurde gesetzt aber unterbrochen

eSoE_CmdState_SetEnabledNotExecuted = 7
- Kommandoausführung ist noch aktiv

eSoE_CmdState_Error = 15
- Fehler bei der Kommandoausführung, es wurde in den Fehlerstate gewechselt
```

Sample

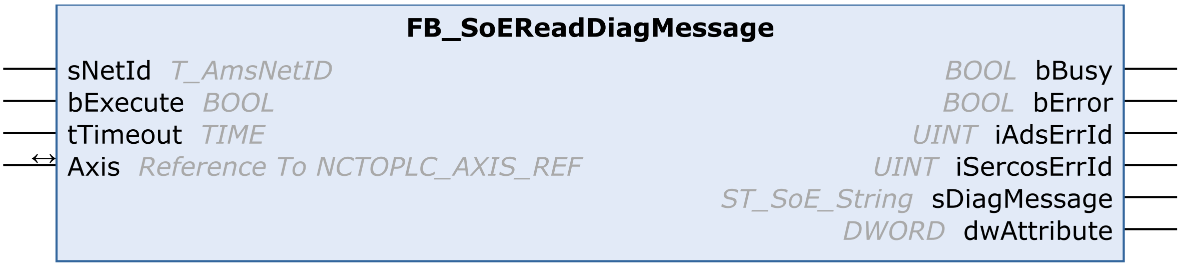
```
fbReadCommandState : FB_SoEReadCommandState;
bReadCommandState : BOOL;
nIdn : WORD;
```

```
eCmdState : E_SoE_CmdState;
(* NcAxis *)
NcToPlc AT %I* : NCTOPLC_AXIS_REF;

IF bReadCommandState THEN
 nIdn := P_0_IDN + 160;
 fbReadCommandState(
    Axis := NcToPlc,
    bExecute := TRUE,
    tTimeout := DEFAULT_ADS_TIMEOUT,
    nIdn := nIdn,
    eCmdState => eCmdState
 );
 IF NOT fbReadCommandState.bBusy THEN
    fbReadCommandState(Axis := NcToPlc, bExecute := FALSE);
    bReadCommandState := FALSE;
 END_IF
END_IF
```

3.1.4 Function blocks for diagnostics

3.1.4.1 FB_SoEReadDiagMessage



With the FB_SoEReadDiagMessage function block the diagnosis message can be read out as a Sercos string (S-0-0095).

Inputs

```
VAR_INPUT
    sNetId   : T_AmsNetId := '';
    bExecute : BOOL;
    tTimeout : TIME := DEFAULT_ADS_TIMEOUT;
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).
bExecute	BOOL	The function block is enabled via a positive edge at this input.
tTimeout	TIME	Maximum time allowed for the execution of the function block.

Inputs/outputs

```
VAR_IN_OUT
    Axis : NCTOPLC_AXIS_REF;(* reference to NC axis *)
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of the type NCTOPLC_AXIS_REF

Outputs

```
VAR_OUTPUT
    bBusy        : BOOL;
    bError       : BOOL;
    iAdsErrId     : UINT;
    iSercosErrId : UINT;
    sDiagMessage : ST_SoE_String;
    dwAttribute  : DWORD;
END_VAR
```

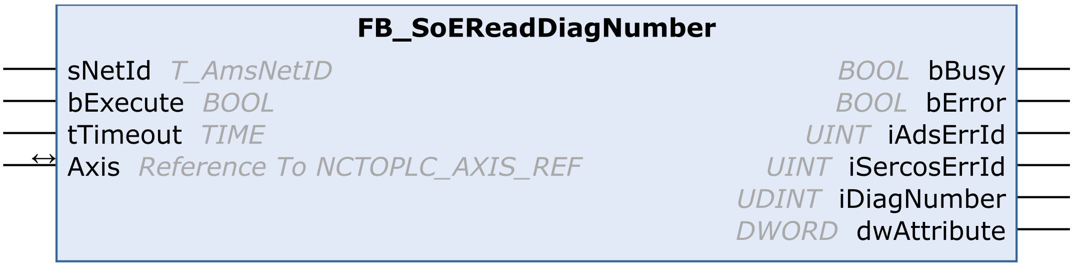
Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
iAdsErrId	UINT	Returns the ADS error code of the last executed command when the bError output is set.
iSercosErrId	UINT	In the case of a set bError output returns the Sercos error of the last executed command.
sDiagMessage	ST_SoE_String	Returns the diagnosis message.
dwAttribute	DWORD	Returns the attributes of the Sercos parameter.

Sample

```
fbDiagMessage : FB_SoEReadDiagMessage;
bDiagMessage : BOOL;
sDiagMessage : ST_SoE_String;
(* NcAxis *)
NcToPlc AT %I* : NCTOPLC_AXIS_REF;

IF bDiagMessage THEN
 fbDiagMessage(
    Axis := NcToPlc,
    bExecute := TRUE,
    tTimeout := DEFAULT_ADS_TIMEOUT,
    sDiagMessage=> sDiagMessage
 );
 IF NOT fbDiagMessage.bBusy THEN
    fbDiagMessage(Axis := NcToPlc, bExecute := FALSE);
    bDiagMessage := FALSE;
 END_IF
END_IF
```

3.1.4.2 FB_SoEReadDiagNumber



With the FB_SoEReadDiagNumber function block the current diagnostic number can be read out as UDINT (S-0-0390).

Inputs

```
VAR_INPUT
    sNetId   : T_AmsNetId := '';
    bExecute : BOOL;
    tTimeout : TIME := DEFAULT_ADS_TIMEOUT;
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).

Name	Type	Description
bExecute	BOOL	The function block is enabled via a positive edge at this input.
tTimeout	TIME	Maximum time allowed for the execution of the function block.

Inputs/outputs

```
VAR_IN_OUT
    Axis : NCTOPLC_AXIS_REF;(* reference to NC axis *)
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of the type NCTOPLC_AXIS_REF

Outputs

```
VAR_OUTPUT
    bBusy        : BOOL;
    bError       : BOOL;
    iAdsErrId    : UINT;
    iSercosErrId : UINT;
    iDiagNumber  : UDINT;
    dwAttribute  : DWORD;
END_VAR
```

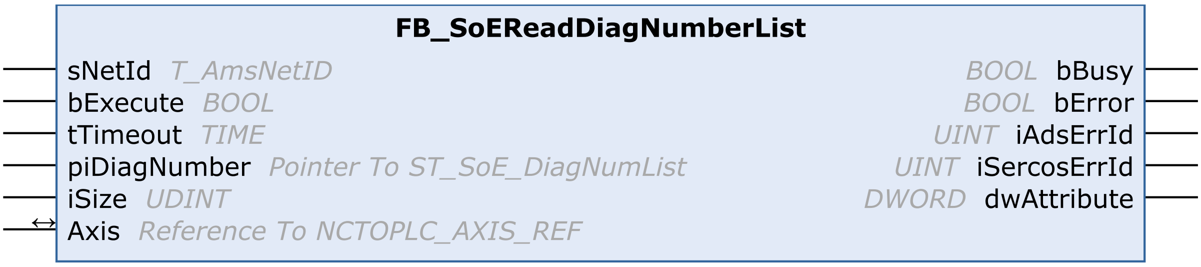
Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
iAdsErrId	UINT	Returns the ADS error code of the last executed command when the bError output is set.
iSercosErrId	UINT	In the case of a set bError output returns the Sercos error of the last executed command.
iDiagNumber	UDINT	Returns the attributes of the Sercos parameter.
dwAttribute	DWORD	Returns the current diagnostic number.

Sample

```
fbDiagNumber : FB_SoEReadDiagNumber;
bDiagNumber : BOOL;
iDiagNumber : UDINT;
(* NcAxis *)
NcToPlc AT %I* : NCTOPLC_AXIS_REF;

IF bDiagNumber THEN
 fbDiagNumber(
    Axis := NcToPlc,
    bExecute := TRUE,
    tTimeout := DEFAULT_ADS_TIMEOUT,
    iDiagNumber => iDiagNumber
 );
 IF NOT fbDiagNumber.bBusy THEN
    fbDiagNumber(Axis := NcToPlc, bExecute := FALSE);
    bDiagNumber := FALSE;
 END_IF
END_IF
```

3.1.4.3 FB_SoEReadDiagNumberList



With the FB_SoEReadDiagNumberList function block a history of the diagnosis numbers can be read out as a list (S-0-0375).

Inputs

```
VAR_INPUT
    sNetId       : T_AmsNetId := '';
    bExecute     : BOOL;
    tTimeout     : TIME := DEFAULT_ADS_TIMEOUT;
    piDiagNumber : POINTER TO ST_SoE_DiagNumList;
    iSize        : UDINT;
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).
bExecute	BOOL	The function block is enabled via a positive edge at this input.
tTimeout	TIME	Maximum time allowed for the execution of the function block.
piDiagNumber	POINTER TO ST_SoE_DiagNumList	Pointer to the list of the last max. 30 error numbers. The list consists of the current and maximum number of bytes in the list as well as the 30 list items.
iSize	UDINT	Size of the list in bytes (as Sizeof())

Inputs/outputs

```
VAR_IN_OUT
    Axis : NCTOPLC_AXIS_REF;(* reference to NC axis *)
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of the type NCTOPLC_AXIS_REF

Outputs

```
VAR_OUTPUT
    bBusy        : BOOL;
    bError       : BOOL;
    iAdsErrId    : UINT;
    iSercosErrId : UINT;
    dwAttribute  : DWORD;
END_VAR
```

Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
iAdsErrId	UINT	Returns the ADS error code of the last executed command when the bError output is set.

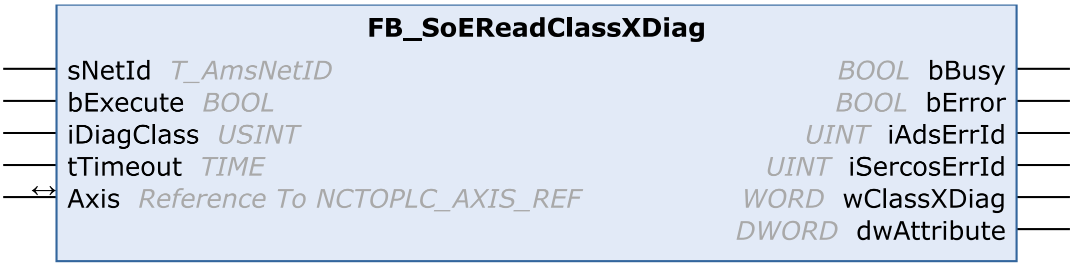
Name	Type	Description
iSercosErrId	UINT	In the case of a set bError output returns the Sercos error of the last executed command.
dwAttribute	DWORD	Returns the attributes of the Sercos parameter.

Sample

```
fbDiagNumberList : FB_SoEReadDiagNumberList;
bDiagNumberList : BOOL;
stDiagNumberList : ST_SoE_DiagNumList;
(* NcAxis *)
NcToPlc AT %I* : NCTOPLC_AXIS_REF;

IF bDiagNumberList THEN
 fbDiagNumberList(
    Axis := NcToPlc,
    bExecute := TRUE,
    tTimeout := DEFAULT_ADS_TIMEOUT,
    piDiagNumber:= ADR(stDiagNumberList),
    iSize := SIZEOF(stDiagNumberList),
 );
 IF NOT fbDiagNumberList.bBusy THEN
    fbDiagNumberList(Axis := NcToPlc, bExecute := FALSE);
    bDiagNumberList := FALSE;
 END_IF
END_IF
```

3.1.4.4 FB_SoEReadClassXDiag



With the function block FB_SoEReadClassXDiag, the current Class 1 diagnosis (S-0-0011) ... Class 3 diagnosis (S-0-0013) can be read out as WORD. There is the conversion function F_ConvWordToSTAX5000C1D for the evaluation of the Class 1 diagnosis as a structure ST_AX5000_C1D, (see TwinCAT 3 PLC Lib Tc2_Drive documentation).

Inputs

```
VAR_INPUT
    sNetId     : T_AmsNetId := '';
    bExecute   : BOOL;
    iDiagClass : USINT:= 1; (* 1: C1D (S-0-0011) is default, 2: C2D (S-0-0012), 3: C3D (S-0-0013) *)
    tTimeout   : TIME := DEFAULT_ADS_TIMEOUT;
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).
bExecute	BOOL	The function block is enabled via a positive edge at this input.
iDiagClass	USINT	Specifies which diagnosis should be read. The diagnostics parameters may vary from vendor to vendor. All diagnostics parameters (C1D ... C3D) or all bits are not always implemented in them. 1: Error: Class 1 Diag (S-0-0011) 2: Warnings: Class 2 Diag (S-0-0012) 3: Information: Class 3 Diag (S-0-0013)

Name	Type	Description
tTimeout	TIME	Maximum time allowed for the execution of the function block.

Inputs/outputs

```
VAR_IN_OUT
    Axis : NCTOPLC_AXIS_REF;(* reference to NC axis *)
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of the type NCTOPLC_AXIS_REF

Outputs

```
VAR_OUTPUT
    bBusy        : BOOL;
    bError       : BOOL;
    iAdsErrId    : UINT;
    iSercosErrId : UINT;
    wClassXDiag  : WORD;
    dwAttribute  : DWORD;
END_VAR
```

Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
iAdsErrId	UINT	Returns the ADS error code of the last executed command when the bError output is set.
iSercosErrId	UINT	In the case of a set bError output returns the Sercos error of the last executed command.
wClassXDiag	WORD	Returns the current Class X diagnosis.
dwAttribute	DWORD	Returns the attributes of the Sercos parameter.

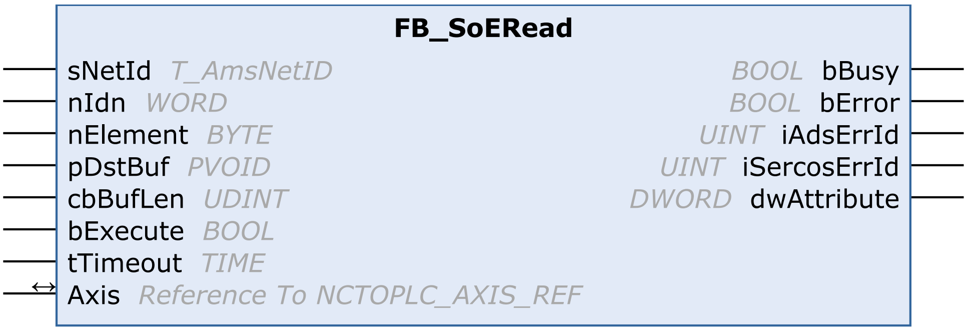
Sample

```
fbClassXDiag : FB_SoEReadClassXDiag;
bClassXDiag : BOOL;
iDiagClass : USINT := 1;
wClass1Diag : WORD;
stAX5000C1D : ST_AX5000_C1D;
wClass2Diag : WORD;
(* NcAxis *)
NcToPlc AT %I* : NCTOPLC_AXIS_REF;

IF bClassXDiag THEN
 fbClassXDiag(
    Axis := NcToPlc,
    bExecute := TRUE,
    iDiagClass := iDiagClass,
    tTimeout := DEFAULT_ADS_TIMEOUT
 );
 IF NOT fbClassXDiag.bBusy THEN
    fbClassXDiag(Axis := NcToPlc, bExecute := FALSE);
    bClassXDiag := FALSE;
    CASE fbClassXDiag.iDiagClass OF
    1:
     wClass1Diag := fbClassXDiag.wClassXDiag;
     stAX5000C1D := F_ConvWordToSTAX5000C1D(wClass1Diag);
    2:
     wClass2Diag := fbClassXDiag.wClassXDiag;
    END_CASE
 END_IF
END_IF
```

3.1.5 Function blocks for determining current values

3.1.5.1 FB_SoERead



With the FB_SoERead function block a parameter can be read.

Inputs

```
VAR_INPUT
    sNetId    : T_AmsNetId := '';
    nIdn      : WORD;
    nElement  : BYTE;
    pDstBuf   : PVOID;
    cbBufLen  : UDINT;
    bExecute  : BOOL;
    tTimeout  : TIME := DEFAULT_ADS_TIMEOUT;
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).
nIdn	WORD	Parameter number to which FB_SoERead refers, e.g. "S_0_IDN + 33" for S-0-0033.

nElement	BYTE	Specifies which part of the parameter should be accessed, e.g. 16#40 is the value (Value) of the parameter.
		EC_SOE_ELEMENT_DATASTATE :BYTE :=16#01;
		EC_SOE_ELEMENT_NAME :BYTE :=16#02;
		EC_SOE_ELEMENT_ATTRIBUTE :BYTE :=16#04;
		EC_SOE_ELEMENT_UNIT :BYTE :=16#08;
		EC_SOE_ELEMENT_MIN :BYTE :=16#10;
		EC_SOE_ELEMENT_MAX :BYTE :=16#20;
		EC_SOE_ELEMENT_VALUE :BYTE :=16#40;
		EC_SOE_ELEMENT_DEFAULT :BYTE :=16#80;

pDstBuf	PVOID	ADR() of the variables to which the value should be read.
cbBufLen	UDINT	SIZEOF() of the variables to which the value should be read.
bExecute	BOOL	The function block is enabled via a positive edge at this input.
tTimeout	TIME	Maximum time allowed for the execution of the function block.

Inputs/outputs

```
VAR_IN_OUT
    Axis : NCTOPLC_AXIS_REF;(* reference to NC axis *)
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of the type NCTOPLC_AXIS_REF

Outputs

```
VAR_OUTPUT
    bBusy         : BOOL;
    bError        : BOOL;
    iAdsErrId     : UINT;
    iSercosErrId : UINT;
    dwAttribute   : DWORD;
END_VAR
```

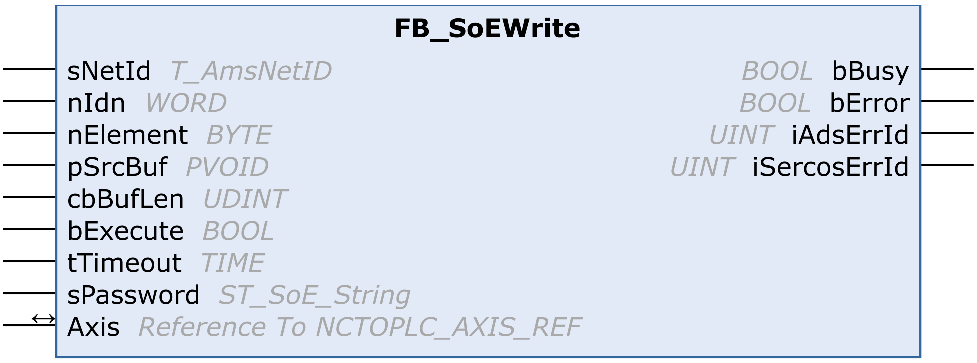
Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
iAdsErrId	UINT	Returns the ADS error code of the last executed command when the bError output is set.
iSercosErrId	UINT	In the case of a set bError output returns the Sercos error of the last executed command.
dwAttribute	DWORD	Returns the attributes of the Sercos parameter.

Sample

```
fbRead       : FB_SoERead;
bRead        : BOOL;
iReadValue   : UINT;
nIdn         : WORD;
bReadValue   : UINT;
(* NcAxis *)
NcToPlc AT %I* : NCTOPLC_AXIS_REF;

IF bRead THEN
 nIdn := S_0_IDN + 33;
 fbRead(
    Axis     := NcToPlc,
    nIdn     := nIdn,
    nElement := 16#40,
    pDstBuf  := ADR(iReadValue),
    cbBufLen := SIZEOF(iReadValue),
    bExecute := TRUE,
    tTimeout := DEFAULT_ADS_TIMEOUT,
 );
 IF NOT fbRead.bBusy THEN
    fbRead(Axis := NcToPlc, bExecute := FALSE);
    bRead       := FALSE;
 END_IF
END_IF
```

3.1.5.2 FB_SoEWrite



With the FB_SoEWrite function block a parameter can be written.

Inputs

```
VAR_INPUT
    sNetId    : T_AmsNetId := '';
    nIdn      : WORD;
    nElement  : BYTE;
    pSrcBuf   : PVOID;
    cbBufLen  : UDINT;
    bExecute  : BOOL;
    tTimeout  : TIME := DEFAULT_ADS_TIMEOUT;
    sPassword : ST_SoE_String;
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).
nIdn	WORD	Parameter number to which FB_SoERead refers, e.g. "S_0_IDN + 47" for S-0-0047.

nElement	BYTE	Specifies which part of the parameter should be accessed, e.g. 16#40 is the value (Value) of the parameter. Usually there is only write access to the value, other components of the parameter are read-only.
		`EC_SOE_ELEMENT_DATASTATE :BYTE :=16#01;`
		`EC_SOE_ELEMENT_NAME :BYTE :=16#02;`
		`EC_SOE_ELEMENT_ATTRIBUTE :BYTE :=16#04;`
		`EC_SOE_ELEMENT_UNIT :BYTE :=16#08;`
		`EC_SOE_ELEMENT_MIN :BYTE :=16#10;`
		`EC_SOE_ELEMENT_MAX :BYTE :=16#20;`
		`EC_SOE_ELEMENT_VALUE :BYTE :=16#40;`
		`EC_SOE_ELEMENT_DEFAULT :BYTE :=16#80;`

pSrcBuf	PVOID	ADR() of the variable containing the value to be written.
cbBufLen	UDINT	SIZEOF() of the variable containing the value to be written.
bExecute	BOOL	The function block is enabled via a positive edge at this input.
tTimeout	TIME	Maximum time allowed for the execution of the function block.
sPassword	ST_SoE_String	Password as sercos string. Currently not used. The password must be written with FB_SoEWritePassword [▸ 11].

Inputs/outputs

```
VAR_IN_OUT
    Axis : NCTOPLC_AXIS_REF;(* reference to NC axis *)
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of the type NCTOPLC_AXIS_REF

Outputs

```
VAR_OUTPUT
    bBusy         : BOOL;
    bError        : BOOL;
    iAdsErrId     : UINT;
    iSercosErrId  : UINT;
END_VAR
```

Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.

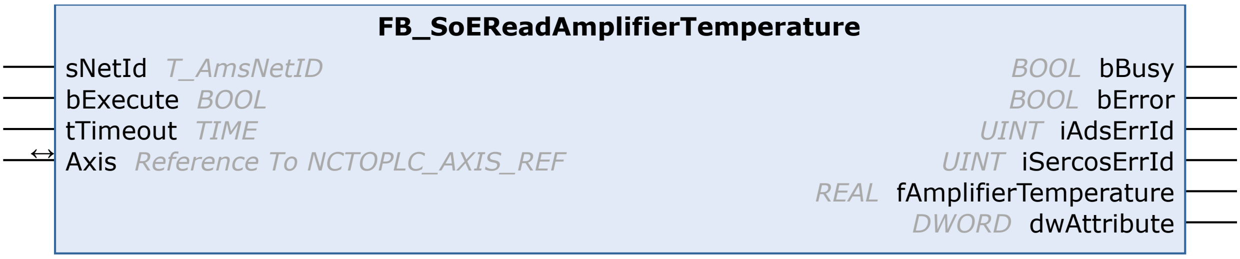
Name	Type	Description
iAdsErrId	UINT	Returns the ADS error code of the last executed command when the bError output is set.
iSercosErrId	UINT	In the case of a set bError output returns the Sercos error of the last executed command.

Sample

```
fbWrite        : FB_SoEWrite;
bWrite         : BOOL;
nIdn           : WORD;
iWriteValue    : UINT;
sPassword      : ST_SoE_String;
(* NcAxis *)
NcToPlc AT %I* : NCTOPLC_AXIS_REF;

IF bWrite THEN
 nIdn := S_0_IDN + 33;
 fbWrite(
    Axis      := NcToPlc,
    nIdn      := nIdn,
    nElement  := 16#40,
    pSrcBuf   := ADR(iWriteValue),
    cbBufLen  := SIZEOF(iWriteValue),
    sPassword := sPassword,
    bExecute  := TRUE,
    tTimeout  := DEFAULT_ADS_TIMEOUT,
 );
 IF NOT fbWrite.bBusy THEN
    fbWrite(Axis := NcToPlc, bExecute := FALSE);
    bWrite    := FALSE;
 END_IF
END_IF
```

3.1.5.3 FB_SoEReadAmplifierTemperature



With the FB_SoEReadAmplifierTemperature function block the temperature of the drive (S-0-0384) can be read.

Inputs

```
VAR_INPUT
    sNetId   : T_AmsNetId := '';
    bExecute : BOOL;
    tTimeout : TIME := DEFAULT_ADS_TIMEOUT;
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).
bExecute	BOOL	The function block is enabled via a positive edge at this input.
tTimeout	TIME	Maximum time allowed for the execution of the function block.

Inputs/outputs

```
VAR_IN_OUT
    Axis : NCTOPLC_AXIS_REF;(* reference to NC axis *)
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of the type NCTOPLC_AXIS_REF

Outputs

```
VAR_OUTPUT
    bBusy                  : BOOL;
    bError                 : BOOL;
    iAdsErrId              : UINT;
    iSercosErrId           : UINT;
    fAmplifierTemperature  : REAL;
    dwAttribute            : DWORD;
END_VAR
```

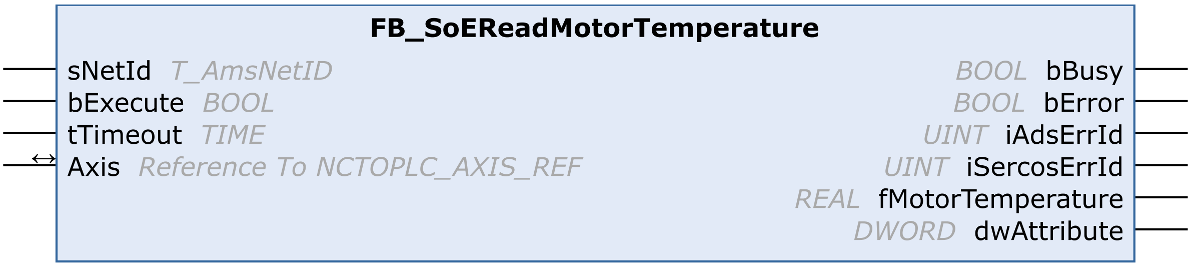
Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
iAdsErrId	UINT	Returns the ADS error code of the last executed command when the bError output is set.
iSercosErrId	UINT	In the case of a set bError output returns the Sercos error of the last executed command.
fAmplifierTemperature	REAL	Returns the drive temperature (e.g. 26.2 corresponds to 26.2 °C).
dwAttribute	DWORD	Returns the attributes of the Sercos parameter.

Sample

```
fbReadAmplifierTemp :
FB_SoEReadAmplifierTemperature;
bReadAmplifierTemp : BOOL;
fAmplifierTemperature : REAL;
(* NcAxis *)
NcToPlc AT %I* : NCTOPLC_AXIS_REF;

IF bReadAmplifierTemp THEN
 fbReadAmplifierTemp(
    Axis := NcToPlc,
    bExecute := TRUE,
    tTimeout := DEFAULT_ADS_TIMEOUT,
    fAmplifierTemperature=>fAmplifierTemperature
 );
 IF NOT fbReadAmplifierTemp.bBusy THEN
    fbReadAmplifierTemp(Axis := NcToPlc, bExecute := FALSE);
    bReadAmplifierTemp := FALSE;
 END_IF
END_IF
```

3.1.5.4 FB_SoEReadMotorTemperature



With the function block FB_SoEReadMotorTemperature the temperature of the motor (S-0-0383) can be read. If the motor does not contain a temperature sensor, this is 0.0, i.e. 0.0 °C.

Inputs

```
VAR_INPUT
    sNetId   : T_AmsNetId := '';
    bExecute : BOOL;
    tTimeout : TIME := DEFAULT_ADS_TIMEOUT;
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).
bExecute	BOOL	The function block is enabled via a positive edge at this input.
tTimeout	TIME	Maximum time allowed for the execution of the function block.

Inputs/outputs

```
VAR_IN_OUT
    Axis : NCTOPLC_AXIS_REF;(* reference to NC axis *)
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of the type NCTOPLC_AXIS_REF

Outputs

```
VAR_OUTPUT
    bBusy             : BOOL;
    bError            : BOOL;
    iAdsErrId         : UINT;
    iSercosErrId      : UINT;
    fMotorTemperature : REAL;
    dwAttribute       : DWORD;
END_VAR
```

Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
iAdsErrId	UINT	Returns the ADS error code of the last executed command when the bError output is set.
iSercosErrId	UINT	In the case of a set bError output returns the Sercos error of the last executed command.
fMotorTemperature	REAL	Returns the motor temperature (e.g. 30.5 corresponds to 30.5 °C). If the motor does not contain a temperature sensor, this is 0.0, i.e. 0.0 °C.

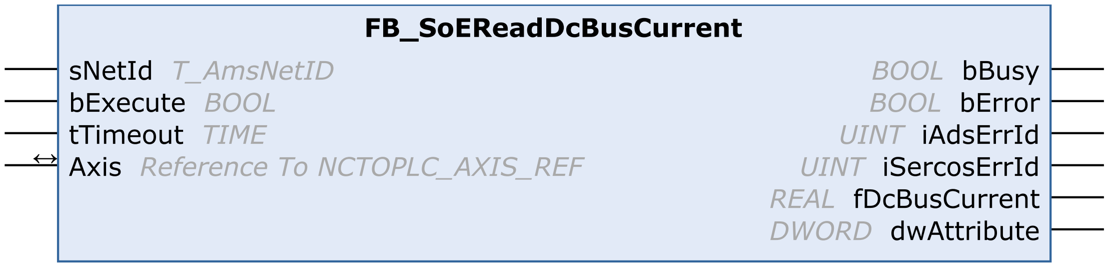
Name	Type	Description
dwAttribute	DWORD	Returns the attributes of the Sercos parameter.

Sample

```
fbReadMotorTemp : FB_SoEReadMotorTemperature;
bReadMotorTemp : BOOL;
fMotorTemperature : REAL;
(* NcAxis *)
NcToPlc AT %I* : NCTOPLC_AXIS_REF;

IF bReadMotorTemp AND NOT bInit THEN
 fbReadMotorTemp(
    Axis := NcToPlc,
    bExecute := TRUE,
    tTimeout := DEFAULT_ADS_TIMEOUT,
    fMotorTemperature=>fMotorTemperature
 );
 IF NOT fbReadMotorTemp.bBusy THEN
    fbReadMotorTemp(Axis := NcToPlc, bExecute := FALSE);
    bReadMotorTemp := FALSE;
 END_IF
END_IF
```

3.1.5.5 FB_SoEReadDcBusCurrent



With the function block FB_SoEAX5000ReadDcBusCurrent the DC-Bus current
(S-0-0381) can be read.

Inputs

```
VAR_INPUT
    sNetId   : T_AmsNetId := '';
    bExecute : BOOL;
    tTimeout : TIME := DEFAULT_ADS_TIMEOUT;
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).
bExecute	BOOL	The function block is enabled via a positive edge at this input.
tTimeout	TIME	Maximum time allowed for the execution of the function block.

Inputs/outputs

```
VAR_IN_OUT
    Axis : NCTOPLC_AXIS_REF;(* reference to NC axis *)
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of the type NCTOPLC_AXIS_REF

Outputs

```
VAR_OUTPUT
    bBusy         : BOOL;
    bError        : BOOL;
    iAdsErrId     : UINT;
    iSercosErrId  : UINT;
    fDcBusCurrent : REAL;
    dwAttribute   : DWORD;
END_VAR
```

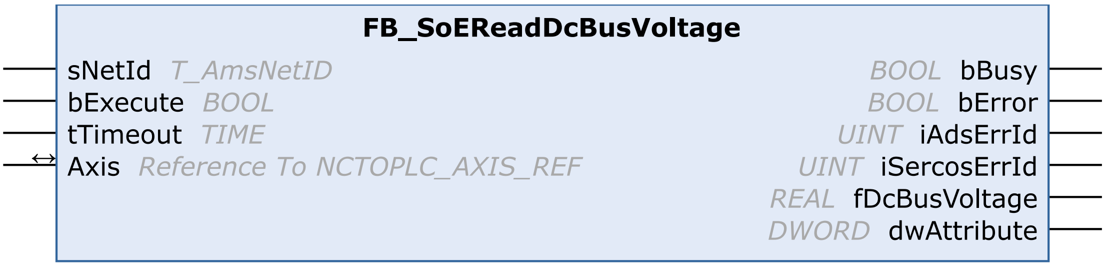
Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
iAdsErrId	UINT	Returns the ADS error code of the last executed command when the bError output is set.
iSercosErrId	UINT	In the case of a set bError output returns the Sercos error of the last executed command.
fDcBusCurrent	REAL	Returns the attributes of the Sercos parameter.
dwAttribute	DWORD	Returns the DC bus current (e.g. 2,040 equals 2,040 A).

Sample

```
fbReadDcBusCurrent : FB_SoEReadDcBusCurrent_ByDriveRef;
bReadDcBusCurrent : BOOL;
fDcBusCurrent : REAL;
(* NcAxis *)
NcToPlc AT %I* : NCTOPLC_AXIS_REF;

IF bReadDcBusCurrent THEN
 fbReadDcBusCurrent(
    Axis := NcToPlc,
    bExecute := TRUE,
    tTimeout := DEFAULT_ADS_TIMEOUT,
    fDcBusCurrent=>fDcBusCurrent
 );
 IF NOT fbReadDcBusCurrent.bBusy THEN
    fbReadDcBusCurrent(Axis := NcToPlc, bExecute := FALSE);
    bReadDcBusCurrent := FALSE;
 END_IF
END_IF
```

3.1.5.6 FB_SoEReadDcBusVoltage



With the FB_SoEReadDcBusVoltage function block the Dc-Bus voltage of the drive (S-0-0380) can be read.

Inputs

```
VAR_INPUT
    sNetId   : T_AmsNetId := '';
    bExecute : BOOL;
    tTimeout : TIME := DEFAULT_ADS_TIMEOUT;
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).
bExecute	BOOL	The function block is enabled via a positive edge at this input.
tTimeout	TIME	Maximum time allowed for the execution of the function block.

Inputs/outputs

```
VAR_IN_OUT
    Axis : NCTOPLC_AXIS_REF;(* reference to NC axis *)
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of the type NCTOPLC_AXIS_REF

Outputs

```
VAR_OUTPUT
    bBusy         : BOOL;
    bError        : BOOL;
    iAdsErrId     : UINT;
    iSercosErrId  : UINT;
    fDcBusVoltage : REAL;
    dwAttribute   : DWORD;
END_VAR
```

Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
iAdsErrId	UINT	Returns the ADS error code of the last executed command when the bError output is set.
iSercosErrId	UINT	In the case of a set bError output returns the Sercos error of the last executed command.
fDcBusVoltage	REAL	Returns the DC-Bus voltage (e.g. 294.0 corresponds to 294.0 V).
dwAttribute	DWORD	Returns the attributes of the Sercos parameter.

Sample

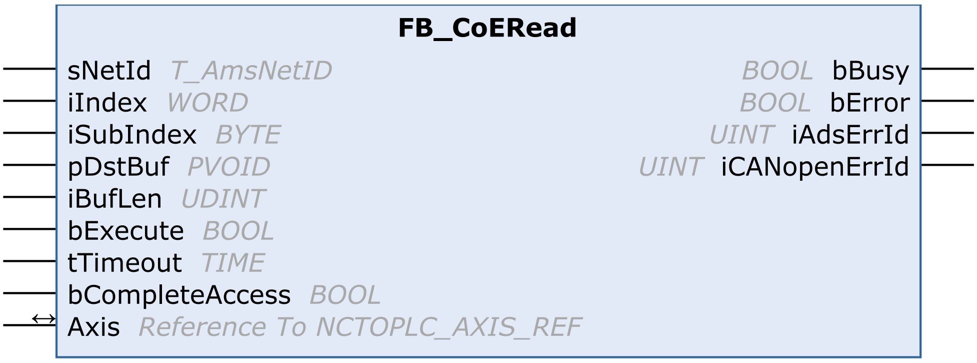
```
fbReadDcBusVoltage : FB_SoEReadDcBusVoltage;
bReadDcBusVoltage : BOOL;
fDcBusVoltage : REAL;
(* NcAxis *)
NcToPlc AT %I* : NCTOPLC_AXIS_REF;

IF bReadDcBusVoltage THEN
 fbReadDcBusVoltage(
    Axis := NcToPlc,
    bExecute := TRUE,
    tTimeout := DEFAULT_ADS_TIMEOUT,
    fDcBusVoltage=>fDcBusVoltage
 );
 IF NOT fbReadDcBusVoltage.bBusy THEN
    fbReadDcBusVoltage(Axis := NcToPlc, bExecute := FALSE);
    bReadDcBusVoltage := FALSE;
 END_IF
END_IF
```

3.2 General CoE

3.2.1 Function blocks for determining current values

3.2.1.1 FB_CoERead



The function block FB_CoERead allows data to be read from an object directory of an EtherCAT slave through an SDO (Service Data Object) access. This requires the slave to have a mailbox and to support the CoE (CANopen over EtherCAT) protocol. With the help of the SubIndex and Index parameters a selection is made as to which object should be read out. Via CompleteAccess := TRUE the parameter can be read with sub-elements.

Inputs

```
VAR_INPUT
   sNetId          : T_AmsNetId;(*netID of PC with NC*)
   iIndex          : WORD;(*CoE object index*)
   iSubIndex       : BYTE;(*CoE sub index*)
   pDstBuf         : PVOID;(*Contains the address of the buffer for the received data*)
   iBufLen         : UDINT;(*Contains the max. number of bytes to be received*)
   bExecute        : BOOL;(*Function block execution is triggered by a rising edge at this input.*)
   tTimeout        : TIME := DEFAULT_ADS_TIMEOUT;
(*States the time before the function is cancelled.*)
   bCompleteAccess : BOOL;(*Function block reads the complete object with all sub index*)
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).
iIndex	WORD	Index of the object that is to be read.
iSubIndex	BYTE	Subindex of the object that is to be read.
pDstBuf	PVOID	Address (pointer) to the receive buffer
iBufLen	UDINT	Maximum available buffer size (in bytes) for the data to be read
bExecute	BOOL	The function block is enabled via a positive edge at this input.
tTimeout	TIME	Maximum time allowed for the execution of the function block.
bCompleteAccess	BOOL	Via Complete Access the complete object can be accessed at once.

Inputs/outputs

```
VAR_IN_OUT
    Axis  :  NCTOPLC_AXIS_REF;
END_VAR
```

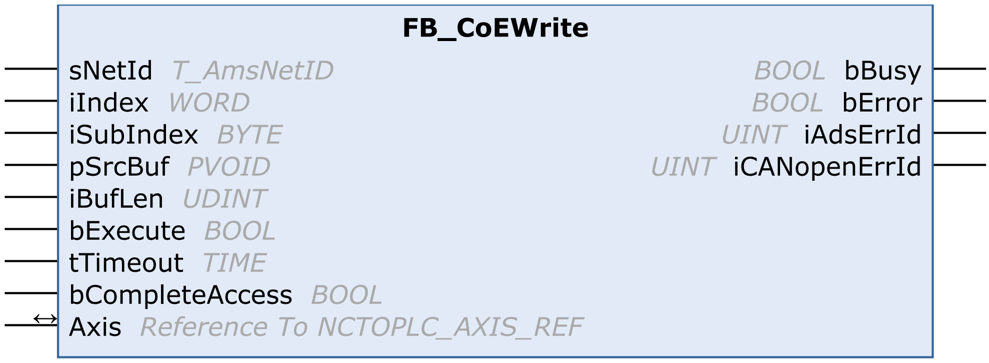
Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of type NCTOPLC_AXIS_REF, which addresses an axis uniquely within the system. Among other things it contains the current state of the axis such as the position, the velocity and the error state.

Outputs

```
VAR_OUTPUT
    bBusy         : BOOL;
    bError        : BOOL;
    iAdsErrId     : UINT;
    iCANopenErrId : UINT;
END_VAR
```

Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
iAdsErrId	UINT	In the event of a set bError output returns the ADS error code.
iCANopen ErrId	UINT	In the event of a set bError output returns the CANopen error code.

3.2.1.2 FB_CoEWrite



With the function block FB_CoEWrite, an object from the object directory of an EtherCAT slave can be written via an SDO (Service Data Object) download. This requires the slave to have a mailbox and to support the CoE (CANopen over EtherCAT) protocol. With the help of the SubIndex and Index parameters a selection is made as to which object should be written. Via CompleteAccess := TRUE the parameter can be written with sub-elements.

Inputs

```
VAR_INPUT
   sNetId          : T_AmsNetId;(*netID of PC with NC*)
   iIndex          : WORD;(*CoE object index*)
   iSubIndex       : BYTE;(*CoE sub index*)
   pDstBuf         : PVOID;(*Contains the address of the buffer for the received data*)
   iBufLen         : UDINT;(*Contains the max. number of bytes to be received*)
   bExecute        : BOOL;(*Function block execution is triggered by a rising edge at this input.*)
   tTimeout        : TIME := DEFAULT_ADS_TIMEOUT;
(*States the time before the function is cancelled.*)
   bCompleteAccess : BOOL;(*Function block reads the complete object with all sub index*)
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).
iIndex	WORD	Index of the object that is supposed to be written.
iSubIndex	BYTE	Subindex of the object that is supposed to be written.

Name	Type	Description
pDstBuf	PVOID	Address (pointer) to the transmit buffer
iBufLen	UDINT	Maximum available buffer size (in bytes) for the data to be read
bExecute	BOOL	The function block is enabled via a positive edge at this input.
tTimeout	TIME	Maximum time allowed for the execution of the function block.
bCompleteAccess	BOOL	Via Complete Access the complete object can be accessed at once.

Inputs/outputs

```
VAR_IN_OUT
    Axis   :  NCTOPLC_AXIS_REF;
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of type NCTOPLC_AXIS_REF, which addresses an axis uniquely within the system. Among other things it contains the current state of the axis such as the position, the velocity and the error state.

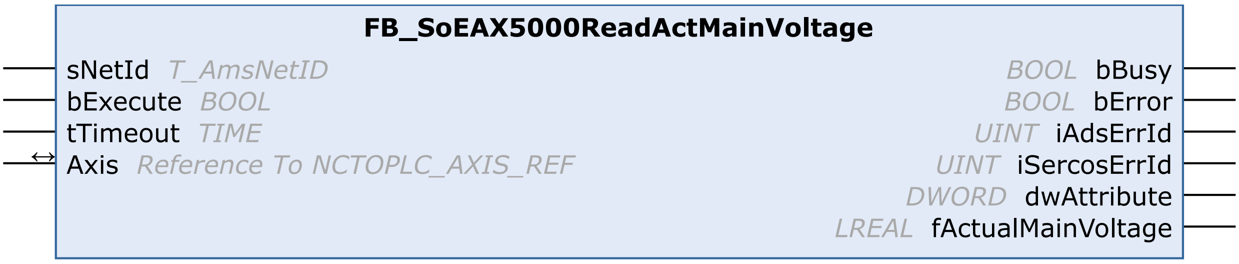
Outputs

```
VAR_OUTPUT
    bBusy         : BOOL;
    bError        : BOOL;
    iAdsErrId     : UINT;
    iCANopenErrId : UINT;
END_VAR
```

Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
iAdsErrId	UINT	In the event of a set bError output returns the ADS error code.
iCANopenErrId	UINT	In the event of a set bError output returns the CANopen error code.

3.3 AX5000 SoE

3.3.1 FB_SoEAX5000ReadActMainVoltage



With the FB_SoEAX5000ReadActMainVoltage function block the current peak value of the mains voltage of the AX5000 (P-0-0200) can be read.

Inputs

```
VAR_INPUT
    sNetId   : T_AmsNetId := '';
    bExecute : BOOL;
    tTimeout : TIME := DEFAULT_ADS_TIMEOUT;
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).
bExecute	BOOL	The function block is enabled via a positive edge at this input.
tTimeout	TIME	Maximum time allowed for the execution of the function block.

Inputs/outputs

```
VAR_IN_OUT
    Axis : NCTOPLC_AXIS_REF;(* reference to NC axis *)
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of the type NCTOPLC_AXIS_REF

Outputs

```
VAR_OUTPUT
    bBusy              : BOOL;
    bError             : BOOL;
    iAdsErrId          : UINT;
    iSercosErrId       : UINT;
    dwAttribute        : DWORD;
    fActualMainVoltage : LREAL;
END_VAR
```

Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
iAdsErrId	UINT	Returns the ADS error code of the last executed command when the bError output is set.
iSercosErrId	UINT	In the case of a set bError output returns the Sercos error of the last executed command.
dwAttribute	DWORD	Returns the attributes of the Sercos parameter.
fActualMainVoltage	LREAL	Returns the peak value of the current mains voltage of the AX5000 (e.g. 303.0 corresponds to 303.0 V).

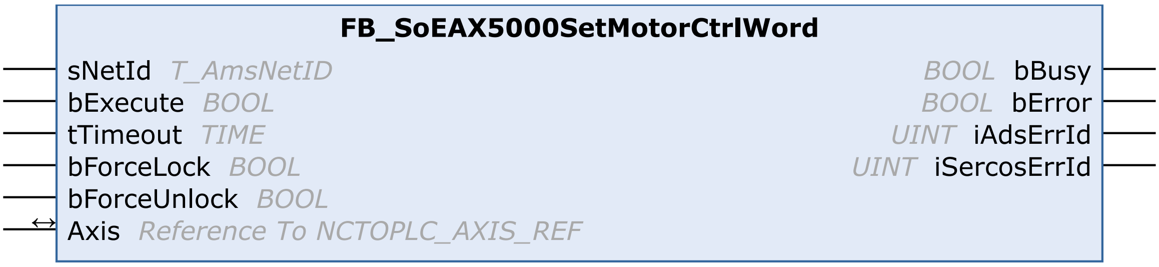
Sample

```
fbReadActMainVoltage : FB_SoEAX5000ReadActMainVoltage;
bReadActMainVoltage : BOOL;
fActualMainVoltage : REAL;
(* NcAxis *)
NcToPlc AT %I* : NCTOPLC_AXIS_REF;

IF bReadActMainVoltage THEN
   fbReadActMainVoltage(
     Axis := NcToPlc,
     bExecute := TRUE,
     tTimeout := DEFAULT_ADS_TIMEOUT,
     fActualMainVoltage=>fActualMainVoltage
   );
   IF NOT fbReadActMainVoltage.bBusy THEN
```

```
      fbReadActMainVoltage(Axis := NcToPlc, bExecute := FALSE);
      bReadActMainVoltage := FALSE;
   END_IF
END_IF
```

3.3.2 FB_SoEAX5000SetMotorCtrlWord



With the function block FB_SoEAX5000SetMotorCtrlWord the ForceLock bit (Bit 0) or the ForceUnlock bit can be set in the Motor Control Word (P-0-0096) to activate or release the brake. Normally the brake is automatically controlled via the Enable of the drive.

With the ForceLock bit, the brake can be activated independently from the Enable, with the ForceUnlock bit, the brake can be released independently from the Enable. In the case of simultaneously set ForceLock and ForceUnlock, ForceLock (Brake activated) has the higher priority.

Inputs

```
VAR_INPUT
    sNetId       : T_AmsNetId := '';
    bExecute     : BOOL;
    tTimeout     : TIME := DEFAULT_ADS_TIMEOUT;
    bForceLock   : BOOL;
    bForceUnlock : BOOL;
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	String, which contains the AMS Network ID of the PC (type: T_AmsNetId).
bExecute	BOOL	The function block is enabled via a positive edge at this input.
tTimeout	TIME	Maximum time allowed for the execution of the function block.
bForceLock	BOOL	Activates the brake independently of the enable.
bForceUnlock	BOOL	Releases the brake independently of the enable.

Inputs/outputs

```
VAR_IN_OUT
    Axis : NCTOPLC_AXIS_REF;(* reference to NC axis *)
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of the type NCTOPLC_AXIS_REF

Outputs

```
VAR_OUTPUT
    bBusy        : BOOL;
    bError       : BOOL;
    iAdsErrId    : UINT;
    iSercosErrId : UINT;
END_VAR
```

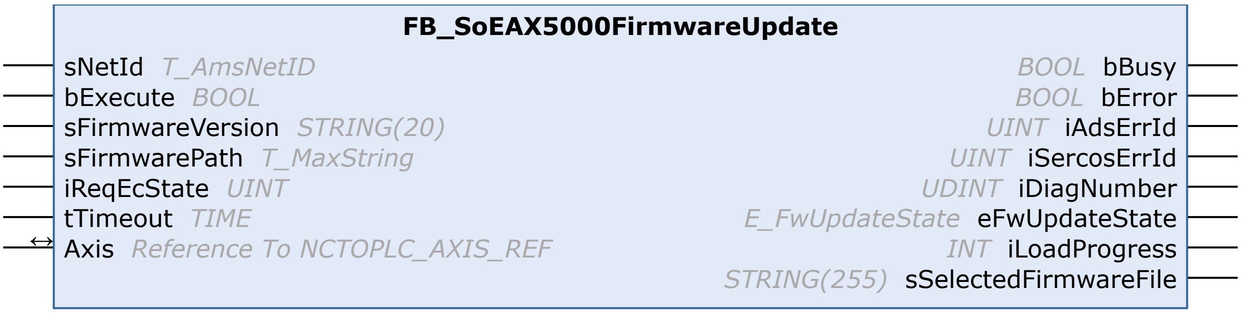
Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
iAdsErrId	UINT	Returns the ADS error code of the last executed command when the bError output is set.
iSercosErrId	UINT	In the case of a set bError output returns the Sercos error of the last executed command.

Sample

```
fbSetMotorCtrlWord : FB_SoEAX5000SetMotorCtrlWord;
bSetMotorCtrlWord : BOOL;
bForceLock : BOOL;
bForceUnlock : BOOL;
(* NcAxis *)
NcToPlc AT %I* : NCTOPLC_AXIS_REF;

IF bSetMotorCtrlWord THEN
   fbSetMotorCtrlWord(
     Axis := NcToPlc,
     bExecute := TRUE,
     tTimeout := DEFAULT_ADS_TIMEOUT,
     bForceLock := bForceLock,
     bForceUnlock:= bForceUnlock
   );
   IF NOT fbSetMotorCtrlWord.bBusy THEN
      fbSetMotorCtrlWord(Axis := NcToPlc, bExecute := FALSE);
      bSetMotorCtrlWord := FALSE;
   END_IF
END_IF
```

3.3.3 FB_SoEAX5000FirmwareUpdate


FB_SoEAX5000FirmwareUpdate
sNetId T_AmsNetID
bExecute BOOL
sFirmwareVersion STRING(20)
sFirmwarePath T_MaxString
iReqEcState UINT
tTimeout TIME
Axis Reference To NCTOPLC_AXIS_REF
BOOL bBusy
BOOL bError
UINT iAdsErrId
UINT iSercosErrId
UDINT iDiagNumber
E_FwUpdateState eFwUpdateState
INT iLoadProgress
STRING(255) sSelectedFirmwareFile

With the FB_SoEAX5000FirmwareUpdate function block the Firmware of the AX5000 can be checked and automatically changed to a given version (Revision and Build) or to the current Build of the configured revision.

For the update:

- the configured slave type is determined, e.g. AX5103-0000-0010
- the current slave is determined with the predefined slave address, e.g. AX5103-0000-0009
- the current slave firmware is determined, e.g. v1.05_b0009
- a comparison of the configuration and the found slave regarding number of channels, current, revision and firmware is made
- the name of the required firmware file is determined and a search for the file performed
- the firmware update is executed (if necessary)
- the current slave with the predefined slave address is determined again
- the slave is switched to the predefined EtherCAT state

A successful update ends with eFwUpdateState = eFwU_FwUpdateDone.

If the update is not required, this is signaled via eFwUpdateState = eFwU_NoFwUpdateRequired.

The firmware update takes place via the specified channel (A=0 or B=1) from stDriveRef. In the case of two-channel devices only one of the two channels can be used. The other channel signals
eFwUpdateState = eFwU_UpdateViaOtherChannelActive or
eFwUpdateState = eFwU_UpdateViaOtherChannel.

During the firmware update (eFwUpdateState = eFwU_FwUpdateInProgress), iLoadProgress signals the progress in percent.

NOTICE

Faulty update due to interruptions

Interruptions during the update may result in it not being executed or executed incorrectly. Afterwards, the terminal may no longer be usable without the appropriate firmware.

The rules during the update are:

- The PLC and TwinCAT must not be stopped.
- The EtherCAT connection must not be interrupted.
- The AX5000 must not be switched off.

Inputs

```
VAR_INPUT
    sNetId           : T_AmsNetId;
    bExecute         : BOOL;
    sFirmwareVersion : STRING(20); (* version string vx.yy_bnnnn, e.g. "v1.05_b0009" for v1.05 Build
 0009*)
    sFirmwarePath    : T_MaxString; (* drive:\path, e.g. "C:
\TwinCAT\Io\TcDriveManager\FirmwarePool" *)
    iReqEcState      : UINT := EC_DEVICE_STATE_OP;
    tTimeout         : TIME := DEFAULT_ADS_TIMEOUT;
END_VAR
```

Name	Type	Description
sNetId	T_AmsNetId	AMS-NetID of the controller (IPC).
bExecute	BOOL	The function block is activated by a positive edge at this input.
sFirmwareVersion	STRING(20)	Specifies the desired firmware version in the form of vx.yy_bnnnn, e.g. "v1.05_b0009" for Version v1.05 Build 0009. **Release-Builds:** • "v1.05_b**00**09" for a specific build, e.g. v1.05 Build 0009 • "v1.05_b**00**??" latest build of a specified version, e.g. v1.05 • "v1.??_b**00**??" latest build of a specified main version, e.g. v1 • "v?.??_b**00**??" latest build of the latest version • " " latest build of the latest version **Customer-specific Firmware-Builds:** • "v1.05_b**10**09" for a specific build, e.g. v1.05 Build 0009 • "v1.05_b**10**??" latest build of a specified version, e.g. v1.05 • "v1.??_b**10**??" latest build of a specified main version, e.g. v1 • "v?.??_b**10**??" latest build of the latest version ... • "v1.05_b**89**09" for a specific build, e.g. v1.05 Build 8909

Name	Type	Description
		• "v1.05_b**89**??" latest build of a specified version, e.g. v1.05 • "v1.??_b**89**??" latest build of a specified main version, e.g. v1 • "v?.??_b**89**??" latest build of the latest version **Debug-Builds:** • "v1.05_b**90**09" for a specific build, e.g. v1.05 Build 9009 • "v1.05_b**90**??" latest build of a specified version, e.g. v1.05 • "v1.??_b**90**??" latest build of a specified main version, e.g. v1 • "v?.??_b**90**??" latest build of the latest version
sFirmwarePath	T_MaxString	Specifies the path for the firmware pool in which the firmware files are located, e.g. *C:\TwinCAT\Io\TcDriveManager\FirmwarePool*.
iReqEcState	UINT	Desired EtherCAT state after the update (only if an update is actually being executed). The states are defined in PLC Lib Tc2_EtherCAT as global constants.
tTimeout	TIME	Since the firmware update for large EtherCAT networks can take longer, only the timeout for individual internal ADS instances is specified here.

Inputs/outputs

```
VAR_IN_OUT
    Axis : NCTOPLC_AXIS_REF;(* reference to NC axis *)
END_VAR
```

Name	Type	Description
Axis	NCTOPLC_AXIS_REF	Axis data structure of the type NCTOPLC_AXIS_REF

Outputs

```
VAR_OUTPUT
bBusy                   : BOOL;
bError                  : BOOL;
iAdsErrId               : UINT;
iSercosErrId            : UINT;
iDiagNumber             : UDINT;
eFwUpdateState          : E_FwUpdateState;
iLoadProgress           : INT;
sSelectedFirmwareFile : STRING(MAX_STRING_LENGTH); (* found firmware file, e.g. "AX5yxx_xxxx_-0010_v
1_05_b0009.efw" *)
END_VAR
```

Name	Type	Description
bBusy	BOOL	This output is set when the function block is activated, and remains set until a feedback is received.
bError	BOOL	This output is set after the bBusy output has been reset when an error occurs in the transmission of the command.
iAdsErrId	UINT	Returns the ADS error code of the last executed command when the bError output is set.
iSercosErrId	UINT	In the case of a set bError output returns the Sercos error of the last executed command.
iDiagNumber	UDINT	In the case of a set bError output returns the drive error of the last executed firmware update.

Name	Type	Description
eFwUpdateState	E_FwUpdateState	Returns the status of the firmware update.
iLoadProgress	INT	Returns the progress of the actual firmware update as a percentage.
sSelectedFirmwareFile	STRING(MAX_STRING_LENGTH)	Displays the name of the firmware file being searched for.

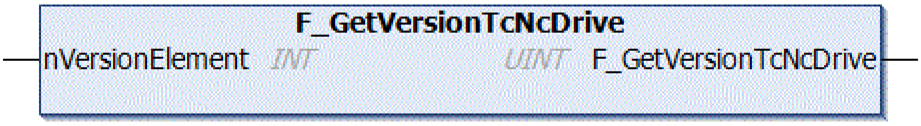
Sample

```
VAR CONSTANT
 iNumOfDrives : INT := 2;
END_VAR

VAR
 fbFirmwareUpdate : ARRAY [1..iNumOfDrives] OF FB_SoEAX5000FirmwareUpdate;
 NcToPlc AT %I* : ARRAY [1..iNumOfDrives] OF NCTOPLC_AXIS_REF;
 sFirmwareVersion : ARRAY [1..iNumOfDrives] OF STRING(20)(* := 2('v1.04_b0002')*);
 eFwUpdateState : ARRAY [1..iNumOfDrives] OF E_FwUpdateState;
 sSelectedFirmwareFile: ARRAY [1..iNumOfDrives] OF STRING(MAX_STRING_LENGTH);
 iUpdateState : INT;
 bExecute : BOOL;
 sNetIdIPC : T_AmsNetId := '';
 sFirmwarePath : T_MaxString := 'C:\TwinCAT\Io\TcDriveManager\FirmwarePool';
 I : INT;
 bAnyBusy : BOOL;
 bAnyError : BOOL;
END_VAR

CASE iUpdateState OF
0:
  IF bExecute THEN
     iUpdateState := 1;
  END_IF
1:
  FOR I := 1 TO iNumOfDrives DO
    fbFirmwareUpdate[I](
     Axis := NcToPlc[I],
     bExecute := TRUE,
     tTimeout := T#15s,
     sFirmwareVersion := sFirmwareVersion[I],
     sFirmwarePath := sFirmwarePath,
     sNetId := sNetIdIPC,
     iReqEcState := EC_DEVICE_STATE_OP,
     eFwUpdateState => eFwUpdateState[I],
    );
  END_FOR
  iUpdateState := 2;
2:
  bAnyBusy := FALSE;
  bAnyError:= FALSE;
  FOR I := 1 TO iNumOfDrives DO
    fbFirmwareUpdate[I](
     Axis := NcToPlc[I],
     eFwUpdateState => eFwUpdateState[I],
     sSelectedFirmwareFile => sSelectedFirmwareFile[I],
    );
    IF NOT fbFirmwareUpdate[I].bBusy THEN
      fbFirmwareUpdate[I](bExecute := FALSE, Axis := NcToPlc[I]);
      IF fbFirmwareUpdate[I].bError THEN
         bAnyError := TRUE;
      END_IF
      ELSE
      bAnyBusy := TRUE;
    END_IF
  END_FOR
  IF NOT bAnyBusy THEN
    bExecute := FALSE;
    IF NOT bAnyError THEN
      iUpdateState := 0; (* OK *)
    ELSE
      iUpdateState := 0; (* Error *)
    END_IF
  END_IF
END_CASE
```

3.4 F_GetVersionTcNcDrive



This function can be used to read PLC library version information.

FUNCTION F_GetVersionTcNcDrive: UINT

```
VAR_INPUT
    nVersionElement : INT;
END_VAR
```

nVersionElement: Version element to be read. Possible parameters:

- 1 : major number;
- 2 : minor number;
- 3 : revision number;

4 Support and Service

Beckhoff and their partners around the world offer comprehensive support and service, making available fast and competent assistance with all questions related to Beckhoff products and system solutions.

Download finder

Our download finder contains all the files that we offer you for downloading. You will find application reports, technical documentation, technical drawings, configuration files and much more.

The downloads are available in various formats.

Beckhoff's branch offices and representatives

Please contact your Beckhoff branch office or representative for local support and service on Beckhoff products!

The addresses of Beckhoff's branch offices and representatives round the world can be found on our internet page: www.beckhoff.com

You will also find further documentation for Beckhoff components there.

Beckhoff Support

Support offers you comprehensive technical assistance, helping you not only with the application of individual Beckhoff products, but also with other, wide-ranging services:

- support
- design, programming and commissioning of complex automation systems
- and extensive training program for Beckhoff system components

Hotline:	+49 5246 963-157
e-mail:	support@beckhoff.com

Beckhoff Service

The Beckhoff Service Center supports you in all matters of after-sales service:

- on-site service
- repair service
- spare parts service
- hotline service

Hotline:	+49 5246 963-460
e-mail:	service@beckhoff.com

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

Phone:	+49 5246 963-0
e-mail:	info@beckhoff.com
web:	www.beckhoff.com

More Information:
www.beckhoff.com/te1000

Beckhoff Automation GmbH & Co. KG
Hülshorstweg 20
33415 Verl
Germany
Phone: +49 5246 9630
info@beckhoff.com
www.beckhoff.com