

Handbuch | DE

# TS6271

TwinCAT 2 | PROFINET RT Controller





# Inhaltsverzeichnis

<b>1</b>	<b>Vorwort.....</b>	<b>5</b>
1.1	Versionsidentifikation EtherCAT Geräte .....	5
1.2	Hinweise zur Dokumentation .....	6
1.3	Hinweise zur Informationssicherheit .....	8
1.4	Zu Ihrer Sicherheit.....	8
<b>2</b>	<b>Produktübersicht .....</b>	<b>10</b>
2.1	Technische Daten .....	10
2.2	Freischaltung des PROFINET-Controller Supplement.....	11
2.3	EL6632 PROFINET-IRT-Controller .....	14
2.4	Technische Daten - EL6631 PROFINET-RT-Controller.....	15
2.5	Technische Daten - EL6632 PROFINET-IRT-Controller.....	16
<b>3</b>	<b>EL6631/EL6632 .....</b>	<b>18</b>
3.1	Montage und Verdrahtung .....	18
3.1.1	Empfohlene Tragschienen .....	18
3.1.2	Montage und Demontage.....	18
3.1.3	Montage und Demontage.....	20
3.1.4	Einbaulagen .....	22
3.1.5	ATEX - Besondere Bedingungen .....	24
3.2	Diagnose .....	25
3.2.1	EL6631 - LEDs.....	25
<b>4</b>	<b>PN Controller Protokoll.....</b>	<b>27</b>
4.1	Einbindung des TwinCAT PROFINET Controller Protokolls über eine RealTimeEthernet Schnittstelle .....	27
4.2	Einbindung des TwinCAT PROFINET Controller Protokolls über eine EL663x Schnittstelle .....	27
4.3	Einstellungen / Diagnose .....	28
4.3.1	PROFINET .....	28
4.3.2	Taskkonfiguration.....	33
4.3.3	PROFINET Controller spezifische Einstellungen .....	34
4.3.4	Analyse der Box States .....	35
4.3.5	Diagnose Historie am Controller Protokoll .....	38
4.3.6	Zyklische Daten.....	39
4.3.7	Azyklische Daten.....	39
<b>5</b>	<b>Geräte am Protokoll .....</b>	<b>41</b>
5.1	PROFINET Devices anfügen .....	41
5.2	Vergleichen von Soll- und Istbestückung .....	42
5.3	Einstellungen.....	47
5.3.1	Projektierung des PROFINET Device .....	47
5.3.2	BK9xx3.....	49
5.3.3	EL663x .....	50
5.3.4	IRT Controller.....	50
5.3.5	Shared Device.....	51
5.4	Module .....	51
5.4.1	Diagnose auf Modul Ebene.....	51

5.4.2	Submodule .....	52
<b>6</b>	<b>TwinCAT Bibliothek und Programmierung.....</b>	<b>55</b>
6.1	Funktionsbausteine .....	55
6.1.1	FB_PN_IM0_READ.....	55
6.1.2	FB_PN_IM1_READ.....	56
6.1.3	FB_PN_IM1_WRITE .....	57
6.1.4	FB_PN_IM2_READ.....	58
6.1.5	FB_PN_IM2_WRITE .....	59
6.1.6	FB_PN_IM3_READ.....	60
6.1.7	FB_PN_IM3_WRITE .....	61
6.1.8	FB_PN_IM4_Read.....	62
6.1.9	FB_PN_IM4_WRITE .....	63
6.1.10	FB_PN_GET_PORT_STATISTIC.....	64
6.1.11	FB_PN_READ_PORT_DIAG.....	65
6.1.12	FB_PN_ALARM_DIAG .....	65
6.2	Datenstrukturen.....	67
6.2.1	str_SW_Rec.....	67
6.2.2	str_IM_0xAFF1.....	68
6.2.3	str_IM_0xAFF2.....	68
6.2.4	str_IM_0xAFF3.....	68
6.2.5	str_IM_0xAFF4.....	68
6.2.6	str_GetPortStatistic .....	68
6.2.7	str_PortDiag .....	69
6.2.8	ST_PN_DiagMessage.....	69
6.2.9	ST_PN_Diag .....	69
6.2.10	ST_PN_AlarmDiagData .....	70
6.2.11	E_PN_ALARM_TYP .....	70
<b>7</b>	<b>Anhang.....</b>	<b>72</b>
7.1	UL Hinweise .....	72
7.2	Firmware Kompatibilität.....	73
7.3	Firmware EL/ES/EM/EPxxxx.....	73
7.4	ATEX-Dokumentation .....	82
7.5	Support und Service.....	83

# 1 Vorwort

## 1.1 Versionsidentifikation EtherCAT Geräte

### Bezeichnung

Ein Beckhoff EtherCAT-Gerät verfügt über eine 14stellige Bezeichnung, die sich zusammensetzt aus

- Familienschlüssel
- Typ
- Version
- Revision

Beispiel	Familie	Typ	Version	Revision
EL3314-0000-0016	EL-Klemme (12 mm, nicht steckbare Anschlussebene)	3314 (4 kanalige Thermoelementklemme)	0000 (Grundtyp)	0016
CU2008-0000-0000	CU-Gerät	2008 (8 Port FastEthernet Switch)	0000 (Grundtyp)	0000
ES3602-0010-0017	ES-Klemme (12 mm, steckbare Anschlussebene)	3602 (2 kanalige Spannungsmessung)	0010 (Hochpräzise Version)	0017

- Die **Bestellbezeichnung** setzt sich zusammen aus
  - Familienschlüssel
  - Typ
  - Version
- Die **Revision** gibt den technischen Fortschritt wieder und wird von Beckhoff verwaltet. Prinzipiell kann ein Gerät mit höherer Revision ein Gerät mit niedrigerer Revision ersetzen, wenn nicht anders angegeben. Jeder Revision zugehörig ist üblicherweise eine XML-Beschreibung (ESI, EtherCAT Slave Information), die zum Download auf der Beckhoff Webseite bereitsteht.
- Typ, Version und Revision werden als dezimale Zahlen gelesen.

### Seriennummer

Beckhoff EtherCAT Geräte verfügen in der Regel über eine 8-stellige Seriennummer, die auf dem Gerät aufgedruckt oder auf einem Aufkleber angebracht ist. Diese Seriennummer gibt den Bauzustand im Auslieferungszustand an.

### Syntax der Seriennummer

Aufbau der Seriennummer: KK YY FF HH

- KK - Produktionswoche (Kalenderwoche)
- YY - Produktionsjahr
- FF - Firmware-Stand
- HH - Hardware-Stand

Beispiel mit Ser. Nr.: 12 06 3A 02:

- 12 - Produktionswoche 12
- 06 - Produktionsjahr 2006
- 3A - Firmware-Stand 3A
- 02 - Hardware-Stand 02

## Beispiele für Seriennummerangaben



Abb. 1: EL6688 EL-Klemme



Abb. 2: CU2016 Switch



Abb. 3: EK1100 EtherCAT Koppler

## 1.2 Hinweise zur Dokumentation

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs- und Automatisierungstechnik, das mit den geltenden nationalen Normen vertraut ist.

Zur Installation und Inbetriebnahme der Komponenten ist die Beachtung der Dokumentation und der nachfolgenden Hinweise und Erklärungen unbedingt notwendig.

Das Fachpersonal ist verpflichtet, stets die aktuell gültige Dokumentation zu verwenden.

Das Fachpersonal hat sicherzustellen, dass die Anwendung bzw. der Einsatz der beschriebenen Produkte alle Sicherheitsanforderungen, einschließlich sämtlicher anwendbaren Gesetze, Vorschriften, Bestimmungen und Normen erfüllt.

### Disclaimer

Diese Dokumentation wurde sorgfältig erstellt. Die beschriebenen Produkte werden jedoch ständig weiterentwickelt.

Wir behalten uns das Recht vor, die Dokumentation jederzeit und ohne Ankündigung zu überarbeiten und zu ändern.

Aus den Angaben, Abbildungen und Beschreibungen in dieser Dokumentation können keine Ansprüche auf Änderung bereits gelieferter Produkte geltend gemacht werden.

**Marken**

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® und XPlanar® sind eingetragene und lizenzierte Marken der Beckhoff Automation GmbH.

Die Verwendung anderer in dieser Dokumentation enthaltenen Marken oder Kennzeichen durch Dritte kann zu einer Verletzung von Rechten der Inhaber der entsprechenden Bezeichnungen führen.

**Patente**

Die EtherCAT-Technologie ist patentrechtlich geschützt, insbesondere durch folgende Anmeldungen und Patente:

EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702

mit den entsprechenden Anmeldungen und Eintragungen in verschiedenen anderen Ländern.

**EtherCAT®** 

EtherCAT® ist eine eingetragene Marke und patentierte Technologie lizenziert durch die Beckhoff Automation GmbH, Deutschland

**Copyright**

© Beckhoff Automation GmbH & Co. KG, Deutschland.

Weitergabe sowie Vervielfältigung dieses Dokuments, Verwertung und Mitteilung seines Inhalts sind verboten, soweit nicht ausdrücklich gestattet.

Zuwerhandlungen verpflichten zu Schadenersatz. Alle Rechte für den Fall der Patent-, Gebrauchsmuster- oder Geschmacksmustereintragung vorbehalten.

## 1.3 Hinweise zur Informationssicherheit

Die Produkte der Beckhoff Automation GmbH & Co. KG (Beckhoff) sind, sofern sie online zu erreichen sind, mit Security-Funktionen ausgestattet, die den sicheren Betrieb von Anlagen, Systemen, Maschinen und Netzwerken unterstützen. Trotz der Security-Funktionen sind die Erstellung, Implementierung und ständige Aktualisierung eines ganzheitlichen Security-Konzepts für den Betrieb notwendig, um die jeweilige Anlage, das System, die Maschine und die Netzwerke gegen Cyber-Bedrohungen zu schützen. Die von Beckhoff verkauften Produkte bilden dabei nur einen Teil des gesamtheitlichen Security-Konzepts. Der Kunde ist dafür verantwortlich, dass unbefugte Zugriffe durch Dritte auf seine Anlagen, Systeme, Maschinen und Netzwerke verhindert werden. Letztere sollten nur mit dem Unternehmensnetzwerk oder dem Internet verbunden werden, wenn entsprechende Schutzmaßnahmen eingerichtet wurden.

Zusätzlich sollten die Empfehlungen von Beckhoff zu entsprechenden Schutzmaßnahmen beachtet werden. Weiterführende Informationen über Informationssicherheit und Industrial Security finden Sie in unserem <https://www.beckhoff.de/secguide>.

Die Produkte und Lösungen von Beckhoff werden ständig weiterentwickelt. Dies betrifft auch die Security-Funktionen. Aufgrund der stetigen Weiterentwicklung empfiehlt Beckhoff ausdrücklich, die Produkte ständig auf dem aktuellen Stand zu halten und nach Bereitstellung von Updates diese auf die Produkte aufzuspielen. Die Verwendung veralteter oder nicht mehr unterstützter Produktversionen kann das Risiko von Cyber-Bedrohungen erhöhen.

Um stets über Hinweise zur Informationssicherheit zu Produkten von Beckhoff informiert zu sein, abonnieren Sie den RSS Feed unter <https://www.beckhoff.de/secinfo>.

## 1.4 Zu Ihrer Sicherheit

### Sicherheitsbestimmungen

Lesen Sie die folgenden Erklärungen zu Ihrer Sicherheit. Beachten und befolgen Sie stets produktspezifische Sicherheitshinweise, die Sie gegebenenfalls an den entsprechenden Stellen in diesem Dokument vorfinden.

### Haftungsausschluss

Die gesamten Komponenten werden je nach Anwendungsbestimmungen in bestimmten Hard- und Software-Konfigurationen ausgeliefert. Änderungen der Hard- oder Software-Konfiguration, die über die dokumentierten Möglichkeiten hinausgehen, sind unzulässig und bewirken den Haftungsausschluss der Beckhoff Automation GmbH & Co. KG.

### Qualifikation des Personals

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs-, Automatisierungs- und Antriebstechnik, das mit den geltenden Normen vertraut ist.

### Signalwörter

Im Folgenden werden die Signalwörter eingeordnet, die in der Dokumentation verwendet werden. Um Personen- und Sachschäden zu vermeiden, lesen und befolgen Sie die Sicherheits- und Warnhinweise.

### Warnungen vor Personenschäden

#### **GEFAHR**

Es besteht eine Gefährdung mit hohem Risikograd, die den Tod oder eine schwere Verletzung zur Folge hat.

#### **WARNUNG**

Es besteht eine Gefährdung mit mittlerem Risikograd, die den Tod oder eine schwere Verletzung zur Folge haben kann.

**⚠ VORSICHT**

Es besteht eine Gefährdung mit geringem Risikograd, die eine mittelschwere oder leichte Verletzung zur Folge haben kann.

**Warnung vor Umwelt- oder Sachschäden****HINWEIS**

Es besteht eine mögliche Schädigung für Umwelt, Geräte oder Daten.

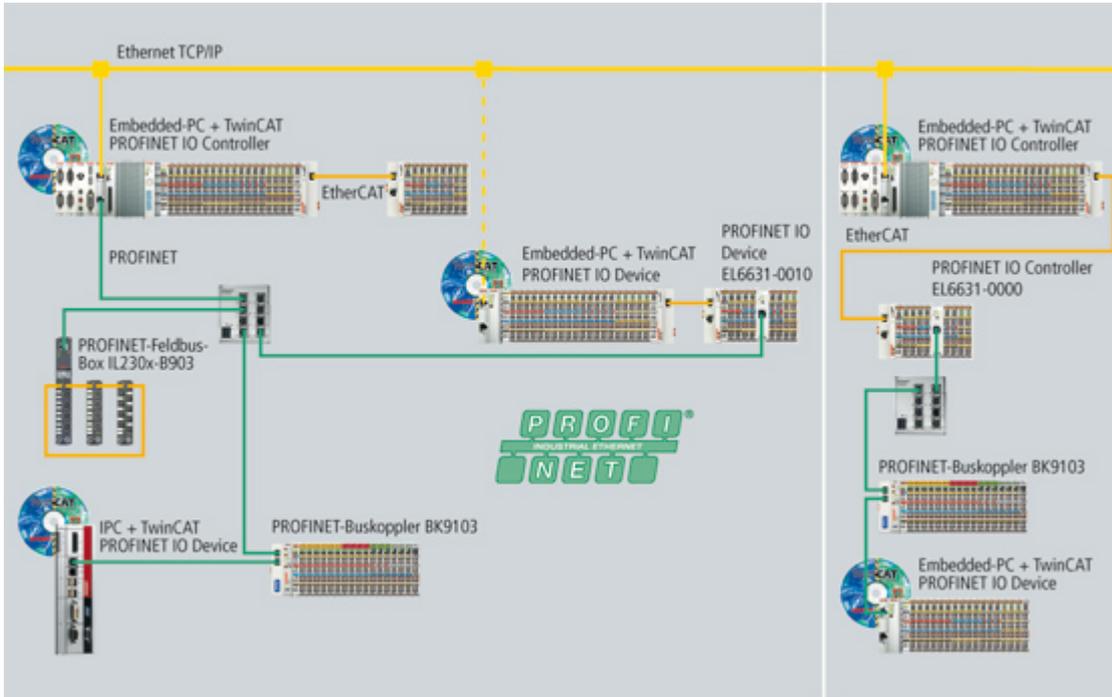
**Information zum Umgang mit dem Produkt**

Diese Information beinhaltet z. B.:  
Handlungsempfehlungen, Hilfestellungen oder weiterführende Informationen zum Produkt.

## 2 Produktübersicht

Der PROFINET IO Controller (Master) ist ein TwinCAT-Supplement, das aus jeder PC-basierten Steuerung von Beckhoff einen PROFINET-IO-Controller macht. Eine Standard-Ethernet-Schnittstelle wird durch Installation des Supplements zu einem PROFINET-Master. Auf PCs und Embedded-PCs kann dieses Supplement genutzt werden.

In Verbindung mit der PROFINET-Klemme EL663x für das EtherCAT-I/O-System kann PROFINET einfach mit EtherCAT gekoppelt werden, wobei hier die Klemme EtherCat-seitig einen Slave darstellt, zum PROFINET hin jedoch als Controller agiert. Damit kann jedes EtherCAT-Netzwerk auch mit PROFINET-IO Teilnehmern Daten austauschen.



Es gibt zwei Arten von PROFINET: PROFINET I/O und PROFINET CBA. TwinCAT unterstützt PROFINET I/O.

Innerhalb von PROFINET I/O sind wiederum aktuell 4 Arten der zyklischen Kommunikation definiert, RTClass1..3 und RTOverUDP. Der TwinCAT PROFINET I/O Controller und die EL6631 unterstützen aktuell RTClass1, die EL6632 kann über RTClass1 und RTClass3 kommunizieren. Es werden alle für RTClass1 definierten Zykluszeiten unterstützt, von 1 ms in 2er Potenz Schritten (1, 2, 4, 8, ...ms).

Die kleinste derzeit unterstützte Zykluszeit der EL6632 ist 500µs (für RTClass3).

### 2.1 Technische Daten

Technische Daten	PROFINET via RT Ethernet	PROFINET via EtherCAT (EL6632)	PROFINET via EtherCAT (EL6631)
Ethernet-Hardware	Real-Time-Ethernet-Hardware	PROFINET-Klemme EL6632	PROFINET-Klemme EL6631
Betriebssystem	Windows XP, XP Embedded (CE in Vorbereitung)		
Software	TwinCAT I/O, PLC, NC, NC I, CNC		
Zielsysteme	PC (x86), Windows-CE-Geräte	PC (x86), Windows-CE-Geräte mit EtherCAT-Interface	PC (x86), Windows-CE-Geräte mit EtherCAT-Interface
Zykluszeit	min. 1 ms	min. 500us	min. 1ms
Anzahl der möglichen IO-Devices	begrenzt durch CPU-Leistung und Speicher	für IRT bis zu 5 Teilnehmer, für RT bis zu 15 Teilnehmer	bis zu 15 Teilnehmer

## 2.2 Freischaltung des PROFINET-Controller Supplement

### Voraussetzung:

TwinCAT 2.11 Build 1545

Hardware: Ethernet Karte mit Intel Chipsatz

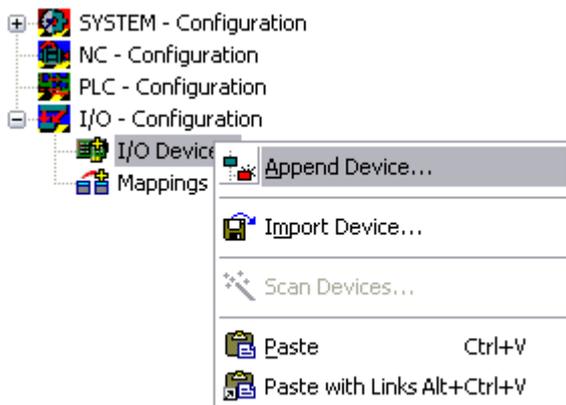
Für die Verwendung des PROFINET-Controller Supplement ist eine Ethernet Karte mit Intel Chipsatz notwendig. Installieren Sie den RT-Ethernet Treiber für diese Karte ([TwinCAT Ethernet Treiber - Installation](#)). Nach Beendigung der Installation des Treibers können Sie im System Manager den PROFINET Controller anlegen und mit der Hardware (Ethernet-Karte) verbinden.

Im Config Mode (TwinCAT Icon blau) kann der PROFINET Controller gescannt werden und im Freerun Modus Werte Schreiben. Um TwinCAT im 'Run Mode zu betreiben, muss der passende Schlüssel eingegeben werden.

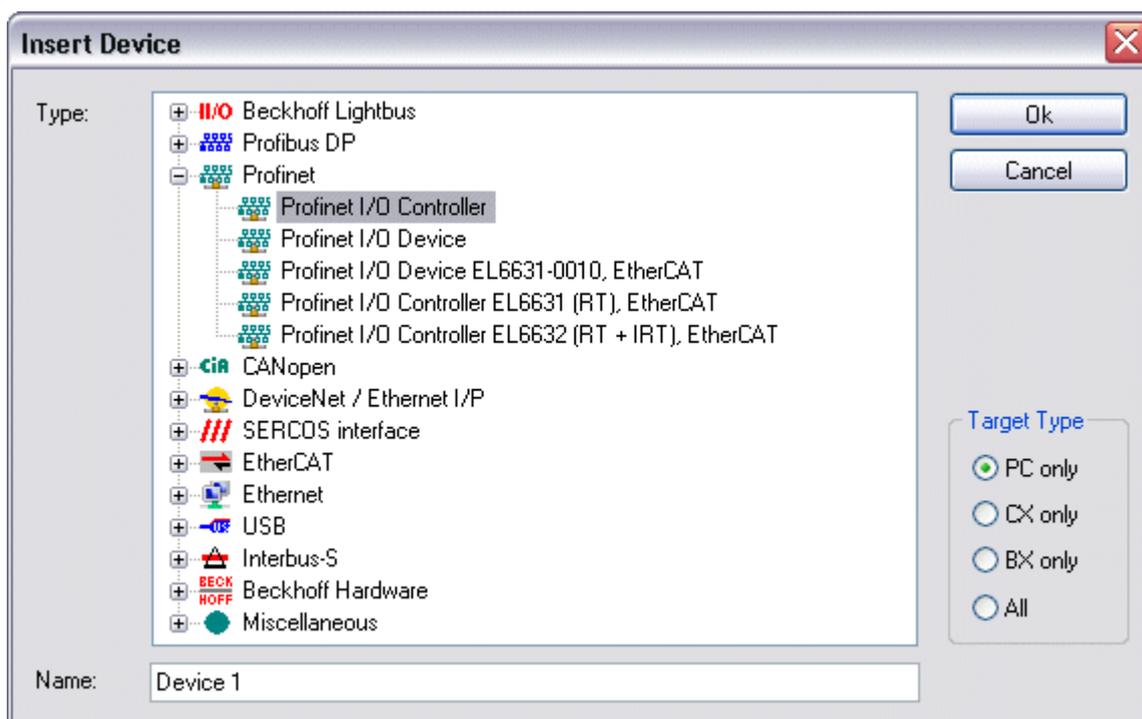
Um den Registrierungsschlüssel zu erhalten, senden Sie mit dem Auftrag die MAC Adresse Ihrer verwendeten Ethernet Karte zu. Sie erhalten daraufhin den Freischalt-Code.

### MAC Adresse

Die MAC Adresse können Sie über den System Manager auslesen. Fügen Sie dafür das PROFINET Controller Device an.



Wählen Sie unter PROFINET den PROFINET I/O Controller aus.



Wechseln Sie zu Adapter und drücken Sie die Search-Taste.

The screenshot shows a configuration window with the following elements:

- Tabbed interface: General, **Adapter**, PROFINET, Sync Task, Settings, Box States
- Radio buttons:  Own Adapter Instance,  Shared Adapter Instance
- Fields:
  - Description: [Empty]
  - Device Name: [Empty]
  - MAC Address: 00 00 00 00 00 00
  - IP Address: 0.0.0.0 (0.0.0.0)
- Buttons: Search..., Compatible Devices...
- Checkboxes:
  - Promiscuous Mode (use with Netmon/Wireshark only)
  - Virtual Device Names
- Shared Adapter Instance: Adapter: [Dropdown menu]
- Freerun Cycle (ms): 4

Wenn der Echtzeittreiber richtig installiert wurde, werden Ihnen die entsprechenden Karten angezeigt. Stehen keine Karten zur Auswahl, wurde der Treiber fehlerhaft installiert. Wählen Sie die Karte aus, auf an der Ihrer PROFINET-Geräte angeschlossen sind.

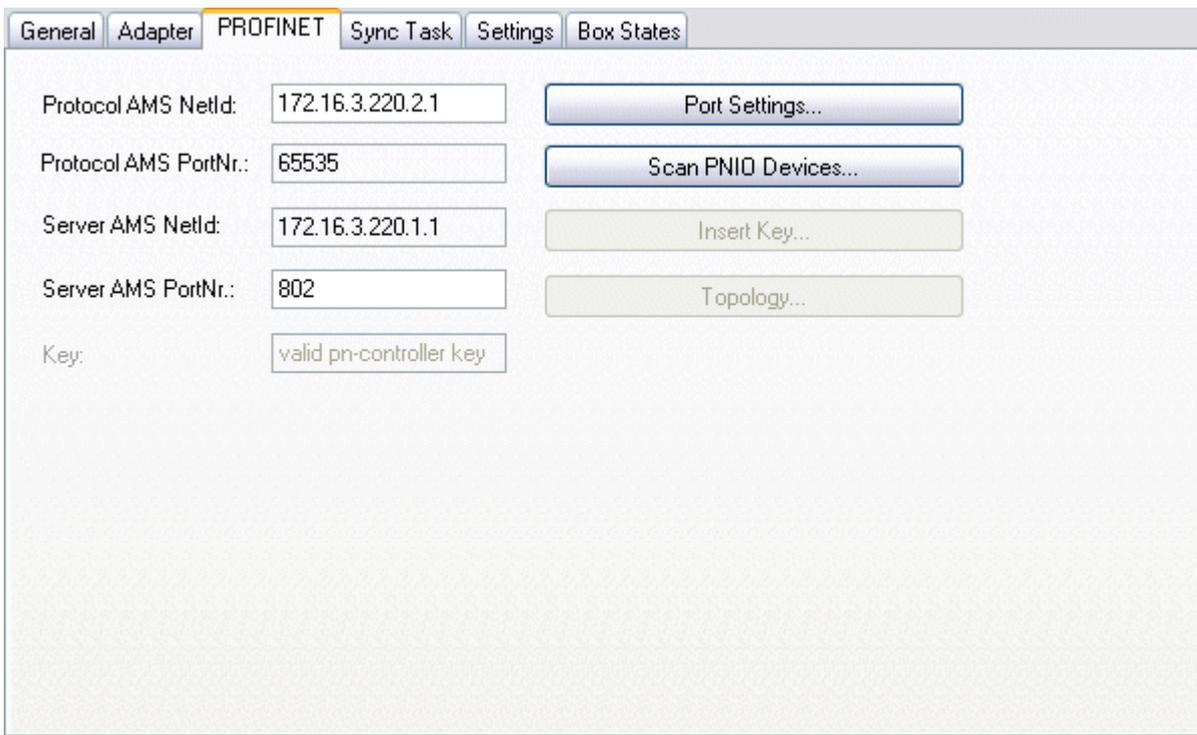
The 'Device Found At' dialog box contains the following information:

- Title: Device Found At
- List of devices:
  - (none)
  - TwinCAT RT 2 (Intel(R) PRO/100 S Desktop Adapter #2 - Paketplaner-
  - TwinCAT RT (TwinCAT-Intel PCI Ethernet Adapter - Paketplaner-Minipo
- Buttons: OK, Cancel, Help
- Radio buttons:
  - Unused
  - All

Nun wird Ihnen unter Adapter -> MAC Address die MAC Adresse der Karte angezeigt. Diese wird nun für den Key benötigt.

Wechseln Sie den Karteireiter PROFINET und drücken Sie "Insert Key...". Tragen Sie nun den Key ein.

Wird der Key akzeptiert steht im Feld Key: *valid pn-controller key*.



## 2.3 EL6632 PROFINET-IRT-Controller



Die PROFINET-IO-Controller-(Master)-Klemme EL6631 unterstützt die komplette Real-Time-Funktionalität (RT) sowie umfangreiche Diagnosemöglichkeiten. Es werden alle Dienste nach Conformance Class B unterstützt.

An der EL6631 können bis zu 15 PROFINET-IO-Devices projektiert werden.



Die PROFINET-IRT-Controller-Klemme EL6632 bringt die komplette RT (Real-Time)- oder IRT (Isochronous Real-Time)-Funktionalität sowie umfangreiche Diagnosemöglichkeiten mit. Es werden alle Dienste nach Konformitätsklasse C unterstützt.

An der EL6632 können, abhängig von der Zykluszeit, bis zu fünf PROFINET-IRT- oder bis zu 15 PROFINET-RT-Devices in Linientopologie betrieben werden.

● **Supplement**

**i** Das TwinCAT Supplement wird für die EL6631 und EL6632 nicht benötigt.

● **TwinCAT Version**

**i** Die freigegebene TwinCAT Version ist TwinCAT 2.11 R3. Es ist darauf zu achten, dass auch das Zielsystem der TwinCAT-Version entspricht. Ältere TwinCAT-Versionen können nicht verwendet werden!

## 2.4 Technische Daten - EL6631 PROFINET-RT-Controller

Technische Daten	EL6631
Technik	PROFINET IO

Technische Daten	EL6631
Anzahl Ports/Kanäle	2
Ethernet-Interface	100BASE-TX-Ethernet mit 2 x RJ 45
Feldbus	PROFINET RT Controller
Leitungslänge	bis 100 m Twisted-Pair
Hardwarediagnose	Status-LEDs
Spannungsversorgung	über den E-Bus
Potenzialtrennung	500 V (E-Bus/Ethernet)
Max. Anzahl RT-fähiger Teilnehmer	15 (abhängig von der Zykluszeit und Anzahl der Daten)
Minimaler RT-Zyklus	1 ms
Konformitätsklasse	B
Protokoll	RT
Treiber	TwinCAT 2.11 R3
Maximale Anzahl an Prozessdaten	1 kByte Eingangsdaten and 1 Byte Ausgangsdaten
Konfiguration	über den EtherCAT-Master
Stromaufnahme E-Bus	400 mA typ.
Besondere Eigenschaften	LLDP, SNMP, Konformitätsklasse B, max. 15 RT-Teilnehmer, min. 1-ms-RT-Zyklus
Montage (Gehäuse mit <a href="#">Zughebelentriegelung</a> <a href="#">[► 18]</a> / Gehäuse mit <a href="#">Frontentriegelung</a> <a href="#">[► 20]</a> )	auf 35 mm Tragschiene nach EN 50022
Gewicht	ca. 75 g
Betriebs-/Lagertemperatur	0...+55 °C/-25...+85 °C
Relative Feuchte	95 % ohne Betauung
Vibrations-/Schockfestigkeit	gemäß EN 60068-2-6/EN 60068-2-27/29
EMV-Festigkeit/-Aussendung	gemäß EN 61000-6-2/EN 61000-6-4
Einbaulage	Standard Einbaulage (bei anderen Einbaulagen gelten 45 °C Betriebstemperatur), siehe auch <a href="#">Hinweis</a> <a href="#">[► 22]</a>
Schutzart	IP 20/
Zulassung	CE <a href="#">Ex</a> <a href="#">[► 24]</a> UL

## 2.5 Technische Daten - EL6632 PROFINET-IRT-Controller

Technische Daten	EL6632
Technik	PROFINET IO
Anzahl Ports/Kanäle	2
Ethernet-Interface	100BASE-TX-Ethernet mit 2 x RJ 45
Feldbus	PROFINET RT / IRT Controller
Leitungslänge	bis 100 m Twisted-Pair
Hardwarediagnose	Status-LEDs
Spannungsversorgung	über den E-Bus
Potenzialtrennung	500 V (E-Bus/Ethernet)
Max. Anzahl IRT-fähiger Teilnehmer	5 (abhängig von der Zykluszeit und Anzahl der Daten)

Technische Daten	EL6632
Max. Anzahl RT-fähiger Teilnehmer	15 (abhängig von der Zykluszeit und Anzahl der Daten)
Minimaler IRT-Zyklus	500 µs
Minimaler RT-Zyklus	1 ms
Konformitätsklasse	C
Protokoll	RT oder IRT
Treiber	TwinCAT 2.11 R3
Maximale Anzahl an Prozessdaten	1 kByte Eingangsdaten and 1 Byte Ausgangsdaten
Konfiguration	über den EtherCAT-Master
Stromaufnahme E-Bus	400 mA typ.
Besondere Eigenschaften	Konformitätsklasse C, max. 5 IRT-Teilnehmer in Reihe, max. 15 RT-Teilnehmer, min. 500-µs-IRT-Zyklus, min. 1-ms-RT-Zyklus
Abmessungen (B x H x T)	ca. 26 mm x 100 mm x 52 mm (Breite angereicht: 23 mm)
Montage (Gehäuse mit <a href="#">Zughebelentriegelung</a> [► 18] / Gehäuse mit <a href="#">Frontentriegelung</a> [► 20])	auf 35 mm Tragschiene nach EN 50022
Gewicht	ca. 75 g
Betriebs-/Lagertemperatur	0...+55 °C/-25...+85 °C
Relative Feuchte	95 % ohne Betauung
Vibrations-/Schockfestigkeit	gemäß EN 60068-2-6/EN 60068-2-27/29
Einbaulage	Standard Einbaulage (bei anderen Einbaulagen gelten 45 °C Betriebstemperatur), siehe auch <a href="#">Hinweis</a> [► 22]
Schutzart	IP 20/
Zulassung	CE <a href="#">Ex</a> [► 24] UL

## 3 EL6631/EL6632

### 3.1 Montage und Verdrahtung

#### 3.1.1 Empfohlene Tragschienen

Klemmenmodule und EtherCAT-Module der Serien KMxxxx, EMxxxx, sowie Klemmen der Serien EL66xx und EL67xx können Sie auf folgende Tragschienen aufrasten:

- Hutschiene TH 35-7.5 mit 1 mm Materialstärke (nach EN 60715)
- Hutschiene TH 35-15 mit 1,5 mm Materialstärke

#### **i** Materialstärke der Hutschiene beachten

Klemmenmodule und EtherCAT-Module der Serien KMxxxx, EMxxxx, sowie Klemmen der Serien EL66xx und EL67xx passen nicht auf die Hutschiene TH 35-15 mit 2,2 bis 2,5 mm Materialstärke (nach EN 60715)!

#### 3.1.2 Montage und Demontage

Die Klemmenmodule werden mit Hilfe einer 35 mm Tragschiene (z.B. Hutschiene TH 35-7.5 nach EN 60715) auf der Montagefläche befestigt.

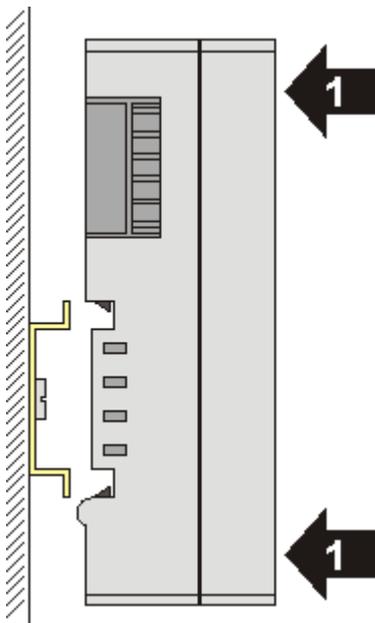
#### **⚠** WARNUNG

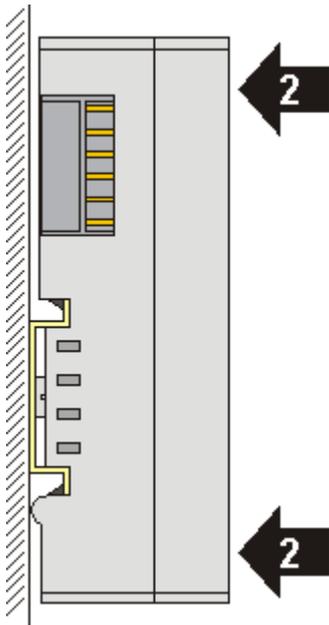
#### **Verletzungsgefahr durch Stromschlag und Beschädigung des Gerätes möglich!**

Setzen Sie das Busklemmen-System in einen sicheren, spannungslosen Zustand, bevor Sie mit der Montage, Demontage oder Verdrahtung der Busklemmen beginnen!

#### Montage

- Montieren Sie die Tragschiene an der vorgesehenen Montagestelle



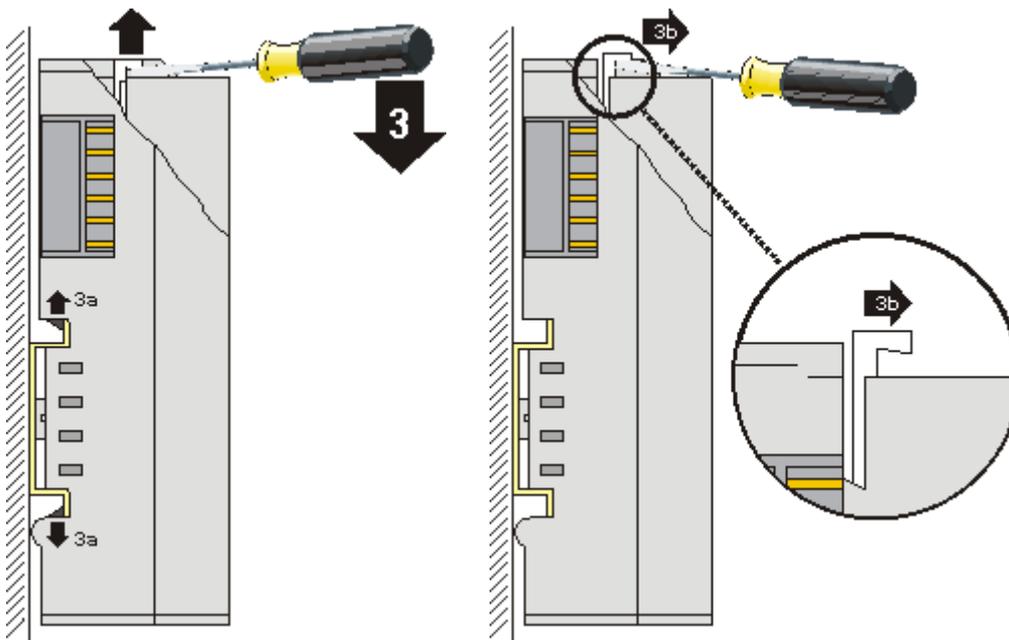


und drücken Sie (1) das Klemmenmodul gegen die Tragschiene, bis es auf der Tragschiene Einrastet (2).

- Schließen Sie die Leitungen an.

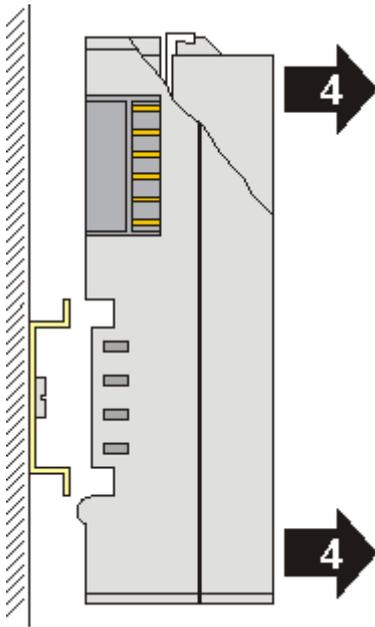
**Demontage**

- Entfernen Sie alle Leitungen. Dank der KM/EM-Steckverbinder müssen Sie hierzu nicht alle Leitungen einzeln entfernen, sondern pro KM/EM-Steckverbinder nur 2 Schrauben lösen um diese abziehen zu können (stehende Verdrahtung)!
- Hebeln Sie auf der linken Seite des Klemmenmoduls mit einem Schraubendreher (3) den Entriegelungshaken nach oben. Dabei
  - ziehen sich über einen internen Mechanismus die beiden Rastnasen (3a) an der Hutschiene ins Klemmenmodul zurück,
  - bewegt sich der Entriegelungshaken nach vorne (3b) und rastet ein



- Bei 32- und 64-kanaligen Klemmenmodulen (KMxxx4 und KMxxx8 bzw. EMxxx4 und EMxxx8) hebeln Sie nun den zweiten Entriegelungshaken auf der rechten Seite des Klemmenmoduls auf die gleiche Weise nach oben.

- Ziehen Sie (5) das Klemmenmodul von der Montagefläche weg.



### 3.1.3 Montage und Demontage

Die Klemmenmodule werden mit Hilfe einer 35 mm Tragschiene (z.B. Hutschiene TH 35-15) auf der Montagefläche befestigt.

#### ● Tragschienenbefestigung

**i** Der Verriegelungsmechanismus der Klemmen reicht in das Profil der Tragschiene hinein. Achten Sie bei der Montage der Komponenten darauf, dass der Verriegelungsmechanismus nicht in Konflikt mit den Befestigungsschrauben der Tragschiene gerät. Verwenden Sie zur Befestigung der empfohlenen Tragschienen unter den Klemmen flache Montageverbindungen wie Senkkopfschrauben oder Blindnieten.

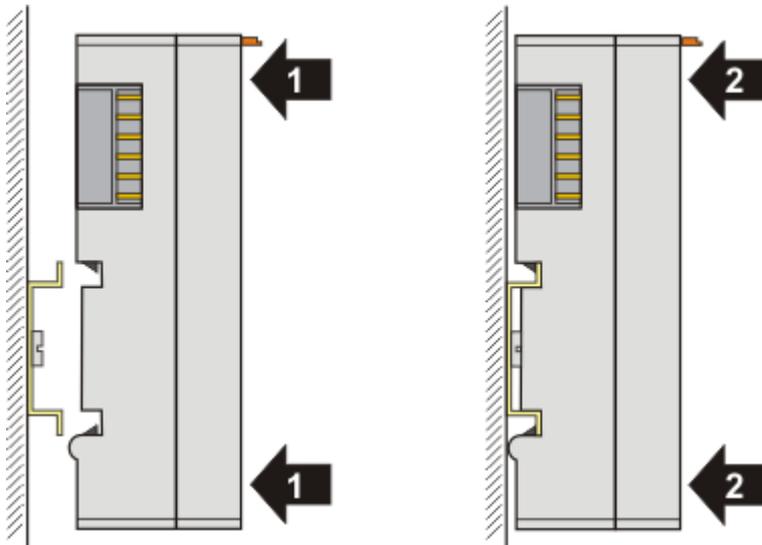
#### ⚠ WARNUNG

#### Verletzungsgefahr durch Stromschlag und Beschädigung des Gerätes möglich!

Setzen Sie das Busklemmen-System in einen sicheren, spannungslosen Zustand, bevor Sie mit der Montage, Demontage oder Verdrahtung der Busklemmen beginnen!

#### Montage

- Montieren Sie die Tragschiene an der vorgesehenen Montagestelle

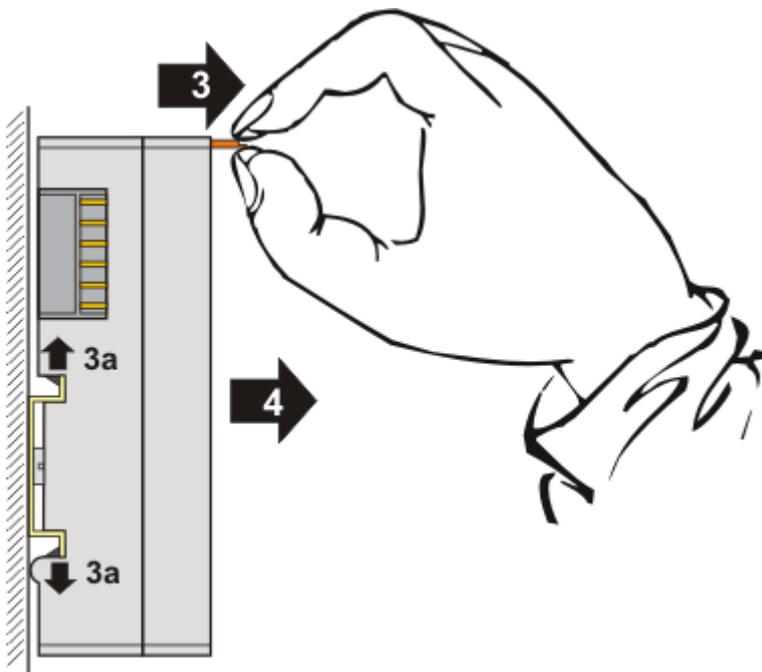


und drücken Sie (1) das Klemmenmodul gegen die Tragschiene, bis es auf der Tragschiene einrastet (2).

- Schließen Sie die Leitungen an.

**Demontage**

- Entfernen Sie alle Leitungen.
- Ziehen Sie mit Daumen und Zeigefinger die orange Entriegelungslasche (3) zurück. Dabei ziehen sich über einen internen Mechanismus die beiden Rastnasen (3a) an der Hutschiene ins Klemmenmodul zurück.



- Ziehen Sie (4) das Klemmenmodul von der Montagefläche weg.

### 3.1.4 Einbaulagen

#### HINWEIS

##### Einschränkung von Einbaulage und Betriebstemperaturbereich

Entnehmen Sie den technischen Daten zu einer Klemme, ob sie Einschränkungen bei Einbaulage und/oder Betriebstemperaturbereich unterliegt. Sorgen Sie bei der Montage von Klemmen mit erhöhter thermischer Verlustleistung dafür, dass im Betrieb oberhalb und unterhalb der Klemmen ausreichend Abstand zu anderen Komponenten eingehalten wird, so dass die Klemmen ausreichend belüftet werden!

##### Optimale Einbaulage (Standard)

Für die optimale Einbaulage wird die Tragschiene waagrecht montiert und die Anschlussflächen der EL/KL-Klemmen weisen nach vorne (siehe Abb. 1). Die Klemmen werden dabei von unten nach oben durchlüftet, was eine optimale Kühlung der Elektronik durch Konvektionslüftung ermöglicht. Bezugsrichtung "unten" ist hier die Erdbeschleunigung.

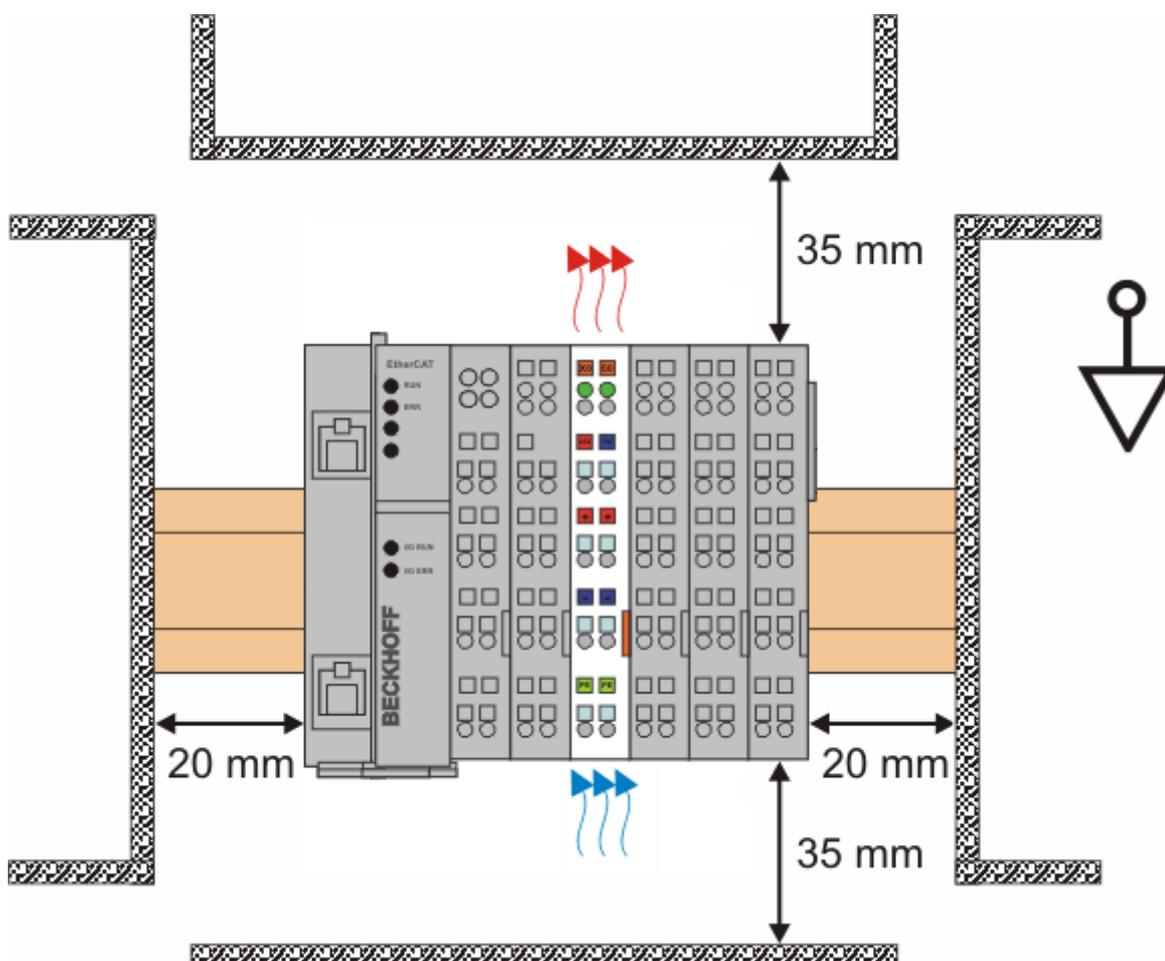


Abb. 1: empfohlene Abstände bei Standard Einbaulage

Die Einhaltung der Abstände nach Abb. 1 wird empfohlen.

##### Weitere Einbaulagen

Alle anderen Einbaulagen zeichnen sich durch davon abweichende räumliche Lage der Tragschiene aus, s. Abb 2.

Auch in diesen Einbaulagen empfiehlt sich die Anwendung der oben angegebenen Mindestabstände zur Umgebung.

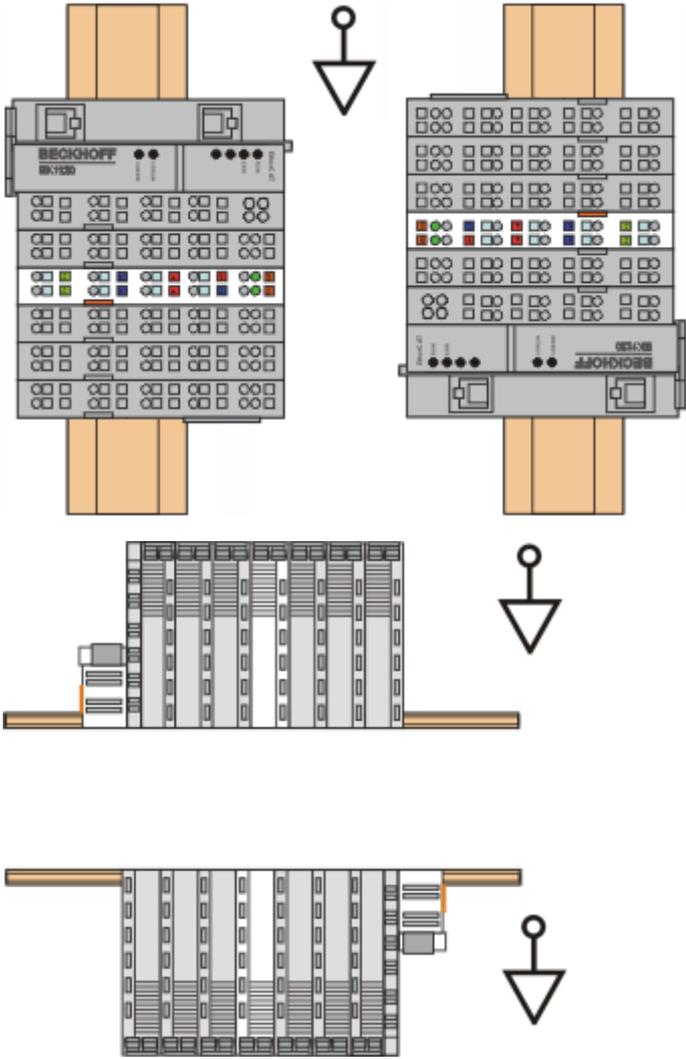


Abb. 2: weitere Einbaulagen

### 3.1.5 ATEX - Besondere Bedingungen

#### ⚠️ WARNUNG

**Beachten Sie die besonderen Bedingungen für die bestimmungsgemäße Verwendung von Beckhoff-Feldbuskomponenten in explosionsgefährdeten Bereichen (Richtlinie 94/9/EG)!**

- Die zertifizierten Komponenten sind in ein geeignetes Gehäuse zu errichten, das eine Schutzart von mindestens IP54 gemäß EN 60529 gewährleistet! Dabei sind die Umgebungsbedingungen bei der Verwendung zu Berücksichtigungen!
- Wenn die Temperaturen bei Nennbetrieb an den Einführungsstellen der Kabel, Leitungen oder Rohrleitungen höher als 70°C oder an den Aderverzweigungsstellen höher als 80°C ist, so müssen Kabel ausgewählt werden, deren Temperaturdaten den tatsächlich gemessenen Temperaturwerten entsprechen!
- Beachten Sie beim Einsatz von Beckhoff-Feldbuskomponenten in explosionsgefährdeten Bereichen den zulässigen Umgebungstemperaturbereich von 0 - 55°C!
- Es müssen Maßnahmen zum Schutz gegen Überschreitung der Nennbetriebsspannung durch kurzzeitige Störspannungen um mehr als 40% getroffen werden!
- Die einzelnen Klemmen dürfen nur aus dem Busklemmensystem gezogen oder entfernt werden, wenn die Versorgungsspannung abgeschaltet wurde bzw. bei Sicherstellung einer nicht-explosionsfähigen Atmosphäre!
- Die Anschlüsse der zertifizierten Komponenten dürfen nur verbunden oder unterbrochen werden, wenn die Versorgungsspannung abgeschaltet wurde bzw. bei Sicherstellung einer nicht-explosionsfähigen Atmosphäre!
- Die Sicherung der Einspeiseklemmen KL92xx dürfen nur gewechselt werden, wenn die Versorgungsspannung abgeschaltet wurde bzw. bei Sicherstellung einer nicht-explosionsfähigen Atmosphäre!
- Adresswahlschalter und ID-Switche dürfen nur eingestellt werden, wenn die Versorgungsspannung abgeschaltet wurde bzw. bei Sicherstellung einer nicht-explosionsfähigen Atmosphäre!

#### Normen

Die grundlegenden Sicherheits- und Gesundheitsanforderungen werden durch Übereinstimmung mit den folgenden Normen erfüllt:

- EN 60079-0: 2006
- EN 60079-15: 2005

#### Kennzeichnung

Die für den explosionsgefährdeten Bereich zertifizierten Beckhoff-Feldbuskomponenten tragen eine der folgenden Kennzeichnungen:



II 3 G Ex nA II T4

KEMA 10ATEX0075 X Ta: 0 - 55°C

oder



II 3 G Ex nA nC IIC T4

KEMA 10ATEX0075 X Ta: 0 - 55°C

### Seriennummer

Die Beckhoff-Feldbuskomponenten tragen eine Seriennummer, die wie folgt aufgebaut ist:

WW YY FF HH

WW - Produktionswoche (Kalenderwoche)

YY - Produktionsjahr

FF - Firmware-Stand

HH - Hardware-Stand

Beispiel mit Ser. Nr.: 35 04 1B 01:

35 - Produktionswoche 35

04 - Produktionsjahr 2004

1B - Firmware-Stand 1B

01 - Hardware-Stand 01

## 3.2 Diagnose

### 3.2.1 EL6631 - LEDs



## LEDs zur EtherCAT-Diagnose

LED		Anzeige	Beschreibung
RUN	grün	aus	Zustand der EtherCAT State Machine: <b>INIT</b> = Initialisierung der Klemme; <b>BOOTSTRAP</b> = Funktion für Firmware Updates der Klemme
		blinkt 200 ms	Zustand der EtherCAT State Machine: <b>PREOP</b> = Funktion für Mailbox-Kommunikation und abweichende Standard-Einstellungen gesetzt
		aus (1 s) an (200 ms)	Zustand der EtherCAT State Machine: <b>SAFEOP</b> = Überprüfung der Kanäle des Sync-Managers und der Distributed Clocks. Ausgänge bleiben im sicheren Zustand
		an	Zustand der EtherCAT State Machine: <b>OP</b> = normaler Betriebszustand; Mailbox- und Prozessdatenkommunikation ist möglich

## LED Diagnose PROFINET RUN/Err

Farbe grün	Farbe rot	Bedeutung
aus	blinkt 200 ms	Klemme startet
blinkt 200 ms	aus	Keinen Namen
1s aus, 200 ms an	aus	Keine IP-Adresse
an	aus	EL-Klemme ist parametrier

## LED Diagnose PROFINET Err

Farbe grün	Farbe rot	Bedeutung
aus	blinkt 500 ms	mit keinem Teilnehmer eine AR aufgebaut
aus	1s aus, 200 ms an	Mindestens ein Teilnehmer hat keine AR aufgebaut
1s aus, 200 ms an	aus	Mindestens ein Teilnehmer hat einen Fehler gemeldet, z. B. dass ein Modul-Unterschied vorliegt oder das Fehlerbit für eine IO-CR gesetzt ist (Problem Indicator)
blinkt 200 ms	aus	Mindestens ein Teilnehmer meldet, dass er im Status "Stopp" steht (Provider State Stop)
an	aus	Alle PROFINET Geräte sind im Datenaustausch

Liegen mehrere unterschiedliche Fehler vor, so wird immer der Fehler angezeigt, der in der Tabelle oben (oder höher) steht.

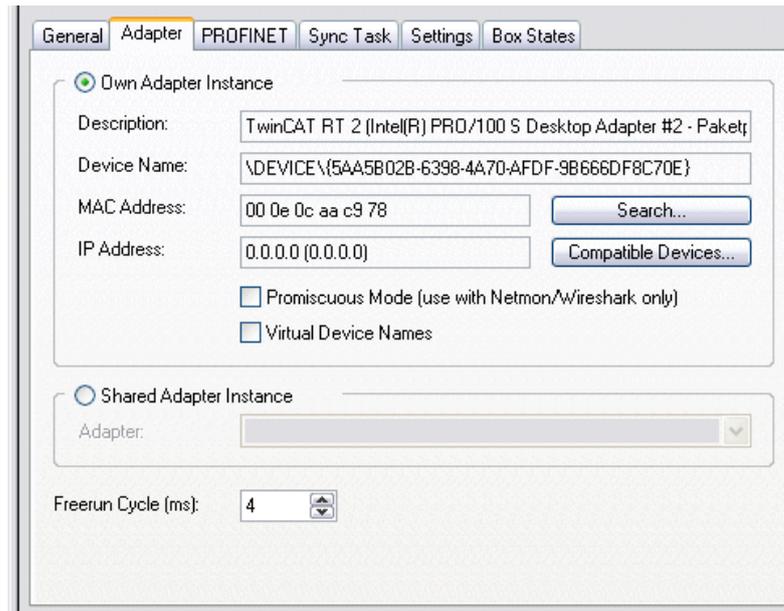
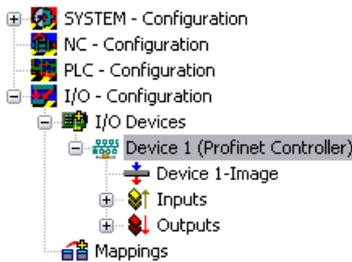
## LEDs im Hochlauf

Run	PN Run/Err	PN Err	Bedeutung
aus	aus	aus	Keine Spannung am E-Bus angeschlossen. Sollten EtherCAT Klemmen dahinter funktionieren, muss die EL6631 getauscht werden.
aus	aus	rot an	EL-Klemme läuft hoch, nach ca. 10 s sollte die LED aus gehen; ist das nicht der Fall, muss das EL6631 Modul getauscht werden.

## 4 PN Controller Protokoll

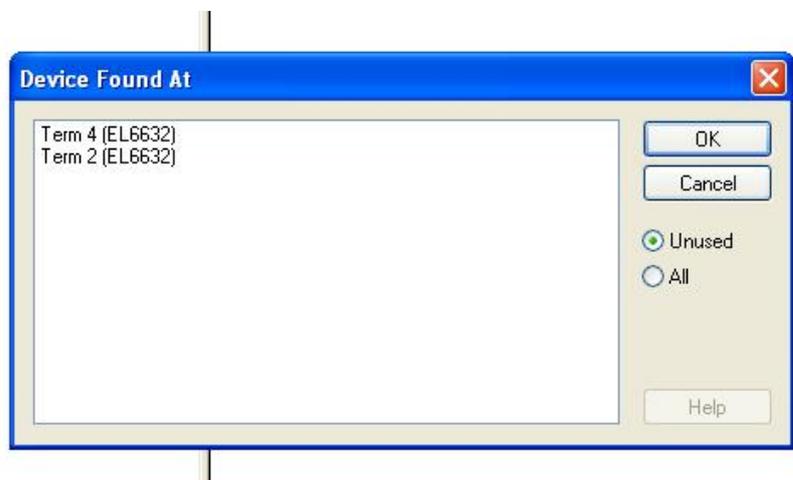
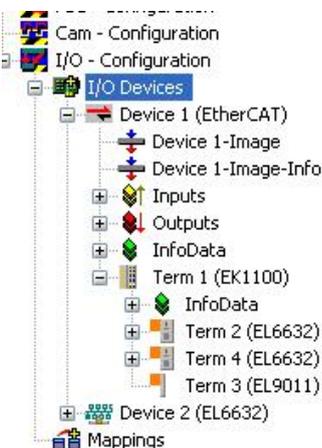
### 4.1 Einbindung des TwinCAT PROFINET Controller Protokolls über eine RealTimeEthernet Schnittstelle

Das Controller-Protokoll wird direkt an das I/O Device angefügt. Die zur Verfügung stehenden Netzwerkschnittstellen werden beim Anfügen direkt angezeigt und stehen nun zur Auswahl. Sollen diese im nachhinein geändert bzw. kontrolliert werden, so kann dies im Karteireiter "Adapter" erfolgen..

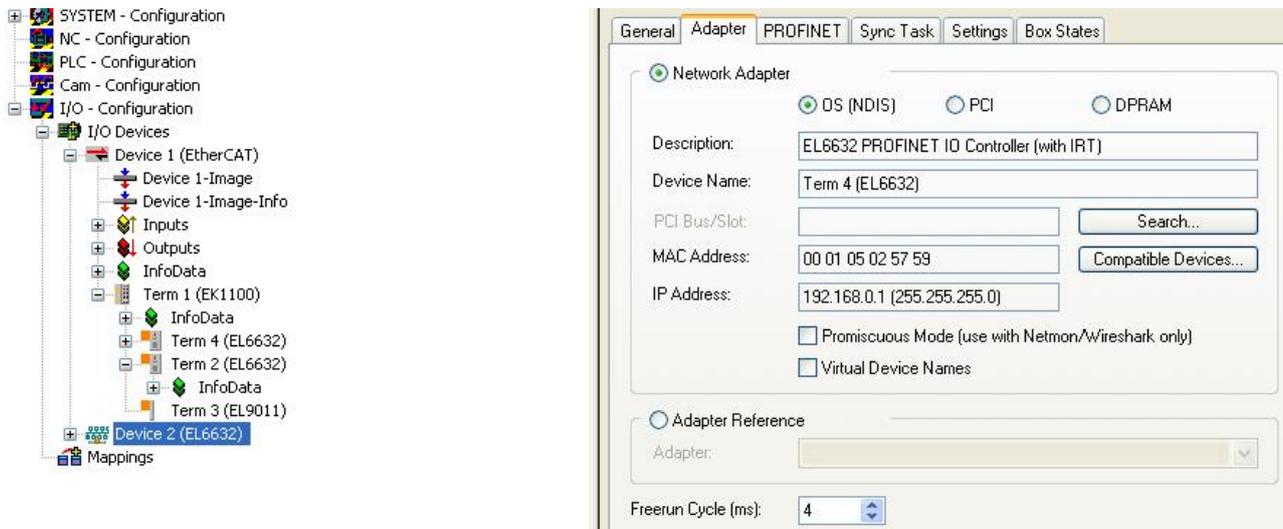


### 4.2 Einbindung des TwinCAT PROFINET Controller Protokolls über eine EL663x Schnittstelle

Das Controller Protokoll wird direkt an das I/O Device angefügt. Je nachdem welche Klemme genutzt werden soll (EL6631 oder EL6632), ist das entsprechende Protokoll auszuwählen. Befindet sich eine solche Klemme am projektierten EtherCAT-Strang, so wird beim Anfügen des Protokolls direkt der zugehörige Adapter angezeigt. Bei mehreren Klemmen kann die entsprechende ausgewählt werden.



Für den Betrieb mehrerer EL663x Klemmen muss das entsprechende PROFINET Protokoll mehrfach angefügt werden. Soll die Klemmenzuweisung im nachhinein geändert bzw. kontrolliert werden, kann dies im Karteireiter "Adapter" erfolgen.



## 4.3 Einstellungen / Diagnose

### 4.3.1 PROFINET

#### 4.3.1.1 AMS Settings

##### Textfeld Protocol AMS NetID

Das ist die NetID, über die das PROFINET Controller Protokoll via AMS erreicht werden kann.

##### Textfeld Protocol AMS PortNr

Das ist die PortNr, über die das PROFINET Controller Protokolls via AMS erreicht werden kann. Diese ist immer fest eingestellt auf 0xFFFF.

##### Textfeld Server AMS NetID

Das ist die NetID, an die vom PROFINET Treiber aus bestimmte AMS Nachrichten weitergeleitet werden (z.B. PN Records im Indexbereich 0x1000 - 0x1FFF). Dies ist derzeit immer die SystemNetId.

##### Textfeld Server AMS PortNr

Das ist die PortNr, an die vom PROFINET Treiber aus bestimmte AMS Nachrichten weitergeleitet werden (z.B. PN Records im Indexbereich 0x1000 - 0x1FFF). Dies ist per default der PLC Port 802 von Laufzeitsystem 1.

#### 4.3.1.2 PROFINET Key

##### Textfeld Key

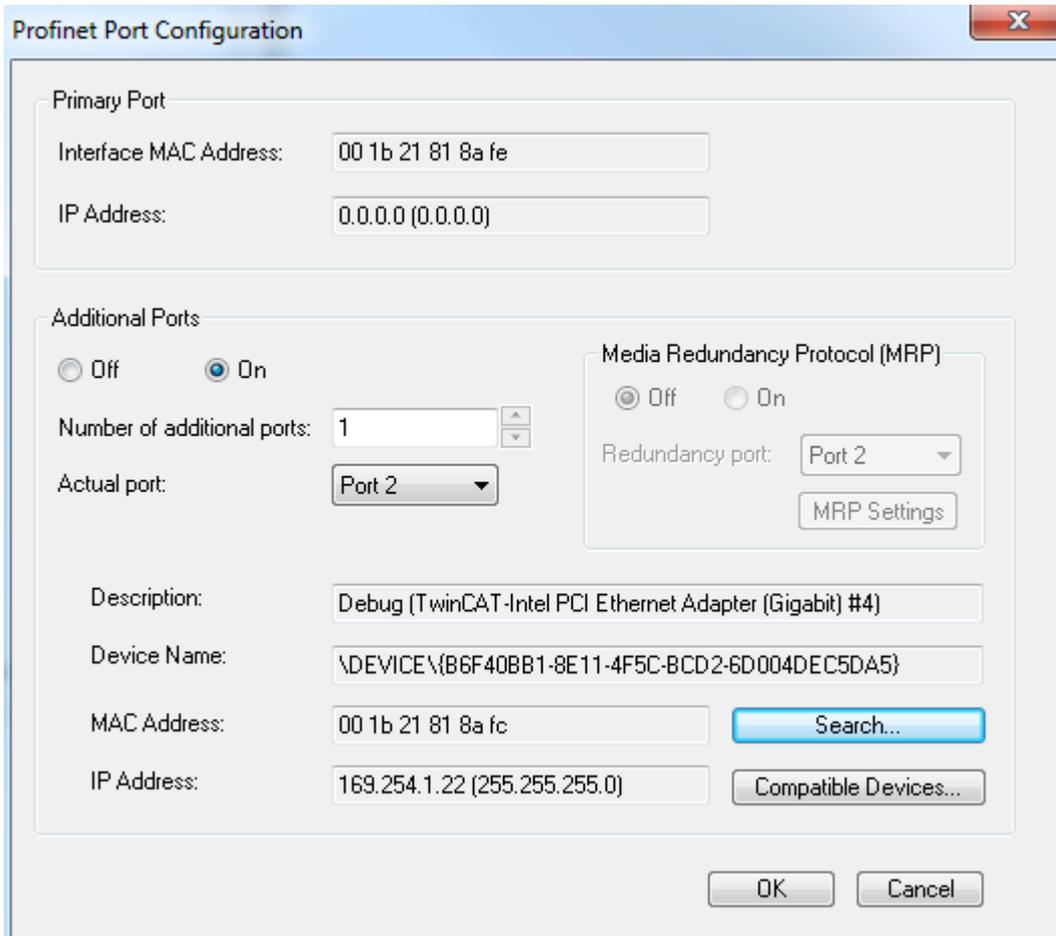
Hier kann der Schlüssel zur Freischaltung des PROFINET Controller Protokolls eingegeben werden. Wurde bereits ein gültiger Key eingegeben so steht die entsprechende Meldung in dem Fenster und es kann keine Eingabe mehr erfolgen.

**Button Insert Key...**

Hierüber wird der eingegebene Key übernommen. Der Key wird dabei gecheckt ob er gültig ist. Falls nicht kommt die entsprechende Meldung. Ist ein gültiger Key bereits eingetragen so wird dieser Button inaktiv.

**4.3.1.3 Button Port settings**

Diese Feature ist derzeit nur für das Realtime Ethernet Protokoll frei gegeben (keine EL663x). Hiermit kann anhand einer zweiten Netzwerkkarte (Intel-Chipsatz) ein zweiter PROFINET Port und somit ein intelligenter Switch realisiert werden. Es ist vorgesehen, dieses Feature x-fach zu wiederholen, derzeit ist es jedoch auf einen zusätzlichen Port begrenzt.

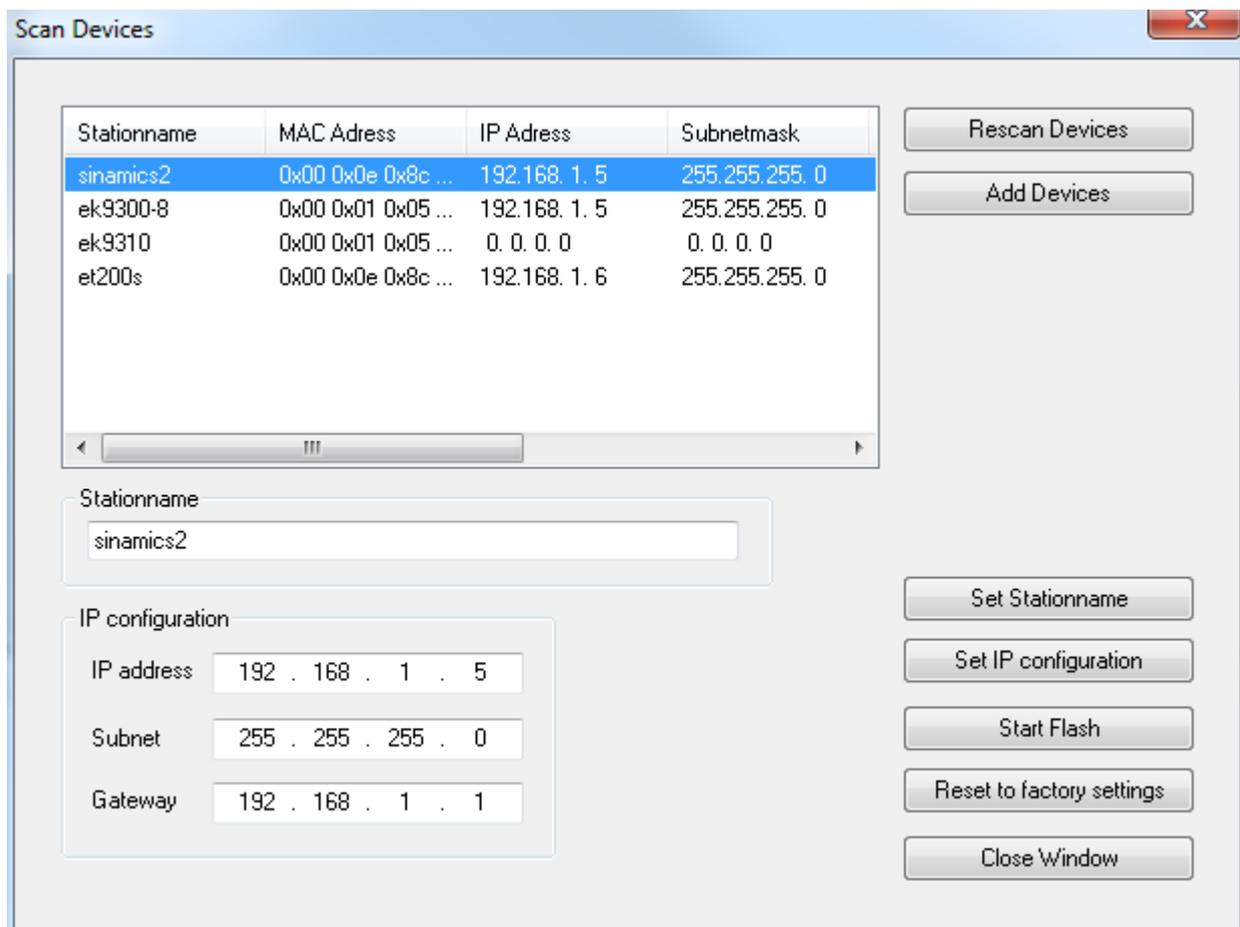


Zukünftig kann außerdem über dieses Menü die MRP (Media Redundancy Protocol) Funktionalität freigeschaltet werden; hierfür können diverse Einstellungen vorgenommen werden.

**4.3.1.4 Button Scan PNIO Devices**

Dieses Feature ist vergleichbar mit dem "ScanBoxes" Feature, welches allerdings nur im CONFIG Mode verfügbar ist.

Nach dem erfolgreichen Scannen öffnet sich folgender Dialog (wenn Geräte gefunden wurden).



Hier können verschiedene Einstellungen bzw. Projektierungen der Geräte erfolgen. Diese werden erst übernommen, wenn explizit der entsprechende Button betätigt wird. Beim Setzen des Namens ist darauf zu achten, dass nur PROFINET-konforme Zeichen verwendet werden. Das gilt auch für die IP Adresse; es sind nur gültige Kombinationen von IP und Subnetz zu verwenden. Name und IP werden beim Setzen vom PROFINET Device auf Richtigkeit geprüft. Ist dies nicht der Fall, wird der DCP\_SET mit einem Fehler quittiert. Durch Drücken des Rescan Buttons können vorgenommene Änderungen zurück gelesen werden.

Das ausgewählte Gerät kann außerdem signalisiert werden. Diese Funktionalität ist PROFINET-spezifisch. Wie die Signalisierung erfolgt ist jedoch herstellerspezifisch. Als Standard gilt jedoch, dass das Signal mit einer Frequenz von 2 Hz einzutreffen hat.

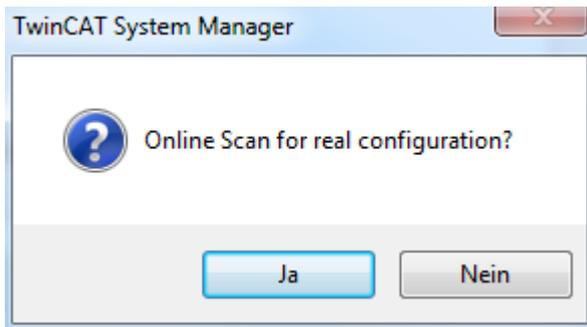
Als Beispiel meldet der Beckhoff Buskoppler BK9103 sich durch das abwechselnde Blinken zweier LEDs im 2 Hz Takt. Diese Funktion ist sehr hilfreich, um die Geräte in dieser Liste zu identifizieren. Durch erneutes Drücken des Buttons wird das Blinken wieder gestoppt. Das Blinken wird durch das Schließen des "Scan Devices" Fensters gestoppt.

Anschließend können ein oder mehrere Geräte mit der Strg-Taste markiert werden. Durch Drücken von "Add Devices" werden die ausgewählten Geräte in das Projekt übernommen.

### ● Geräte GSDML

**i** Die zugehörige Geräte GSDML muss sich im Ordner "..\TwinCAT\Io\ProfiNet" befinden!

Durch Betätigen von "Add Devices" öffnet sich die folgende Nachfrage:



#### Button "Ja":

Es wird zunächst versucht über einen impliziten Lesezugriff die ModuleldentNumber des DAPs (Device Access Point) zu ermitteln. Schlägt dies fehl, öffnet sich ein entsprechender Dialog mit den möglichen DAPs, worüber eine händische Auswahl erfolgen muss.

Sind alle Boxen angefügt, erfolgt automatisch ein "Reload Devices", d.h. dem PROFINET Treiber werden die angelegten Geräte (Adapter) übermittelt. Anschließend wird unterschieden, ob es sich bei der Box um ein normales Gerät oder um einen Antrieb mit Profidrive Unterstützung handelt.

Beim normalen Gerät erfolgt erneut über einen impliziten Lesezugriff das Auslesen der wirklichen Modulbestückung (ReallidentificationData). Bei einem Profidrive Gerät hingegen erfolgt das Auslesen der benötigten Informationen über einen Profidrive Zugriff. Hierfür wird eine Supervisor AR aufgebaut. Innerhalb dieser können die erforderlichen Schreibzugriffe erfolgen. Als Parameter Access Point wird hier das Interface Submodule am DAP genommen. Der Parameterzugriff erfolgt über den Datenrecord 47, ähnlich wie es bereits bei Profibus der Fall war. Beim Einsatz von Sinamics ist jedoch zu beachten, dass diese einen solchen Zugriff erst ab Version 4.3 SP2 unterstützen. Wird eine ältere Version verwendet, erscheint eine entsprechende Fehlermeldung und die Parametrierung muss händisch erfolgen.

Ist die automatische Modulparametrierung abgeschlossen, erscheint die Nachfrage zum automatischen Einlesen der Portdaten. Hierbei wird wiederum über einen impliziten Lesezugriff die Portverschaltung der einzelnen Geräte ausgelesen.

Die reale Portverschaltung muss für für verschiedene Dienste bekannt sein. Das können einfach nur Diagnosedienste sein, aber auch der automatische Geräteanlauf setzt dies voraus (über Alias), oder aber die Erstellung der IRT-Planung.

Wird dieser Dialog mit "Nein" quittiert oder der Lesezugriff ist fehlgeschlagen, kann eine solche Verschaltung im TwinCAT-Projekt an den einzelnen Ports auch händisch erfolgen.

Wurde die Portverschaltung erfolgreich automatisch generiert, kommt im Falle eines IRT Controllers (z.B. Projektierung an einer EL6632) noch die Nachfrage, ob automatisch alle Geräte (insofern sie es unterstützen) in den IRT Mode (RTClass3) geschaltet werden sollen.

Wird dies bejaht, wird außerdem die Kabellänge an allen projektierten Ports auf 10 m Kupferleitung gesetzt. Der IRT Algorithmus benötigt diese Information zur Berechnung der Signallaufzeiten. Die exakte Kabellänge ist hier weniger wichtig (ca. +/-10 m), denn die Laufzeitverzögerungen sind bei 100MBit/s eher gering (5ns/m). Soll die automatische Umschaltung nicht sofort erfolgen, können diese Punkte auch im Nachhinein entweder am Protokoll oder aber an den einzelnen Geräten (am Interface- bzw. Portsubmodul) geändert werden.

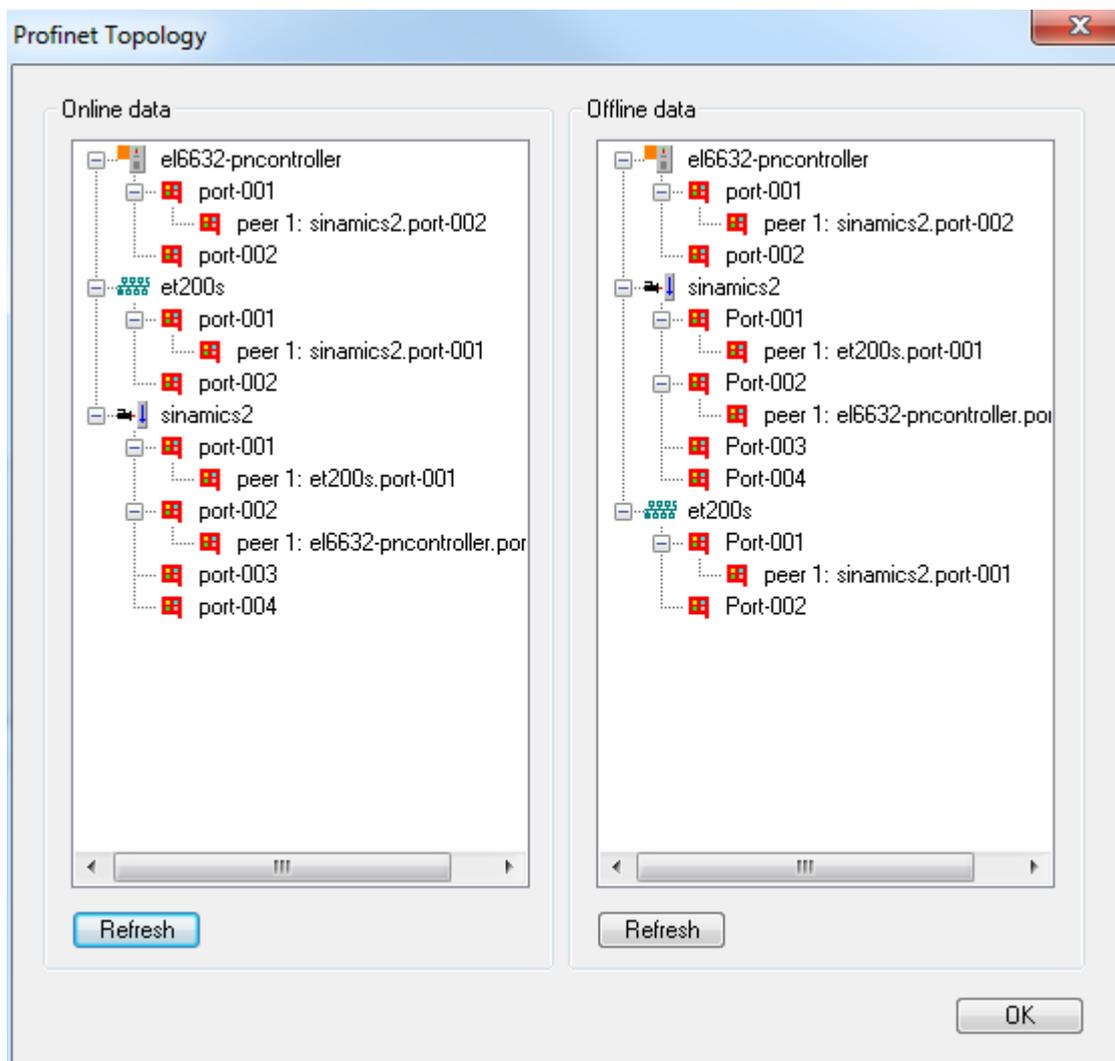
#### Button "Nein":

Bei jedem Gerät wird geprüft, ob die GSDML im entsprechenden Ordner vorhanden ist ("..\TwinCAT\Io\ProfiNet"). Ist dies der Fall, wird die Liste der möglichen DAPs eingelesen. Danach wird ein Auswahldialog geöffnet, um den entsprechenden DAP auszuwählen.

Sind die Geräte im Projekt angefügt, kann anschließend auf das API unter der Box gegangen und hierüber händisch die Module und Submodule angefügt werden.

### 4.3.1.5 Button Topology

Über diesen Dialog kann die Offline-Topologie mit der Online-Topologie verglichen werden.

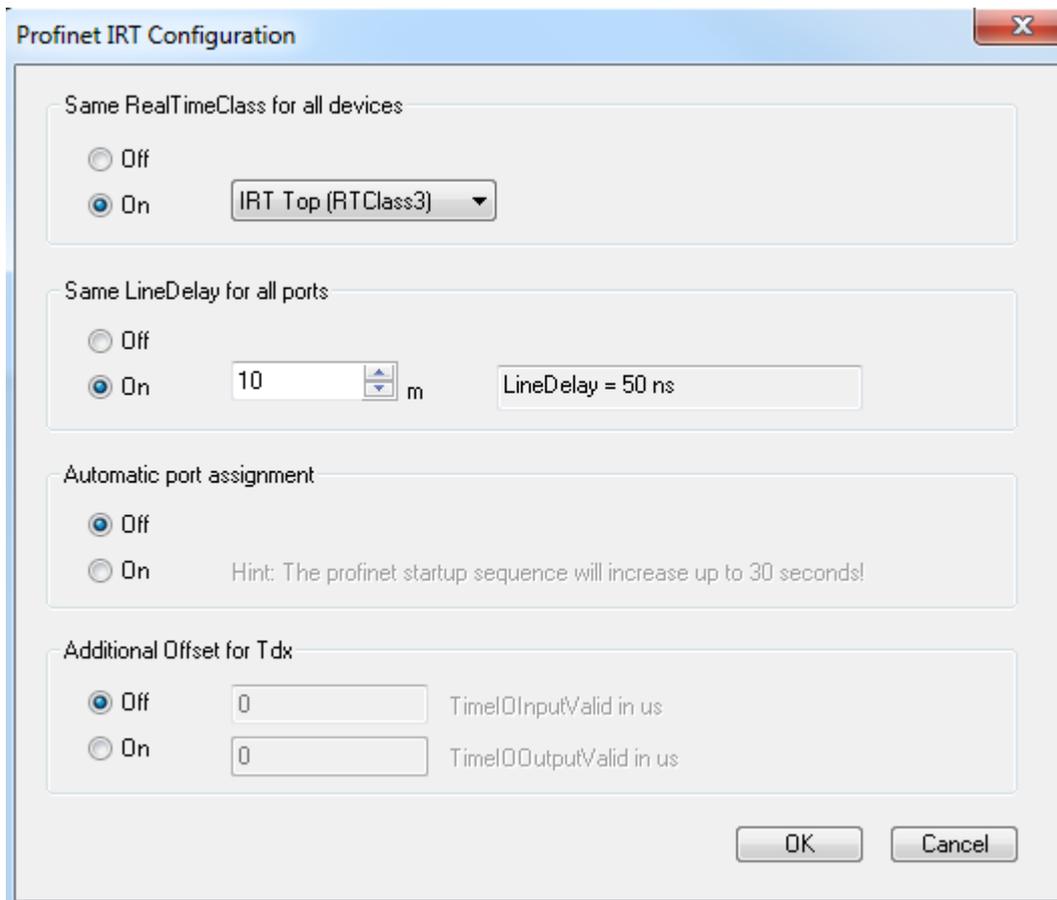


Es kann in der Online-Ansicht durchaus möglich sein, dass ein Gerät an einem Port mehrere Partner hat. Dies ist z.B. der Fall, wenn ein Switch im PROFINET genutzt wird, der kein LLDP (Protokoll zur Nachbarschaftskennung) unterstützt.

In der Offline-Ansicht können wiederum Partner zugewiesen worden sein, die im Projekt nicht vorhanden sind. Dies erfolgt, wenn beim Scannen und automatischen Anfügen das Einlesen der Port-Eigenschaften aktiviert wurde. In diesem Fall hat das Gerät einen "Nachbarn", der im Projekt übernommen wird, die zugehörige Device Box fehlt aber im \*.tsm File. Bei Aktivierung dieses Projektes wird der im \*.tsm File nicht vorhandene "Nachbar" im Treiber ignoriert.

#### 4.3.1.6 Button IRT Config

Dieses Menü ist nur für einen IRT-fähigen Controller frei gegeben (derzeit nur EL6632). Hierüber kann eine globale Einstellung für alle projektierten Devices vorgenommen werden.



Hier kann zum einen die Art der Kommunikation angegeben werden. Derzeit wird nur RTClass1 (RT) und RTClass3 (IRT Top) unterstützt.

Außerdem besteht in diesem Dialog die Möglichkeit eine allgemeine Kabellänge anzugeben (nur für IRT). Hierbei reicht ein ungefährender Wert bzw. die max. Kabellänge aus, denn für die Berechnung der IRT-Kommunikation ist dieser Wert ein eher geringer (bei 100 MBaud und Kupferleitung 5 ns/m). Zur Optimierung kann dieses Feature im Nachhinein auch wieder ausgeschaltet werden und dann für jeden einzelnen Teilnehmer die genaue Leitungslänge eingetragen werden (an den Portsubmodulen).

Des Weiteren besteht hier die Möglichkeit ein "Automatic port assignment" zu aktivieren. Hierdurch ist die im TwinCAT Projekt eingestellte Portverschaltung irrelevant. Es wird vor jedem Neustart der PN-Kommunikation die Topologie ausgelesen und anhand dieser die IRT-Kommunikation berechnet. Das hat den Vorteil, dass evtl. Verkabelungsfehler minimiert werden. Außerdem können die Ports einfach umgesteckt werden ohne dass das TwinCAT-Projekt geändert und neu geladen werden muss. Lediglich ein Neustart der PN Kommunikation ist erforderlich (z.B. Klemme in PREOP schalten oder Kabel ziehen). Dadurch kann sich der Hochlauf der PROFINET-Kommunikation auf bis zu 30sec verlängern. Als Grund ist der TTL (TimeToLive) Faktor in der LLDP MIB zu nennen. Diese sind per default Default auf 20 sec gestellt, d.h. erst nach dieser Zeit kann sichergestellt werden, dass die ausgelesene Portverschaltung auch die aktuelle ist.

In diesem Menü kann außerdem ein zusätzlicher Offset für alle  $T_i$  /  $T_o$  Werte vorgegeben werden.

### 4.3.2 Taskkonfiguration

Das PROFINET Controller Protokoll muss immer mit einer Task verknüpft werden. Mit der eingestellten Task-Zeit wird auch das Protokoll bearbeitet. Theoretisch kann der Controller z.B. auch über eine PLC- oder NC Task mit bearbeitet werden. Wird aber beispielsweise ein PLC Projekt gestoppt (z.B. durch Restart oder Debugging) hat das zur Folge, dass auch der PROFINET Teil gestoppt wird. Um einen solchen Nebeneffekt zu vermeiden ist es ratsam immer eine freilaufende SyncTask anzulegen.

The screenshot shows the 'Sync Task' configuration window. At the top, there are tabs for 'General', 'Adapter', 'PROFINET', 'Sync Task', 'Settings', 'Box States', and 'Diag History'. The 'Sync Task' tab is active. Below the tabs, there are two main sections:

- Settings:** Contains two radio buttons: 'Standard (via Mapping)' (unselected) and 'Special Sync Task' (selected). Below these is a dropdown menu showing 'Task 2' and a button labeled 'Create new I/O Task'.
- Sync Task:** Contains a text field for 'Name' with the value 'Task 2'. Below it, 'Cycle ticks' is set to '1' in a spinner box, followed by a text field with '1.000' and the unit 'ms'. There is an unchecked checkbox labeled 'Adjustable by Protocol'. Below that, 'Priority' is set to '1' in a spinner box.

Es ist darauf zu achten, dass sich der Takt der Task in einem PROFINET-Takt befindet. D.h. für PROFINET ist der Grundtakt 31,25  $\mu$ s. Dieser Takt wird dann immer mit dem SendClockFactor (SCF) multipliziert und man erhält den Grundtakt. Für RTClass1 ist in der Regel der SendClockFactor auf 32 gesetzt. Für den Beckhoff PROFINET Controller ist für RTClass1 auch dies der minimale PN-Takt. Damit ergibt sich die kleinste Zykluszeit von 1 ms. Die weiteren Untersetzungen erfolgen anhand eines ReductionRatioFactors. Dieser entspricht immer dem Vielfachen des minimalen PN-Takt. Für RTClass1 der kleinste Takt immer zu verdoppeln (zulässige Zykluszeiten (für RTC1) bei einem SCF von 32 sind 1, 2, 4, 8, ... ,512).

Um für RTClass3 auch schnellere Zykluszeiten zu realisieren kann und muss der SCF verringert werden. Dies ist derzeit für einen Beckhoff IRT Controller (EL6632) minimal 16, was wiederum einem Grundtakt von 500  $\mu$ s entspricht. Bei einer solchen Verringerung des PROFINET-Taktes ist auch drauf zu achten, dass auch die Zeit der triggernden Task entsprechend angepasst werden muss.

### 4.3.3 PROFINET Controller spezifische Einstellungen

Über den Karteireiter "Settings" können Einstellungen die direkt den Controller betreffen vorgenommen werden.

The screenshot shows the 'Settings' tab of the Beckhoff PN Controller configuration interface. It contains several sections for configuring network and system parameters:

- IP configuration:** Fields for IP address (192.168.1.1), Subnet (255.255.255.0), and Gateway (192.168.1.1). A button labeled 'Set IP settings...' is located to the right.
- Name of Pnlo Controller Station:** A text field containing 'tc-pncontroller' and a button labeled 'Set System name...'.
- VendorId:** A field containing '0x0120'.
- DeviceId:** A field containing '0x0023'.
- Server UDP Port:** A field containing '0xEE48'.
- Client UDP Port:** A field containing '0xEA60'.
- StationName settings:** A section with a checkbox labeled 'Automatic NameOfStation assignment' which is currently unchecked.

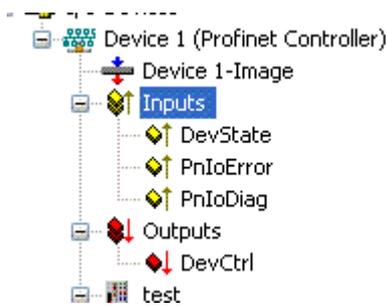
Hier kann eine IP-Einstellung erfolgen. Die Wahl des Adressbereiches muss nicht mit den Einstellungen der Netzwerkkarte übereinstimmen. Die PROFINET-Kommunikation spannt ein eigenes Netz auf, welches hier gewählt werden kann. Die im obigen Bild angezeigten IP Settings sind die Default Einstellungen. D.h., wird nichts geändert nutzt der Controller diese Einstellungen. Das Gleiche gilt für den Controller-Namen (Systemname). Zum Ändern beider Einstellungen muss der entsprechende Button gedrückt werden. Hierüber erfolgt eine Überprüfung auf korrekte Eingabe (z.B. das Format des Controller-Namens muss der PN Spec. entsprechen). Diese Daten werden dann permanent übernommen. Beim Ändern des Subnetzes oder Gateways werden die Einstellungen auch auf evtl. projektierte Geräte übernommen. Es besteht auch die Möglichkeit, diese Settings über ein Supervisor Tool zu ändern.

Außerdem kann in diesem Dialog die VendorID und DeviceID des Controllers ausgelesen werden. Auch eine Einstellung des verwendeten Server- und Client UDP Ports kann hier erfolgen. Die Defaulteinstellungen sollten hier in den meisten Fällen aber ausreichend sein.

Des Weiteren besteht in diesem Dialog die Möglichkeit, einen automatischen PROFINET-Anlauf nach einem Gerätetausch zu ermöglichen (auch für Geräte ohne Wechselmedium). Für die korrekte Funktionsweise muss einmal die Solltopologie vorgegeben werden. Anhand dieser Informationen kann der Controller nach den Aliasnamen der einzelnen Geräte fragen. Jedes Gerät das Aliasnamen unterstützt, generiert für jeden seiner Ports einen solchen Namen. Dieser setzt sich aus den Nachbarschaftskennungen zusammen (PortId.ChassisId). Wird nach diesem Namen gefragt, antwortet das "neue" Gerät. Bei korrekter VendorId und DeviceId wird das Gerät mit dem eigentlichen Namen benannt und es kann anschließend ein normaler PROFINET-Hochlauf erfolgen. Mit diesem Mechanismus könnte auch eine komplette PROFINET-Anlage Anlaufen, ohne das ein einziges Gerät vorher benannt wurde.

#### 4.3.4 Analyse der Box States

Direkt unter dem PROFINET Controller Protokoll gibt es ein Sammel-PROFINET Error und einen Sammel-PROFINET Status. Beide geben die Anzahl der Geräte wieder, bei denen ein Problem aufgetreten bzw. bei welchen eine Diagnose verfügbar ist. D.h. der Fehler zeigt mögliche Probleme beim Verbindungsaufbau an oder Gründe für einen Abbruch. Die Diagnose gibt Status-Infos über eine bestehende Verbindung.



- PnIoError - Anzahl an PROFINET IO Geräten, die einen Fehler haben
- PnIoDiag - Anzahl an PROFINET IO Geräten, die eine Diagnose anstehen haben

Welches Gerät bzw. Box ein Problem hat kann, im Protokoll unter Box States auf einen Blick geprüft werden.

StationName	Box PnIoState	Box PnIoDiag
test	No Error (0x0)	Diagnosis available (0x12)
bk9103-11	No Error (0x0)	Diagnosis available (0x2)
bk9103-6	No Error (0x0)	Diagnosis available (0x418)
bk9103-4	No Error (0x0)	Diagnosis available (0x2)

Aktuell werden folgende Fehlermeldungen über den "PnIoState" angezeigt.

Nummer	Text	Beschreibung	Abhilfe / Grund
0	No error	Kein Fehler	Kein Fehler
1	PROFINETDevice state machine is in boot mode	PROFINET Device StateMachine ist noch in der Hochlauf Phase	Kein Fehler, warten
2	Device not found	Gerät antwortet nicht auf den Identify Request	Verbindung prüfen, Gerät angeschlossen, wurde das Gerät mit dem richtigen Namen benannt?
3	The stationname is not unique	Stationsname ist nicht eindeutig	Es gibt zwei oder mehr Geräte mit dem selben PROFINET-Namen im Netzwerk. Eine korrekte Identifizierung kann nicht erfolgen.
4	IP could not set	IP Adresse konnte nicht gesetzt werden	Das PROFINET Gerät hat aus irgendwelchen Gründen das Setzen der IP settings abgelehnt. Prüfen, ob die IP-Einstellungen korrekt sind.
5	IP conflict	Im Netzwerk trat ist ein IP-Konflikt aufgetreten	Eine mögliche Ursache ist das mehrere Geräte die gleiche IP-Adresse haben.
6	DCP set was not successful	Auf einen DCP Set kam keine bzw. eine fehlerhafte Antwort	Verbindung prüfen, Gerät angeschlossen, wurde das Gerät mit dem richtigen Namen benannt?
7	Watchdog error	Die Verbindung wurde mit einem Watchdog-Fehler abgebrochen	Zykluszeit prüfen, Verbindung prüfen, ggf. Watchdog-Faktor erhöhen.

Num-mer	Text	Beschreibung	Abhilfe / Grund
8	Datahold error	Die Verbindung wurde mit einem Datahold-Fehler abgebrochen	Frame Datenstatus war für die Länge des DataHoldTimers ungültig. Evtl. Gerät neu starten.
9	RTC3: Sync signal could not started	Nur für IRT: Das Sync-Signal konnte nicht gestartet werden.	EtherCAT Sync Signal korrekt bzw. Sync0 gestartet?
10	PROFINET Controller has a link error	Der PROFINET Controller hat keinen Link	Kabel und Verbindung checken.
11	The aliasname is not unique	Der Aliasname ist nicht eindeutig	Es gibt zwei oder mehr Geräte mit dem selben Alias Namen im Netzwerk. Dieser setzt sich aus Nachbarschaftsinformationen zusammen (PortId.ChassisId). Eine korrekte Identifizierung kann nicht erfolgen.
12	The automatic name assignement isn't possible - wrong device type	Das automatische Setzen des Namens ist nicht möglich	An der projektierten Position befindet sich nicht das erwartete PROFINET-Gerät (VendorId oder DeviceId stimmen nicht überein). Somit ist kein automatisches Benennen und damit Geräteanlauf möglich.
31	only for EtherCAT gateways: WC-State of cyclic EtherCAT frame is 1	Nur für EL6631: EtherCAT WC State ist auf 1	Am EtherCAT Master + Slave den Mode checken (OP?).

Im "BoxPnIoDiag" kann im Gegensatz zum State auch mehr als ein Zustand gleichzeitig angezeigt werden, d.h. das Ganze ist bitcodiert und es können bis zu 16 Infos angezeigt werden. Aktuell werden folgende Zustände dargestellt.

- 0x0000 = No diagnosis
- 0xXXX1 = IOC-AR is not established
- 0xXXX2 = IOC-AR is established
- 0xXXX4 = IOC-AR is established but no ApplReady
- 0xXXX8 = IOC-AR is established but module difference
- 0xXX1X = At least one AlarmCR get diagnosis alarm
- 0xX1XX = At least one InputCR is invalid
- 0xX2XX = At least one InputCR Provider is in stop
- 0xX4XX = At least one InputCR Problemindicator is set
- 0x1XXX = At least one OutputCR is invalid
- 0x2XXX = At least one OutputCR Provider is in stop
- 0x4XXX = At least one OutputCR Problemindicator is set

Es werden hier zum einen Infos über den Zustand der IO Controller Single AR angezeigt. Außerdem werden aus den Frame-Datenstati der einzelnen CRs Sammelstati gebildet. Das gilt für die Input- und die Output CRs (aktuell ist nur eine, zukünftig sind mehrere CRs möglich). Des weiteren wird im "PnIoDiag" auch ein PROFINET Alarm angezeigt

**Auslesen über ADS**

Das Auslesen des Box-Status kann über einen ADS Read erfolgen.

ADS Read:

NetId = AMSNETID des PROFINET Controllers

Port = BoxPort (0x1000 + BoxId)

Indexgroup = 0xF829

IndexOffset = 0

Length = sizeof(TPnIoDeviceDiagData);

wobei:

```
typedef struct
{
    WORD pnioState;
    WORD pnioDiag;
    WORD NrOfInputCRs;
    WORD NrOfOutputCRs;
    WORD reserved[8];
} TPnIoDeviceDiagData, *PTPnIoDeviceDiagData;
```

### Auslesen über CoE (für EL663x)

Das Auslesen des Box-Status kann für die EL663x auch über CoE erfolgen. Hierfür ist der Index 0xAyy0 (wobei yy die Adapter / Device Nummer ist ) und der Subindex 0x001 zu nehmen.

## 4.3.5 Diagnose Historie am Controller Protokoll

Über den Karteireiter "Diag History" können geloggte Diagnose-Meldungen vom Controller Protokoll ausgelesen werden. Der Diagnosepuffer arbeitet als Ringpuffer mit einer derzeitigen Größe von max. 1000 Einträgen.

Type	Timestamp	Message	AddInfo	MessageID
Warning	23.09.2011 13:45:56 613 ms	ek9300-1: AR got diagnosis alarm.	Yes	11
Warning	23.09.2011 13:45:56 609 ms	ek9300-1: AR got diagnosis alarm.	Yes	10
Info	23.09.2011 13:45:56 603 ms	ek9300-1: AR is established (got ApplReady).	No	9
Info	23.09.2011 13:45:53 541 ms	ek9300: AR is established (got ApplReady).	No	8
Info	23.09.2011 13:45:52 664 ms	ek9300: Controller send PmEnd.	No	7
Info	23.09.2011 13:45:52 601 ms	ek9300: Controller start the parameterization.	No	6
Info	23.09.2011 13:45:52 468 ms	ek9300: Controller send ConnectReq to device.	No	5
Info	23.09.2011 13:45:52 278 ms	ek9300-1: Controller send PmEnd.	No	4
Info	23.09.2011 13:45:52 245 ms	ek9300-1: Controller start the parameterization.	No	3
Info	23.09.2011 13:45:52 236 ms	ek9300-1: Controller send ConnectReq to device.	No	2
Error	23.09.2011 13:45:44 617 ms	ek9300-1: AR is released.	No	1
Error	23.09.2011 13:45:44 617 ms	ek9300-1: AR send error alarm.	Yes	0

Diagnosis appears alarm (0x0001)  
The diagnosis alarm received from:  
API Number 0x00000000, Slot Number 0x0005, Subslot Number 0x0001

Die möglichen Fehler sind in drei Arten gruppiert:

- Info: z.B. Informationen zum Verbindungsaufbau
- Warning: z.B. PROFINET Diagnose Alarme
- Error: z.B. Verbindungsabbruch

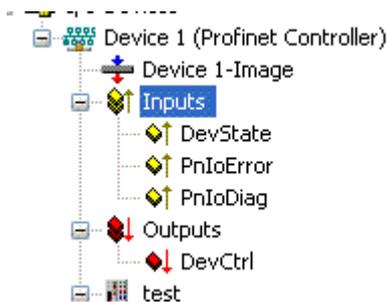
Über "AddInfo" wird angezeigt, ob zusätzliche Informationen zu dem Ereignis vorliegen. Ist dieses mit "Yes" gekennzeichnet, wird durch ein Klicken auf die entsprechende Meldung die Zusatzinformation abgeholt und angezeigt. Handelt es sich um einen Diagnosealarm ("Diagnosis appears") so können auf den entsprechenden Ebenen (Gerät, API oder Modul) die genauen Diagnoseinformationen abgeholt werden.

Über den Button "Clear Diag History" wird der komplette Diagnosepuffer geleert.

Über den Button "Export Diag History" besteht die Möglichkeit, die angezeigten Meldungen in einem .TXT-File abzuspeichern.

### 4.3.6 Zyklische Daten

Direkt unter dem PROFINET Controller Protokoll gibt es einige zyklischen Prozessdaten. Diese Daten werden lediglich zwischen PROFINET Treiber und System Manager ausgetauscht. Sie dienen zur allgemeinen Information über den Zustand der PROFINET-Kommunikation.



In der Variable "DevState". befinden sich Informationen über den phys. Kommunikationszustand des Controllers, wie z.B. der Linkstatus oder ob die Senderesourcen noch ausreichen.

Die weiteren Variablen sind der Sammel-PROFINET Error und der Sammel-PROFINET Status. Beide geben die Anzahl der Geräte wieder, bei denen ein Problem aufgetreten bzw. bei welchen eine Diagnose verfügbar ist. D.h. die Error-Variable zeigt mögliche Probleme beim Verbindungsaufbau an oder Gründe für einen Abbruch. Die Diagnose-Variable gibt Status Infos über eine bestehende Verbindung.

Die Ausgangsvariable "DevCtrl" hat zur Zeit keine Funktion.

Zur weiteren Information bitte auch das Kapitel "[Box States \[► 35\]](#)" lesen.

### 4.3.7 Azyklische Daten

Um azyklische Daten zu verschicken, werden die ADS-Bausteine verwendet. Diese greifen dann auf die Record-Daten des PROFINET zu. Damit azyklische Daten gelesen bzw. beschrieben werden können, muss sich das PROFINET Device im Datenaustausch befinden.

Es wird eine *ADSReadWrite* abgesetzt.

#### ADS Einstellungen

*AMSNetID*: Die AMSNetID des Profinet Controllers

*PORT*: Port Nummer des Device (Entnehmen sie diese des SystemManagers)

*Index GROUP*: 0x0000\_F823

*Index OFFSET*: 0x0000\_0000

#### DATEN

```
typedef struct {
    WORD          RW;
    #define       PN_READ          0
    #define       PN_WRITE        1
    WORD          NrOfAR;
    DWORD         API;
    WORD          Slot;
```

```
WORD          SubSlot;
PNIO_RECORD   RecordData;
} PNIO_CONFIGRECORD
```

nRW	nNr	nAPI	InSlot	SubSlot	nIndex	nLen	nTrans	nLenA
2 Byte	2 Byte	4 Byte	2 Byte	2 Byte	2 Byte	2 Byte	2 Byte	2 Byte

**Beispiel:**

Senden einer Lese-Anforderung für I&M Funktion 0

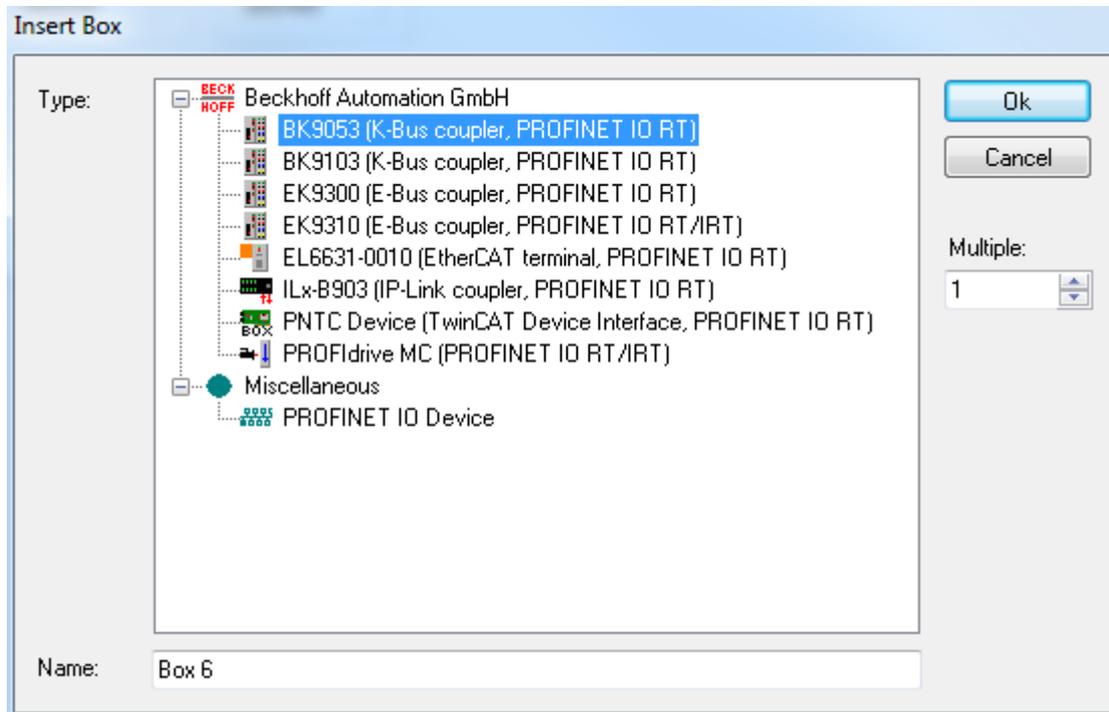
nRW	nNr	nAPI	InSlot	SubSlot	nIndex	nLen	nTrans	nLenA
00 00	00 00	00 00 00 00 0	00 00	01 00	F0 AF	00 00	01 00	00 00

<b>HINWEIS</b>
<p><b>Speichergröße beachten</b></p> <p>Achten Sie auf einen ausreichend großen Empfangsspeicher.</p>

## 5 Geräte am Protokoll

### 5.1 PROFINET Devices anfügen

Durch Klicken der rechten Maustaste auf das Protokoll "Append Box" wählen. Danach wird der folgende Dialog geöffnet:



Hier besteht die Möglichkeit, verschiedene PROFINET Devices auszuwählen. Bei den Beckhoff Geräten wird nach der GSDML unter einem definierten Pfad gesucht (..\TwinCAT\IO\PROFINET ). Diese sollten mit der TwinCAT-Installation bereits vorhanden sein. Gibt es hier mehrere GSDMLs für das gleiche Gerät, wird die mit dem neuesten Datum genommen. Wird keine Gerätebeschreibung gefunden, so erscheint eine entsprechende Fehlermeldung. Es kann entweder die GSDML in den Ordner kopiert und das Menü erneut geöffnet werden, oder aber es wird die gleiche Vorgehensweise wie für die Fremdgeräte gewählt. Klickt man auf "PROFINET IO Device", wird die Möglichkeit angeboten im Windows Explorer zu der entsprechenden GSDML zu navigieren. Diese wird dann in das Projekt integriert.

Als Default-Name wird dafür der DNS-Name aus der GSDML genommen. Beim Anfügen mehrerer Geräte gleichzeitig wird der Default-Name immer um "-Nr." ergänzt (wobei Nr. = 1..n). Der Name der zugewiesen wurde (mit dem das Gerät auch im Baum auftaucht), ist auch gleichzeitig der "PROFINET Station Name", also der Name der mit dem im Device übereinstimmen muss. Eine Überprüfung des Gerätenamens kann durch das Scannen erfolgen.

Auf dem API (Application Profile Interface) können die Module angehängt werden. Auf Slot 0 steckt immer der DAP (DeviceAccessPoint), der bereits fixe Eigenschaften aus der GSDML mitbringt (z.B. Prozessdaten, Interface- und PortSubmodule...).

Dieses Modul ist immer da und kann nicht gelöscht oder verschoben werden. Jedes weitere Modul ist einem bestimmten API zugeordnet. Die Information, um welches es sich dabei handelt kommt aus der GSDML. Standardmäßig ist dies immer das API 0. Alternativ ist aber auch ein API für z.B. das PROFIDRIVE Profil oder aber ein FeldbusAPI denkbar. Durch Klicken im API auf "Append PROFINET Module..." wird ein Geräteverzeichnis geöffnet, aus welchem die entsprechenden Module gewählt und angefügt werden können. Wenn es die Module (in GSDML beschrieben) unterstützt, können wiederum an diesen auf dem gleichen Wege die Submodule angefügt werden.

## 5.2 Vergleichen von Soll- und Istbestückung

Bei einer bestehenden Verbindung kann in dem Reiter "Diagnosis" die Projektierung überprüft werden. "Real Identification Data" gibt auf dieser Ebene die wirklich vorhandenen Module innerhalb einer AR wieder, "Expected Identification Data" die erwarteten Module (also die im Controller projektierten) und "Module Difference" zeigt die vom Gerät festgestellten Unterschiede beim Soll-Ist-Vergleich auf.

ModuleInfo	SubModuleInfo	APINumber	SlotNumber	SubSlotNumber
<input checked="" type="checkbox"/> DAP Module	Interface 1	0x00000000	0	32768
<input checked="" type="checkbox"/> DAP Module	Port 1	0x00000000	0	32769
<input checked="" type="checkbox"/> DAP Module	Port 2	0x00000000	0	32770
<input checked="" type="checkbox"/> DAP Module	EK9300 V2.25, 2 Port	0x00000000	0	1
<input checked="" type="checkbox"/> EL1004	EL1004	0x00000000	1	1
<input checked="" type="checkbox"/> EL2008	EL2008	0x00000000	2	1
<input checked="" type="checkbox"/> EL3314	EL3314	0x00000000	3	1
<input checked="" type="checkbox"/> EL4004	EL4004	0x00000000	4	1
<input checked="" type="checkbox"/> EL2521	EL2521	0x00000000	5	1

At the display are the expected identification data from one AR.

Module Difference    Get Real Configuration    Diagnosis Data

Real Identification Data    API Number

Expected Identification Data

Auf dem "Diagnosis"- Reiter innerhalb des APIs, können Sie das entsprechende API auswählen, über welches Informationen bezogen werden sollen. Handelt es sich bei dem PROFINET Gerät z.B. um einen Antrieb, so unterstützt dieser in der Regel das Profidrive Profil, welches wiederum über API 0x3A00 gekennzeichnet ist. Sollen von diesem API z.B. die Real Identification Daten ausgelesen werden, erfolgt dieser Zugriff über das Profidrive Profil.

General
Diagnosis

Drive Type	ModuleInfo	SubModuleInfo	APINumber	SlotNumber	SubSlotNumber
▶▶ Sinamics_S	DO Control Unit	SIEMENS Telegramm 390, PZD-2/2	0x00003A00	1	3
▶▶ Servo	DO Servo	SIEMENS Telegramm 102, PZD-6/10	0x00003A00	2	3
▶▶ Servo	DO Servo	SIEMENS Telegramm 105, PZD-10/10	0x00003A00	3	3
▶▶ Servo	DO Servo	Standard Telegramm 3, PZD-5/9	0x00003A00	4	3
▶▶ Servo	DO Servo	Standard Telegramm 3, PZD-5/9	0x00003A00	5	3
▶▶ Vector	DO Vector	Standard Telegramm 1, PZD-2/2	0x00003A00	6	3
▶▶ Vector	DO Vector	Standard Telegramm 2, PZD-4/4	0x00003A00	7	3

At the display are the real identification data for API 0x00003A00

Module Difference

Get Real Configuration

Diagnosis Data

Real Identification Data

API Number

0x3a00 ▼

Expected Identification Data

Der Button "Get Real Configuration" wird außerdem innerhalb eines APIs aktiv (außer für Antriebe). Über diesen besteht die Möglichkeit, den eingelesenen Datensatz in das aktuelle Projekt zu übernehmen. Dabei ist zu beachten, dass bereits angelegte Module überschrieben werden. D.h. die Links gehen, auch bei zuvor richtig angelegt Modulen, verloren.

Beim Anzeigen der Modulunterschiede werden durch Markieren der Meldung zusätzliche Infos dargestellt.

General
Device
Diagnosis
Features
ADS

ModuleInfo	SubModuleInfo	APINumber	SlotNumber	SubSlotNumber	ModuleState
No Module	No SubModule	0x00000000	4	0	0x0000
No Module	No SubModule	0x00000000	5	0	0x0000

ModuleState:  
No Module - module not plugged

SubmoduleState:  
No submodule

API Number

### Diagnosis Data

Über den Button "Diagnosis Data" kann die vorliegende Diagnose ausgelesen werden. Auf Geräteebene werden hier alles vorliegenden Diagnosedaten für die bestehende AR ausgelesen.

The screenshot shows the 'Diagnosis' tab in the Beckhoff software. At the top, there are tabs for 'General', 'Device', 'Diagnosis', 'Features', and 'ADS'. Below the tabs is a table with columns 'API', 'Slot', 'SubSlot', and 'Diagnosis and Information'. Two rows are visible, both with a red error icon:

API	Slot	SubSlot	Diagnosis and Information
0x00000000	3	1	Info: Slot 3 (EL3314-0000-0017), EC Fehler: Status Aenderung Abgebrochen,
0x00000000	5	1	Info: Slot 5 (EL2521-0000-1017), EC Fehler: SDO Anfrage Abgebrochen,

Below the table is a scrollable text area containing detailed error information:

```

EC Fehler: Status Aenderung Abgebrochen -> Help: Ein interner Kommunikationsfehler ist am Profinet Geraet aufgetreten.
EC Aktueller Status: Init
EC Angeforderter Status: PreOP
EC AL Status Code: Kein Fehler.
EC VendorId: Beckhoff Automation GmbH
EC ProductCode: 0x0CF23052
EC RevisionNr: 0x00110000
EC SerialNr: 0x00000000
    
```

At the bottom of the interface, there are several buttons: 'Module Difference', 'Get Real Configuration', 'Diagnosis Data', 'Real Identification Data', 'Expected Identification Data', and an 'API Number' dropdown menu.

In der Liste werden höchstens zwei Diagnoseparameter angezeigt, weitere sind durch ein "..." gekennzeichnet. Beim Klicken auf die einzelne Meldung werden in dem unteren Fenster alle vorliegenden Diagnoseinformationen angezeigt.

**Zyklische Diagnose über "PnloBoxState" und "PnloBoxDiag"**

Diese Variablen werden zyklisch mit dem Prozessabbild zwischen PROFINET Treiber und System Manager ausgetauscht.

Aktuell werden folgende Fehlermeldungen über den PnloBoxState angezeigt.

Num-mer	Text	Beschreibung	Abhilfe / Grund
0	No error	kein Fehler	Kein Fehler
1	PROFINET Device state machine is in boot mode	PROFINET Device StateMachine ist noch in der Hochlauf Phase	Kein Fehler, warten
2	Device not found	Gerät antwortet nicht auf den Identify Request	Verbindung prüfen, Gerät angeschlossen, wurde das Gerät mit dem richtigen Namen benannt?
3	The stationname is not unique	Stationsname ist nicht eindeutig	Es gibt zwei oder mehr Geräte mit dem selben PROFINET Namen im Netzwerk. Eine korrekte Identifizierung kann nicht erfolgen.

Num-mer	Text	Beschreibung	Abhilfe / Grund
4	IP could not set	IP Adresse konnte nicht gesetzt werden.	Das PROFINET Gerät hat aus irgendwelchen Gründen das Setzen der IP settings abgelehnt. Prüfen ob die IP-Einstellungen korrekt sind.
5	IP conflict	Im Netzwerk trat ist ein IP-Konflikt aufgetreten.	Eine mögliche Ursache ist das mehrere Geräte die gleiche IP-Adresse haben.
6	DCP set was not successful	Auf einen DCP Set kam keine bzw. eine fehlerhafte Antwort.	Verbindung prüfen, Gerät angeschlossen, wurde das Gerät mit dem richtigen Namen benannt?
7	Watchdog error	Die Verbindung wurde mit einem Watchdog-Fehler abgebrochen.	Zykluszeit prüfen, Verbindung prüfen, ggf. Watchdog-Faktor erhöhen.
8	Datahold error	Die Verbindung wurde mit einem Datahold-Fehler abgebrochen.	Frame Datenstatus war für die Länge des DataHoldTimers ungültig. Evtl. Gerät neu starten.
9	RTC3: Sync signal could not started	Nur für IRT: Das Sync-Signal konnte nicht gestartet werden.	EtherCAT Sync Signal korrekt bzw. Sync0 gestartet?
10	PROFINET Controller has a link error	Der PROFINET Controller hat keinen Link.	Kabel und Verbindung überprüfen.
11	The aliasname is not unique	Der Aliasname ist nicht eindeutig	Es gibt zwei oder mehr Geräte mit demselben Alias-Namen im Netzwerk. Dieser setzt sich aus Nachbarschaftsinformationen zusammen (PortId.ChassisId). Eine korrekte Identifizierung kann nicht erfolgen.
12	The automatic name assignement isn't possible - wrong device type	Das automatische Setzen des Namens ist nicht möglich.	An der projektierten Position befindet sich nicht das erwartete PROFINET Gerät (VendorId oder DeviceId stimmen nicht überein). Somit ist kein automatisches Benennen und damit Geräteanlauf möglich.
31	only for EtherCAT gateways: WC-State of cyclic EtherCAT frame is 1	Nur für EL6631: EtherCAT WC State ist auf 1	Am EtherCAT Master + Slave den Mode checken (OP?).

Im "PnloBoxDiag" kann im Gegensatz zum State auch mehr als ein Zustand gleichzeitig angezeigt werden, d.h. das Ganze ist bitcodiert und es können bis zu 16 Infos angezeigt werden. Aktuell werden folgende Zustände dargestellt.

0x0000 = No diagnosis

Default. Device ist nicht in Kommunikation oder das TwinCAT Projekt wurde noch nicht aktiviert (allgemeiner TwinCAT Anwenderfehler).

0x0001 = IOC-AR is not established

Der Hochlauf der AR wurde gestartet, es wurde aber noch kein PrmEnd versendet. Der genaue Status ist über die DiagHistory ersichtlich.

0x0002 = IOC-AR is established

Kein Fehler, das Device hat die Verbindung aufgebaut und läuft fehlerfrei.

0x0004 = IOC-AR is established but no application ready

Kein Fehler, aber der Controller hat noch kein Application Ready bekommen. Die Dauer ist geräteabhängig, nach Profinet Spezifikation kann dies bis zu 300 s dauern.

0x0008 = IOC-AR is established but module difference

Die Verbindung wurde aufgebaut aber Soll- zu Ist-Konfiguration passt nicht. Überprüfen Sie die Konfigurationseinstellung und prüfen Sie über die Dialoge „Diagnostic“ welches Modul oder Submodul nicht passt.

0x0010 = At least one AlarmCR got a diagnosis alarm

Es wurde ein Alarm empfangen, dieser kann per ADS vom Treiber ausgelesen werden. Nach Auslesen sollte eine Quittierung über PnIoBoxCtrl 0x0001 erfolgen.

Die folgenden Flags beziehen sich auf den Datenstatus der zyklischen Kommunikation. Da theoretisch mehrere CRs zu einem Gerät aufgebaut sein könnten werden diese Flags gesetzt wenn mind. 1 CR ein Problem meldet.

InputCR ist die Eingangsrichtung, das sind die Frames die der Controller von den Devices empfängt.

OutputCR ist die Ausgangsrichtung, das sind die Frames die der Controller an die Devices versendet.

0x0100 = At least one InputCR is invalid

Die Daten der InputCR sind ungültig.

0x0200 = At least one InputCR Provider is in stop

Die Input CR ist im Stopp. Die Geräteapplikation wird in STOPP gehalten, die Ursache sollte auf Device Seite geklärt werden.

0x0400 = At least one InputCR Problemindicator is set

Ein Problem wurde im Device erkannt. Der Diagnosepuffer im Device enthält mind. eine Meldung. Diese können über die zugehörigen Records ausgelesen werden.

0x1000 = At least one OutputCR is invalid

Die Daten der OutputCR sind ungültig.

0x2000 = At least one OutputCR Provider is in stop

Die Output CR ist im Stopp. Dies deutet darauf hin das die triggernde IO Task oder PLC im Stopp ist .

0x4000 = At least one OutputCR Problemindicator is set

Ein Problem wurde im Controller erkannt. Der Diagnosepuffer im Controller enthält mind. eine Meldung. Diese werden z. B. über die DiagHistory angezeigt

Es werden hier zum einen Infos über den Zustand der IO Controller Single AR angezeigt. Außerdem werden aus den Frame-Datenstati der einzelnen CRs Sammelstati gebildet. Das Ganze passiert für die Input- und die Output-CRs (aktuell ist nur eine möglich, zukünftig wird der Controller mehrere CRs unterstützen). Außerdem wird im "PnIoBoxDiag" auch ein PROFINET Alarm angezeigt

## 5.3 Einstellungen

### 5.3.1 Projektierung des PROFINET Device

Bei einem PROFINET-Verbindungsaufbau vergibt der Controller dem Device immer eine IP-Adresse aus seinem eigenen Adressraum (wenn das Gerät noch keine bzw. eine andere hat). In TwinCAT wird per Default für ein Device immer die nächst höhere genommen (von der Controller Adapter-Klasse ausgehend), das Subnet und Gateway sind die Gleichen wie die des Controllers. Vor der eigentlichen IP-Vergabe vom Controller an das Device wird über einen ARP ein evtl. Adresskonflikt getestet bzw. überprüft, ob das Gerät bereits diese IP-Adresse hat. Tritt ein Konflikt auf, z.B. das die IP-Adresse im Netz bereits vergeben ist, stellt dies der IO-Treiber fest und gibt eine entsprechende Meldung im Logger Fenster aus. Erfolgt keine Antwort auf den ARP, nutzt kein Gerät (auch nicht das projektierte Device) diese IP-Konfiguration, was wiederum zur Folge hat, dass der Controller dem Device über einen DCP\_SET die IP-Einstellungen zuweist. Wurde über den ARP festgestellt, dass das gesuchte Gerät bereits die zu projektierende IP Adresse hat, wird das Setzen übersprungen.

General	Device	Diagnosis	Features	ADS	EL663x	Synchronization
<b>Adapter Properties</b>						
Stationname <input type="text" value="sinamics2"/>						
VendorId		DeviceId		HW Version		SW Version
<input type="text" value="0x002A"/>		<input type="text" value="0x0501"/>		<input type="text" value="2.00"/>		<input type="text" value="V 4.30.28"/>
<b>IP configuration</b>						
IP address		<input type="text" value="192 . 168 . 1 . 5"/>				
Subnet		<input type="text" value="255 . 255 . 255 . 0"/>				
Gateway		<input type="text" value="192 . 168 . 1 . 1"/>				
<b>Instance Properties</b>						
ID		FrameId				
<input type="text" value="0x0000"/>		<input type="text" value="Fast Config"/>				
MaxLengthIn		MaxLengthOut		ActLengthIn		ActLengthOut
<input type="text" value="400 Byte"/>		<input type="text" value="400 Byte"/>		<input type="text" value="126 Byte"/>		<input type="text" value="102 Byte"/>

In diesem Fenster kann außerdem die "InstanceID" und die "FrameID" geändert werden. Die Defaulteinstellungen sind jedoch für die meisten Anwendungen ausreichend. Die Instance ID fließt mit in die Bildung der Objekt UUID ein. Eine Änderung sollte also nur in Ausnahmefällen durchgeführt werden. Bei einer Änderung der Frame ID ist die genutzte RTClass zu berücksichtigen (z.B. für RTClass1 unicast 0xC000 - 0xFAFF). Befindet sich das Gerät an einem IRT Controller und es wurden automatisch alle Geräte nach RTClass3 geschaltet, wird die Frame ID automatisch verwaltet und es besteht keine Eingabemöglichkeit (wird durch "Fast Config" gekennzeichnet).

In diesem Menü kann außerdem die aktuelle Prozessdatenlänge überprüft werden. D.h. die MaxLängen geben an, welche Prozessdatengröße von dem entsprechenden Gerät unterstützt wird, die ActLängen bezeichnen die aktuelle Prozessdatenlänge (incl. IOPS und IOCS). Werden beim Anfügen weiterer Module / Submodule die Maximallängen überschritten, erscheint die entsprechende Fehlermeldung.

Unter dem Reiter "Features" können verschiedene Einstellungen bzgl. Zykluszeit vorgenommen werden. Die Zykluszeit des Controller muss immer für RTClass1 einer zweier Potenz, bei 1 ms beginnend, entsprechen ( 1, 2, 4, 8...). Wurde eine falsche Basiszeit gewählt, wird dies über eine entsprechende Meldung angezeigt. Für RTClass3 kann die 1 ms Basiszeit immer wieder durch zwei geteilt werden (bis min. 31,25 µs). Die Device-Zykluszeit kann über den Exponenten verändert werden. Das Minimum ist dabei immer die Controller CycleTime, es sei denn, in der GSDML ist als minimale Zykluszeit eine größere als die des Controllers definiert. Das Maximum beträgt für RTClass1 512 ms. Der "SendClockFactor" steht hier als Zeitbasis fest auf dem Wert 32 (31,25 µs \* 32 = 1 ms). Darauf bezieht sich auch der "ReductionRatioFactor", d.h. ein RRFactor von 4 bedeutet eine Zykluszeit von 4 ms. Über die Phase kann wieder innerhalb eines Zyklus der Sendezeitpunkt verschoben werden, d.h. bei RR = 4 kann die Phase 1 - 4 betragen. Dieser Wert ist aber erst bei einer synchronisierten Übertragung von Bedeutung.

General	Device	Diagnosis	Features	ADS	EL663x	Synchronization
---------	--------	-----------	----------	-----	--------	-----------------

**IO Cyclic Data**

Controller Cycle Time:   Cycle time from master task

Device Cycle Time:   $DevCycleTime = SendClockFactor * 31.25\mu s * RedRatio$

Min Device Interval:

Send Clock Factor:

Reduction Ratio:

Phase:

Watchdog Factor:  Default = 3

Watchdog Time:   $Watchdog\ Time = Watchdog\ Factor * DevCycleTime$

**Comment**

Außerdem besteht hier die Möglichkeit den PROFINET Watchdog-Faktor zu verstellen. D.h. jedes Gerät überwacht anhand dieses Faktors den Eingang der zyklischen Daten. Steht der Faktor auf dem Default-Wert (3) bedeutet das, dass bei einer RR von 4 drei Zyklen 12 ms benötigen. Somit reagiert ein Gerät nach 12 ms auf fehlende Telegramme (z.B. mit einem Alarm und / oder Abbau der AR). Die Grenzen und Werte werden bei Verstellen der einzelnen Faktoren immer wieder neu berechnet.

### 5.3.2 BK9xx3

Beim Beckhoff K-Bus Koppler (derzeit BK9103 oder BK9053), der nicht mit einer EL663x verbunden ist, erscheint hier ein zusätzliches Menü.

General	Device	Diagnosis	Features	ADS	BK9xx3
---------	--------	-----------	----------	-----	--------

PLC Access

**K-Bus**

K-Bus Counter

K-Bus CycleTime (100us)

Error Code:

Error Argument:

K-Bus stop if Profinet error

Hierüber kann leicht auf die zyklischen Prozessdaten im DAP des BKs zugegriffen werden.

Außerdem kann über dieses Menü ein Firmware-Update aus dem System Manager auf dem Buskoppler durchgeführt werden. Erfolgt das Update über IP ist darauf zu achten, dass die IP-Adresse über die DIP-Schalter bezogen wird. Ist dies nicht der Fall, bricht während des Updates die Verbindung ab, da auch der Speicherbereich der IP-Einstellungen formatiert und neu beschrieben wird.

### 5.3.3 EL663x

Wird das Controller Protokoll über eine EL663x betrieben, so erscheint an den Geräten ein zusätzliches Menü.

The screenshot shows the configuration interface for the EL663x controller. The 'EL663x' tab is selected, and the 'Synchronization' sub-tab is active. Under 'EtherCAT mapping properties', there are three unchecked checkboxes: 'alternative mapping model', 'get PN-Stationname from ECAT', and 'get PN-IP-Settings from ECAT'. Below this, the 'PDO mapping' section contains four radio button options: 'Submodule data (0x6nn0, 0x7nn0)' (selected), 'Module data (0x6nn1, 0x7nn1)', 'Submodule data and IOPS (0x6nn2, 0x7nn2)', and 'Module data and IOPS (0x6nn3, 0x7nn3)'.

Für den Controller ist derzeit nur die Wahl des PDO mappings wählbar. D.h. hierüber wird eingestellt, in welcher Form die PROFINET Prozessdaten auf die EtherCAT-seitigen PDOs abgebildet werden.

### 5.3.4 IRT Controller

Wird das Gerät an einem IRT fähigem Controller betrieben, so erscheint ein zusätzliches Menü.

The screenshot shows the configuration interface for the IRT Controller. The 'EL663x' tab is selected, and the 'Synchronization' sub-tab is active. The configuration is organized into a table with columns for 'Factor', 'Basetime', and 'Time'. The 'Time' column contains input fields for the resulting time values.

	Factor	Basetime	Time
Time Ti:	3	125.000 us	375.000 us
Time To:	2	125.000 us	250.000 us
Time Input Valid:			0.000 us
Time Output Valid:			28.360 us

Hierüber besteht die Möglichkeit, die Ti und To - Faktoren für IRT-fähige Geräte vorzugeben. Damit ist die Zeit gemeint, wann die Daten im Gerät innerhalb eines Taktes gültig sind bzw. auf gültig gesetzt werden sollen. Voraussetzung ist, dass dieses Feature auch unterstützt wird. Die Information hierüber liefert die GSMDL. Es gibt hierbei immer einen Grundtakt (Basetime). Über die GSMDL kommt anhand eines minimalen Faktors die Aussage über die minimal mögliche Zeit. Die obere Grenze des Faktors wird durch die verwendete Zykluszeit begrenzt. Die über PROFINET mögliche kleinste Zeit, bei der Daten gültig sein könnten (immer bezogen auf den Takt) wird über die Parameter "Time Input Valid" bzw. "Time Output Valid" angezeigt.

### 5.3.5 Shared Device

Das SharedDevice Feature ist ab TwinCAT 2 Build 22.50 oder TwinCAT 3 Build 4019 verfügbar.

Der Dialog erscheint wenn das Gerät "SharedDevice" unterstützt. Die Information hierzu kommt aus der GSDML.

General Device Diagnosis Features ADS Shared Device					
Name	Slot	Subslot	Access	SharedInput	
Term 1 (DAP Module)					
Subterm 1 (EK9300 V 2.31 (at least FW 2.00))	0	1	true	has output data	
Subterm 2 (Interface)	0	32768	false	no access	
Subterm 3 (Port 1)	0	32769	false	no access	
Subterm 4 (Port 2)	0	32770	false	no access	
Term 2 (EL1018)					
Subterm 1 (EL1018)	1	1	true	true	
Term 3 (EL2008)					
Subterm 1 (EL2008)	2	1	false	no access	
Term 4 (EK1110)					
Subterm 1 (EK1110)	3	1	false	no access	
Term 5 (EK1100)					
Subterm 1 (EK1100)	4	1	true	no input data	
Term 6 (EL3004)					
Subterm 1 (ModuleAccessPoint)	5	1	true	no input data	
Subterm 2 (Standard)	5	2	true	false	
Term 7 (EL4012)					
Subterm 1 (EL4012)	6	1	true	has output data	

Hierüber besteht die Möglichkeit dem Controller Zugriff auf die einzelnen Submodule zu Erteilen oder auch zu Entziehen. Defaultmäßig hat der Controller Zugriff auf alle Submodule, falls SharedInput unterstützt wird ist dieses ausgeschaltet.

Die Textmeldungen für SharedInput haben die folgende Bedeutung:

- "not supported" - SharedInput wird vom Gerät nicht unterstützt (Info aus der GSDML)
- "has output data" - das Submodul hat Ausgänge - Aktivierung von SharedInput nicht möglich
- "no Input data" - das Submodul hat keine Eingänge (und auch keine Ausgänge)
- "no access" - Zugriff ist gesperrt
- "true" bzw. "false" - Eingestellter Wert für SharedInput

Durch Doppelklick auf die einzelnen Submodule können die Einstellungen geändert werden. Wird der Zugriff an einem Port- oder Interfacesubmodul geändert so wird dieser für alle Ports bzw. Interfaces geändert.

## 5.4 Module

### 5.4.1 Diagnose auf Modul Ebene

Die Slotnummer der Module entspricht immer der Position im Baum. D.h. das DAP Modul beginnt immer mit Slotnummer 0, danach geht es der Reihe nach weiter. Auf Modul Ebene besteht im Diagnosereiter die Möglichkeit, die Soll- und Istdaten für das entsprechende Modul zu vergleichen. Außerdem kann die vorliegende Diagnose für das Modul ausgelesen werden.

## 5.4.2 Submodule

### 5.4.2.1 Diagnose auf Submodul-Ebene

Bei den Submodulen werden nach PROFINET aktuell zwischen 4 Arten von Submodulen unterschieden.

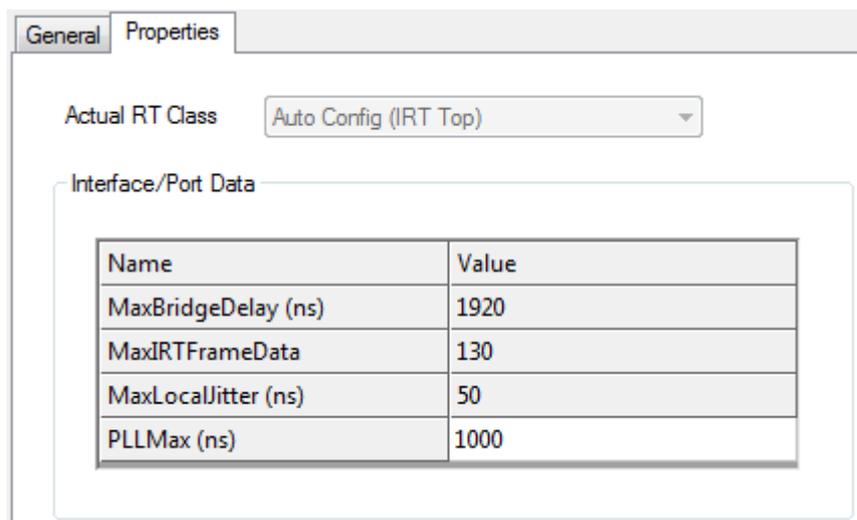
- Virtuelle Submodule:  
Die virtuellen Submodule sind immer fest mit einem Modul verbunden. D.h. beim Einfügen eines Moduls werden auch immer die mitdefinierten virtuellen Submodule auf dem vorgegebenen Subslot eingefügt. Diese Art der Submodule ist die derzeit gängige Methode.
- Reelle Submodule:  
Hier besteht die Möglichkeit, aus einer Submodul-Liste die steckbaren Submodule auszuwählen und an das Modul anzufügen. Die nötigen Informationen werden aus der GSDML bezogen. In TwinCAT kann mir der rechte Maustaste ein Modul aus einer solchen Liste ausgewählt werden (vorausgesetzt, dies wird vom Gerät unterstützt).
- Port Submodule:  
In einem solchen Submodul sind die physikalischen Eigenschaften eines Netzwerkports wiedergegeben.
- Interface Submodule:  
In den Interface Submodulen sind gerätespezifische Eigenschaften definiert. Dies können z.B. zusätzlich unterstützte Protokolle sein, Timing Eigenschaften, unterstützte MIBs, usw.

Generell haben die Submodule die gleichen Diagnoseeigenschaften wie die Module, d.h. auch hier besteht aktuell nur die Möglichkeit, in TwinCAT die Soll- und die Istkonfiguration auszulesen. Die Reihenfolge der Subslotnummern ist nicht zwangsläufig die der Reihenfolge im TwinCAT-Projekt. So wird z.B. im DAP immer mit dem Interface Submodule (ISM) begonnen, die Subslotnummer des ISM ist jedoch in der GSDML definiert und beginnt bei 0x8000. Es gibt 16 mögliche Interfaces (0x8x00) mit jeweils bis zu 256 Ports (0x80xx). Nach einem ISM folgen die Port Submodule mit zuvor erwähnter Subslotnummer.

### 5.4.2.2 Interface Submodul

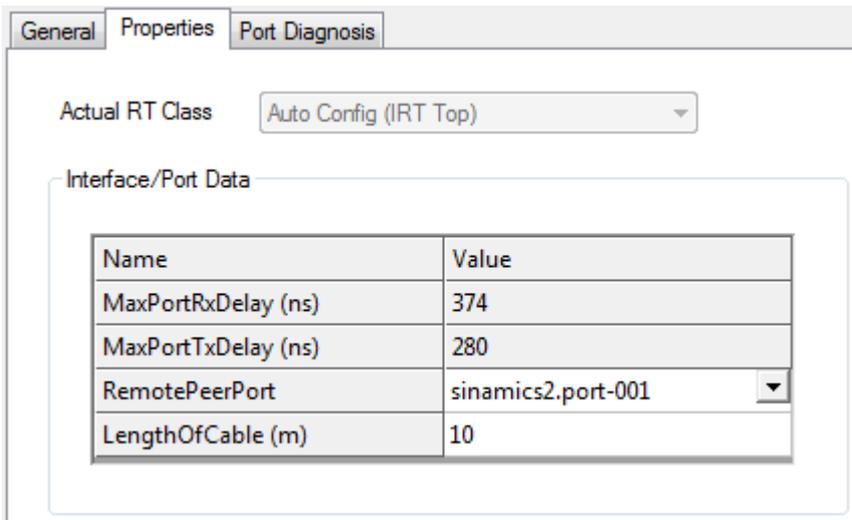
An dem Interface Submodul kann immer die Art der Kommunikation eingestellt werden (derzeit RTClass1 oder RTClass3). Einzige Ausnahme ist der Fall, dass über das Menü "Auto Config..." eine generell gültige RTClass eingestellt wurde.

Wird über RTClass3 kommuniziert, so kann am Interface zusätzlich das PLL Window eingestellt werden.

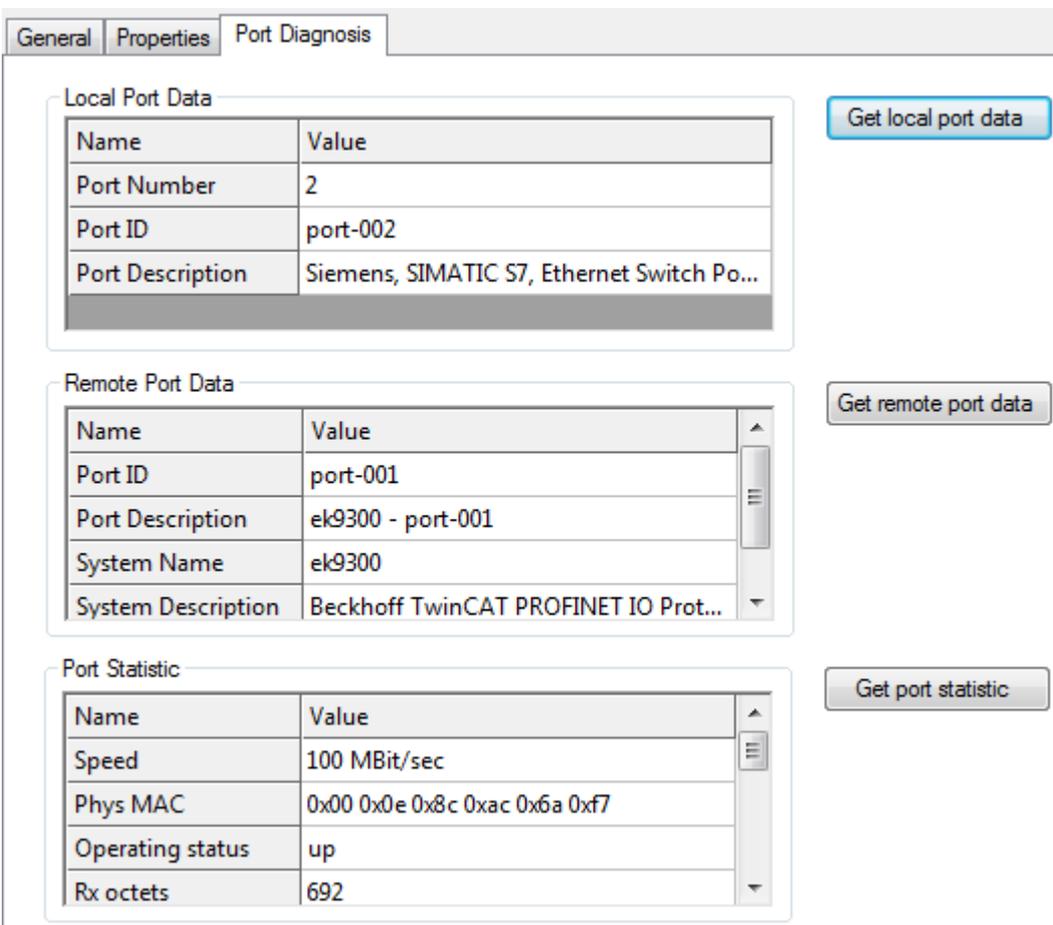


### 5.4.2.3 Port Submodul

Im Reiter "Properties" können portspezifische Einstellungen vorgenommen werden. Das Menü der möglichen Einstellungen ist immer abhängig der verwendeten RTClass.



Außerdem können einige Porteigenschaften ausgelesen werden.



Die Informationen sind hier unterteilt in lokale Portinfos und in Remote-Porteigenschaften. D.h. in PROFINET ist ab Conformance Class A (CCA) das LLDP Protokoll (IEEE Std 802.1AB) vorgeschrieben. Über dieses Protokoll tauschen die Teilnehmer Nachbarschaftskennungen aus, so dass jedem Port sein Nachbar bekannt ist. Weiterhin kann an dieser Stelle das Simple Network Management Protocol (SNMP) zur Hilfe genommen werden. TwinCAT agiert beim Öffnen des Reiters "Port Diagnosis" als Network Management Station (NMS) und sammelt über SNMP die benötigten Informationen des Teilnehmers ein. In vorherigem Bild erkennt man z.B. das der Lokale Port 1 des BK9053 mit dem Port 2 des BK9103 verbunden ist. Für eine korrekte Topologie-Erkennung ist es wichtig, dass nur Teilnehmer im Strang vorhanden sind, die auch das LLDP Protokoll unterstützen (dies gilt auch für Switches!).

### 5.4.2.4 Reelle / Virtuelle Submodule

Haben diese Submodule Parametrierdaten, so werden diese wie im folgenden Bild angezeigt.

The screenshot shows a software window titled 'Parameterize Module' with a 'General' tab. On the left, a tree view shows 'Allgemeine Parameter' expanded to 'Index 0x1'. The main area contains a table with the following data:

Name	R/W	Offline Value	Online Value
Sammeldiagnose	R/W	0	
Diag: Über-/Unterlauf	R/W	0	
Diag: Drahtbruch E-Kanal 0	R/W	0	
Diag: Drahtbruch E-Kanal 1	R/W	0	
Glättung E-Kanal 0	R/W	keine	
Glättung E-Kanal 1	R/W	keine	
Messart/-bereich, E-Kanal 0	R/W	Spannung +/- 10 V	
Messart/-bereich, E-Kanal 1	R/W	Spannung +/- 10 V	
Prozessalarm bei Grenzwertüberschre...	R/W	0	
Oberer Grenzwert E-Kanal 0	R/W	32511	
Unterer Grenzwert E-Kanal 0	R/W	33024	
Prozessalarm bei Grenzwertüberschre...	R/W	0	
Oberer Grenzwert E-Kanal 1	R/W	32511	
Unterer Grenzwert E-Kanal 1	R/W	33024	

Below the table are three buttons: 'Read', 'Write', and 'Set to Default'. To the right of these buttons is a text box with the following text:

Klicken Sie auf das Feld, um sich die verfügbaren Meßarten und Meßbereiche anzeigen zu lassen und auszuwählen. Empfehlung: Nichtbeschaltete Eingangskanäle sollten Sie deaktivieren (Eingabeart: deaktiviert), um die Baugruppenzykluszeit zu verkürzen.

Hier kann zwischen den einzelnen Indizes gewählt werden. Die Daten können je nach Zugriffsart gelesen und/oder geschrieben werden. Beim Rücklesen werden die Online-Werte aktualisiert. Ist ein einzelner Index markiert, so wird beim Klicken auf "Set to default" alle Werte innerhalb eines Index auf Default gesetzt, beim Markieren einzelner Werte werden nur diese zurückgesetzt. Eine Änderung der schreibbaren Werte erfolgt durch Doppelklick auf die entsprechende Zeile.

## 6 TwinCAT Bibliothek und Programmierung

Für den Einsatz des Profinet Controllers gibt es fertige Funktionsbausteine. In der Bibliothek sind weiterhin Bausteine für die EL6631-0010 PROFINET Device Klemme enthalten, welche aber nicht Bestandteil dieser Dokumentation sind..

Download: <https://infosys.beckhoff.com/content/1031/tcprofinetcontroller/Resources/11971387659.zip> 

### I&M Funktionen

Bausteine I&M Funktionen	Bedeutung	Beschreibung
<a href="#">FB_PN_IM0_READ [▶ 55]</a>	Lesen der I&M Funktion 0	Supplement und EL663x
<a href="#">FB_PN_IM1_READ [▶ 56]</a>	Lesen der I&M Funktion 1	Supplement und EL663x
<a href="#">FB_PN_IM2_READ [▶ 58]</a>	Lesen der I&M Funktion 2	Supplement und EL663x
<a href="#">FB_PN_IM3_READ [▶ 60]</a>	Lesen der I&M Funktion 3	Supplement und EL663x
<a href="#">FB_PN_IM4_READ [▶ 63]</a>	Lesen der I&M Funktion 4	Supplement und EL663x
<a href="#">FB_PN_IM1_WRITE [▶ 57]</a>	Schreiben der I&M Funktion 1	Supplement und EL663x
<a href="#">FB_PN_IM2_WRITE [▶ 59]</a>	Schreiben der I&M Funktion 2	Supplement und EL663x
<a href="#">FB_PN_IM3_WRITE [▶ 61]</a>	Schreiben der I&M Funktion 3	Supplement und EL663x
<a href="#">FB_PN_IM4_WRITE [▶ 63]</a>	Schreiben der I&M Funktion 4	Supplement und EL663x

### Statistik und Diagnose Informationen

Bausteine I&M Funktionen	Bedeutung	Beschreibung
<a href="#">FB_PN_GET_PORT_STATISTIC [▶ 64]</a>	Lesen der Port Statistik	Supplement und EL663x
<a href="#">FB_PN_READ_PORT_DIAG [▶ 65]</a>	Lesen der Port Diagnose	Supplement und EL663x

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS-Bibliotheken
TwinCAT v2.11.0 R3	PC or CX (x86, ARM)	TcEtherCAT.lib TcPlcIoFunction.lib TcUtilities.lib TcSystem.lib TcBase.lib

## 6.1 Funktionsbausteine

### 6.1.1 FB\_PN\_IM0\_READ



Mit diesem Funktionsbaustein liest der PROFINET-Controller alle I&M 0 ( Identification & Maintenance ) Daten aus einem über den Eingang *Port* referenzierten Gerät.

Der Frameaufbau der I&M0 Funktion entspricht dem Index [0xAFF0](#) [► 67] nach PROFINET Standard.

### VAR\_INPUT

```
VAR_INPUT
  bStart   : BOOL; NETID   : T_AmsNetId; (* AMS Net ID from Controller *)
  PORT     : T_AmsPort; (* Port used by Controller to communicate with Device *)
END_VAR
```

**bStart** : Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert

**NETID**: AMS Net ID des Controllers

**PORT**: Port, über den der Controller mit dem Gerät kommuniziert ( Port = Device ID + 1000hex )

### VAR\_OUTPUT

```
VAR_OUTPUT
  bBusy    : BOOL;
  IM_AFF0  : str_IM_0xAFF0;
  bError   : BOOL;
  iErrorID : UDINT;
END_VAR
```

**bBusy** : Bei der Aktivierung des Funktionsbausteins wird dieser Ausgang gesetzt und bleibt gesetzt, bis eine Rückmeldung erfolgt. Während *Busy* = TRUE wird an den Eingängen kein neuer Befehl angenommen.

**IM\_AFF0**: Ausgabe des vom Gerät gelieferten I&M0 Frames in einer Struktur. *str\_IM\_0xAFF0*.

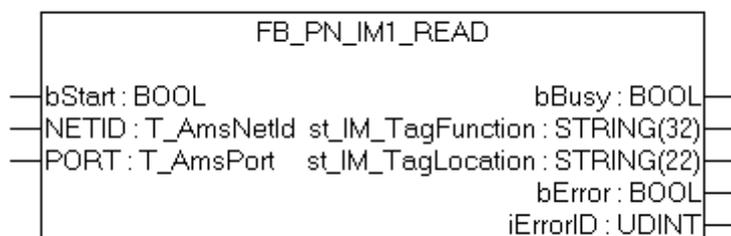
**bError**: Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der *bBusy*-Ausgang zurückgesetzt wurde.

**iErrorID**: Liefert bei einem gesetzten *bError*-Ausgang eine ADS Fehlernummer.

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v2.11.0 R3	PC or CX (x86, ARM)	TcProfinetDiag.Lib

## 6.1.2 FB\_PN\_IM1\_READ



Mit diesem Funktionsbaustein liest der PROFINET-Controller alle Daten I&M1 ( Information & Maintenance ) Daten aus einem über den Eingang *Port* referenzierten Gerät.

Der Frameaufbau der I&M1 Funktion entspricht dem Index [0xAFF1](#) [► 68] nach PROFINET Standard.

### VAR\_INPUT

```
VAR_INPUT
  bStart   : BOOL; NETID   : T_AmsNetId; (* AMS Net ID from Controller *)
  PORT     : T_AmsPort; (* Port used by Controller to communicate with Device *)
END_VAR
```

**bStart** : Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert

**NETID:** AMS Net ID des Controllers

**PORT:** Port, über den der Controller mit dem Gerät kommuniziert ( Port = Device ID + 1000hex )

**VAR\_OUTPUT**

```
VAR_OUTPUT
  bBusy          : BOOL;
  st_IM_TagFunction : STRING(32);
  st_IM_TagLocation : STRING(22);
  bError         : BOOL;
  iErrorID       : UDINT;
END_VAR
```

**bBusy** : Bei der Aktivierung des Funktionsbausteins wird dieser Ausgang gesetzt und bleibt gesetzt, bis eine Rückmeldung erfolgt. Während Busy = TRUE wird an den Eingängen kein neuer Befehl angenommen.

**st\_IM\_TagFunction** : ausgelesenes Label für Function des Gerätes.

**st\_IM\_TagLocation** : ausgelesenes Label für Einbauort des Gerätes.

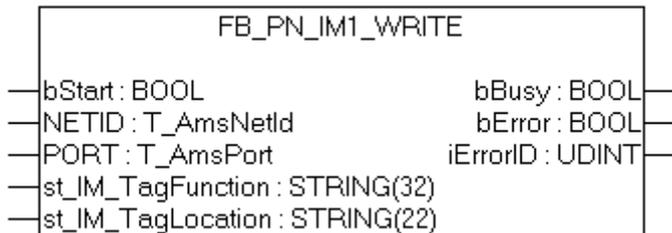
**bError**: Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der *bBusy*-Ausgang zurückgesetzt wurde.

**iErrorID**: Liefert bei einem gesetzten *bError*-Ausgang eine ADS Fehlernummer.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v2.11.0 R3	PC or CX (x86, ARM)	TcProfinetDiag.Lib

**6.1.3 FB\_PN\_IM1\_WRITE**



Mit diesem Funktionsbaustein schreibt der PROFINET-Controller alle I&M1 ( Identification & Maintenance ) Daten auf ein über den Eingang *Port* referenziertes Gerät.

Der Frameaufbau der I&M1 Funktion entspricht dem Index 0xAFF1 [▶ 68] nach PROFINET Standard.

**VAR\_INPUT**

```
VAR_INPUT
  bStart          : BOOL;
  NETID           : T_AmsNetId; (* AMS Net ID from Controller *)
  PORT            : T_AmsPort; (* Port used by Controller to communicate with Device *)
  st_IM_TagFunction : STRING(32);
  st_IM_TagLocation : STRING(22);
END_VAR
```

**bStart** : Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert

**NETID** : AMS Net ID des Controllers

**PORT** : Port, über den der Controller mit dem Gerät kommuniziert ( Port = Device ID + 1000hex )

**st\_IM\_TagFunction** : Mit diesem String wird die Funktionsbeschreibung auf dem Gerät abgelegt.

**st\_IM\_TagLocation** : Mit diesem String wird der Einbauort auf dem Gerät abgelegt.

**VAR\_OUTPUT**

```
VAR_OUTPUT
  bBusy      : BOOL;
  bError     : BOOL;
  iErrorID   : UDINT;
END_VAR
```

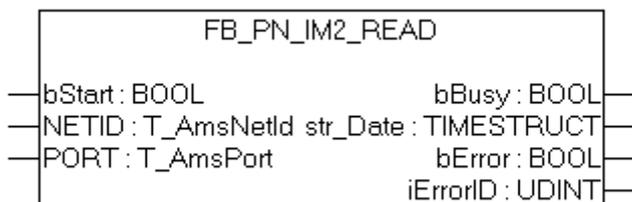
**bBusy** : Bei der Aktivierung des Funktionsbausteins wird dieser Ausgang gesetzt und bleibt gesetzt, bis eine Rückmeldung erfolgt. Während `Busy = TRUE` wird an den Eingängen kein neuer Befehl angenommen.

**bError** : Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der `bBusy`-Ausgang zurückgesetzt wurde.

**iErrorID**: Liefert bei einem gesetzten `bError`-Ausgang eine ADS Fehlernummer..

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v2.11.0 R3	PC or CX (x86, ARM)	TcProfinetDiag.Lib

**6.1.4 FB\_PN\_IM2\_READ**

Mit diesem Funktionsbaustein liest der PROFINET-Controller alle I&M 2 ( Identification & Maintenance ) Daten aus einem über den Eingang *Port* referenzierten Gerät.

Der Frameaufbau der I&M2 Funktion entspricht dem Index [0xAFF2 \[► 68\]](#) nach PROFINET Standard.

**VAR\_INPUT**

```
VAR_INPUT
  bStart : BOOL; NETID : T_AmsNetId; (* AMS Net ID from Controller *)
  PORT   : T_AmsPort; (* Port used by Controller to communicate with Device *)
END_VAR
```

**bStart** : Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert

**NETID** : AMS Net ID des Controllers

**PORT** : Port, über den der Controller mit dem Gerät kommuniziert ( `Port = Device ID + 1000hex` )

**VAR\_OUTPUT**

```
VAR_OUTPUT
  bBusy      : BOOL;
  str_Date   : TIMESTRUCT; (*YYYY-MM-DD HH:MM*)
  bError     : BOOL;
  iErrorID   : UDINT;
END_VAR
```

**bBusy** : Bei der Aktivierung des Funktionsbausteins wird dieser Ausgang gesetzt und bleibt gesetzt, bis eine Rückmeldung erfolgt. Während `Busy = TRUE` wird an den Eingängen kein neuer Befehl angenommen.

**str\_Date** : Liefert das Datum des Geräteeinbaus in dem Format `< YYYY-MM-DD HH:MM >` zurück.

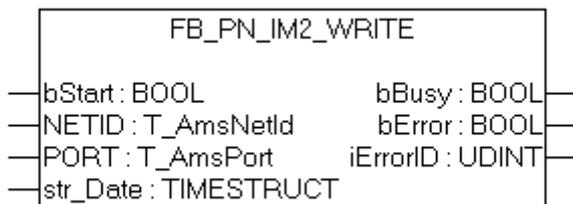
**bError** : Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der *bBusy*-Ausgang zurückgesetzt wurde.

**iErrorID**: Liefert bei einem gesetzten *bError*-Ausgang eine ADS Fehlernummer.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v2.11.0 R3	PC or CX (x86, ARM)	TcProfinetDiag.Lib

**6.1.5 FB\_PN\_IM2\_WRITE**



Mit diesem Funktionsbaustein schreibt der PROFINET-Controller alle I&M 2 ( Identification & Maintenance ) Daten auf ein über den Eingang *Port* referenziertes Gerät.

Der Frameaufbau der I&M2 Funktion entspricht dem Index 0xAFF2 [► 68] nach PROFINET Standard.

**VAR\_INPUT**

```
VAR_INPUT
  bStart      : BOOL;
  NETID       : T_AmsNetId; (* AMS Net ID from Controller *)
  PORT        : T_AmsPort; (* Port used by Controller to communicate with Device *)
  str_Date    : TIMESTRUCT; (*YYYY-MM-DD HH:MM*)
END_VAR
```

**bStart** : Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert

**NETID** : AMS Net ID des Controllers

**PORT** : Port, über den der Controller mit dem Gerät kommuniziert ( Port = Device ID + 1000hex )

**str\_Date** : Schreibt ein Datum ( z.B. Einbaudatum des Gerätes ) in dem Format < YYYY-MM-DD HH:MM > auf das Gerät.

**VAR\_OUTPUT**

```
VAR_OUTPUT
  bBusy       : BOOL;
  bError      : BOOL;
  iErrorID    : UDINT;
END_VAR
```

**bBusy** : Bei der Aktivierung des Funktionsbausteins wird dieser Ausgang gesetzt und bleibt gesetzt, bis eine Rückmeldung erfolgt. Während *Busy* = TRUE wird an den Eingängen kein neuer Befehl angenommen.

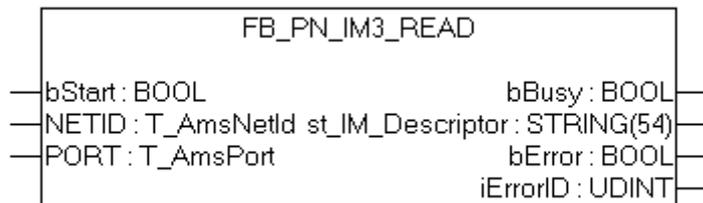
**bError** : Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der *bBusy*-Ausgang zurückgesetzt wurde.

**iErrorID**: Liefert bei einem gesetzten *bError*-Ausgang eine ADS Fehlernummer.

**Voraussetzungen**

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v2.11.0 R3	PC or CX (x86, ARM)	TcProfinetDiag.Lib

## 6.1.6 FB\_PN\_IM3\_READ



Mit diesem Funktionsbaustein liest der PROFINET-Controller alle I&M3 ( Identification & Maintenance ) Daten aus einem über den Eingang *Port* referenziertes Gerät.

Der Frameaufbau der I&M3 Funktion entspricht dem Index [0xAFF3](#) [[▶ 68](#)] nach PROFINET Standard.

### VAR\_INPUT

```
VAR_INPUT
  bStart   : BOOL; NETID   : T_AmsNetId; (* AMS Net ID from Controller *)
  PORT     : T_AmsPort; (* Port used by Controller to communicate with Device *)
END_VAR
```

**bStart** : Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert

**NETID** : AMS Net ID des Controllers

**PORT** : Port, über den der Controller mit dem Gerät kommuniziert ( Port = Device ID + 1000hex )

### VAR\_OUTPUT

```
VAR_OUTPUT
  bBusy      : BOOL;
  st_IM_Descriptor : STRING(54);
  bError     : BOOL;
  iErrorID   : UDINT;
END_VAR
```

**bBusy** : Bei der Aktivierung des Funktionsbausteins wird dieser Ausgang gesetzt und bleibt gesetzt, bis eine Rückmeldung erfolgt. Während Busy = TRUE wird an den Eingängen kein neuer Befehl angenommen.

**st\_IM\_Descriptor** : Liefert die für das Gerät hinterlegte Hersteller Beschreibung zurück.

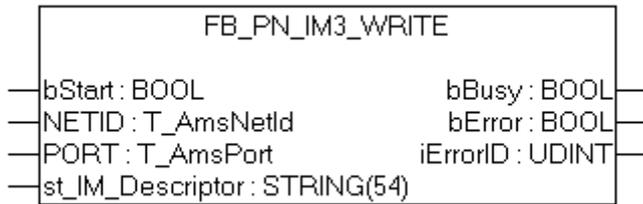
**bError** : Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der *bBusy*-Ausgang zurückgesetzt wurde.

**iErrorID**: Liefert bei einem gesetzten *bError*-Ausgang eine ADS Fehlernummer.

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v2.11.0 R3	PC or CX (x86, ARM)	TcProfinetDiag.Lib

## 6.1.7 FB\_PN\_IM3\_WRITE



Mit diesem Funktionsbaustein schreibt der PROFINET-Controller alle I&M3 ( Identification & Maintenance ) Daten auf ein über den Eingang *Port* referenziertes Gerät.

Der Frameaufbau der I&M3 Funktion entspricht dem Index [0xAFF3 \[► 68\]](#) nach PROFINET Standard.

### VAR\_INPUT

```
VAR_INPUT
  bStart      : BOOL; NETID   : T_AmsNetId; (* AMS Net ID from Controller *)
  PORT       : T_AmsPort; (* Port used by Controller to communicate with Device *)
  st_IM_Descriptor : STRING(54);
END_VAR
```

**bStart** : Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert

**NETID** : AMS Net ID des Controllers

**PORT** : Port, über den der Controller mit dem Gerät kommuniziert ( Port = Device ID + 1000hex )

**st\_IM\_Descriptor** : Liefert die für das Gerät hinterlegte Hersteller Beschreibung zurück.

### VAR\_OUTPUT

```
VAR_OUTPUT
  bBusy      : BOOL;
  bError     : BOOL;
  iErrorID   : UDINT;
END_VAR
```

**bBusy** : Bei der Aktivierung des Funktionsbausteins wird dieser Ausgang gesetzt und bleibt gesetzt, bis eine Rückmeldung erfolgt. Während Busy = TRUE wird an den Eingängen kein neuer Befehl angenommen.

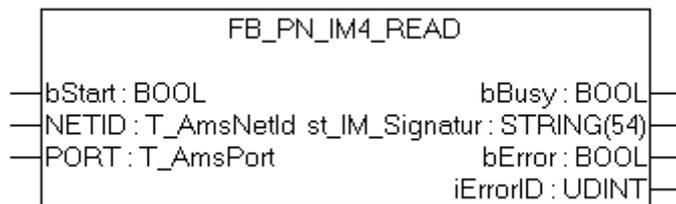
**bError** : Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der *bBusy*-Ausgang zurückgesetzt wurde.

**iErrorID**: Liefert bei einem gesetzten *bError*-Ausgang eine ADS Fehlernummer.

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v2.11.0 R3	PC or CX (x86, ARM)	TcProfinetDiag.Lib

## 6.1.8 FB\_PN\_IM4\_Read



Mit diesem Funktionsbaustein liest der PROFINET-Controller alle I&M4 ( Identification & Maintenance ) Daten aus einem über den Eingang *Port* referenziertes Geräte.

Der Frameaufbau der I&M4 Funktion entspricht dem Index [0xAFF4 \[►\\_68\]](#) nach PROFINET Standard.

### VAR\_INPUT

```
VAR_INPUT
  bStart   : BOOL; NETID   : T_AmsNetId; (* AMS Net ID from Controller *)
  PORT     : T_AmsPort; (* Port used by Controller to communicate with Device *)
END_VAR
```

**bStart** : Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert

**NETID** : AMS Net ID des Controllers

**PORT** : Port, über den der Controller mit dem Gerät kommuniziert ( Port = Device ID + 1000hex )

### VAR\_OUTPUT

```
VAR_OUTPUT
  bBusy      : BOOL;
  st_IM_Signatur : STRING(54);
  bError     : BOOL;
  iErrorID   : UDINT;
END_VAR
```

**bBusy** : Bei der Aktivierung des Funktionsbausteins wird dieser Ausgang gesetzt und bleibt gesetzt, bis eine Rückmeldung erfolgt. Während Busy = TRUE wird an den Eingängen kein neuer Befehl angenommen.

**st\_IM\_Signatur** : Liefert die für das Gerät hinterlegte Hersteller Signatur zurück.

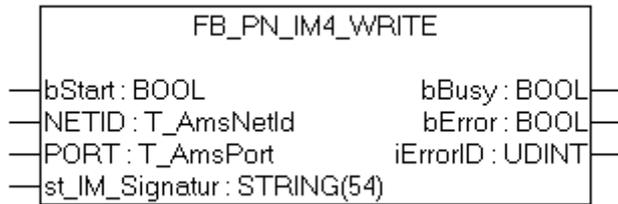
**bError** : Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der *bBusy*-Ausgang zurückgesetzt wurde.

**iErrorID**: Liefert bei einem gesetzten *bError*-Ausgang eine ADS Fehlernummer.

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v2.11.0 R3	PC or CX (x86, ARM)	TcProfinetDiag.Lib

## 6.1.9 FB\_PN\_IM4\_WRITE



Mit diesem Funktionsbaustein schreibt der PROFINET-Controller alle I&M4 ( Identification & Maintenance ) Daten auf ein über *Port* referenziertes Gerät.

Der Frameaufbau der I&M4 Funktion entspricht dem Index [0xAFF4 \[► 68\]](#) nach PROFINET Standard.

### VAR\_INPUT

```
VAR_INPUT
  bStart      : BOOL;
  NETID       : T_AmsNetId; (* AMS Net ID from Controller *)
  PORT        : T_AmsPort; (* Port used by Controller to communicate with Device *)
  st_IM_Signatur : STRING(54);
END_VAR
```

**bStart** : Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert

**NETID** : AMS Net ID des Controllers

**PORT** : Port, über den der Controller mit dem Gerät kommuniziert ( Port = Device ID + 1000hex )

**st\_IM\_Signatur** : Signatur des Herstellers, die auf das Gerät geschrieben wird.

### VAR\_OUTPUT

```
VAR_OUTPUT
  bBusy       : BOOL;
  bError      : BOOL;
  iErrorID    : UDINT;
END_VAR
```

**bBusy** : Bei der Aktivierung des Funktionsbausteins wird dieser Ausgang gesetzt und bleibt gesetzt, bis eine Rückmeldung erfolgt. Während Busy = TRUE wird an den Eingängen kein neuer Befehl angenommen.

**bError** : Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der *bBusy*-Ausgang zurückgesetzt wurde.

**iErrorID**: Liefert bei einem gesetzten *bError*-Ausgang eine ADS Fehlernummer.

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v2.11.0 R3	PC or CX (x86, ARM)	TcProfinetDiag.Lib

## 6.1.10 FB\_PN\_GET\_PORT\_STATISTIC



Dieser Baustein liefert bei Aufruf die statistischen Daten zu den Ports eines PROFINET-Gerätes.

### VAR\_INPUT

```
VAR_INPUT
  bStart   : BOOL;
  NETID    : T_AmsNetId; (* AMS Net ID from Controller *)
  PORT     : T_AmsPort; (* Port used by Controller to communicate with Device *)
END_VAR
```

**bStart** : Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert

**NETID** : AMS Net ID des Controllers

**PORT** : Port über den der Controller mit dem Gerät kommuniziert ( Port = Device ID + 1000hex )

### VAR\_OUTPUT

```
VAR_OUTPUT
  bBusy      : BOOL;
  str_RemotePort_1 : str_GetPortStatistic;
  str_RemotePort_2 : str_GetPortStatistic;
  bPort1     : BOOL;
  bPort2     : BOOL;
END_VAR
```

**bBusy** : Bei der Aktivierung des Funktionsbausteins wird dieser Ausgang gesetzt und bleibt gesetzt, bis eine Rückmeldung erfolgt. Während Busy = TRUE wird an den Eingängen kein neuer Befehl angenommen.

**str\_RemotePort\_1**: Diese [Struktur \[► 68\]](#) enthält die statistischen Daten zu Port 1 .

**str\_RemotePort\_2**: Diese Struktur enthält die statistischen Daten zu Port 2 .

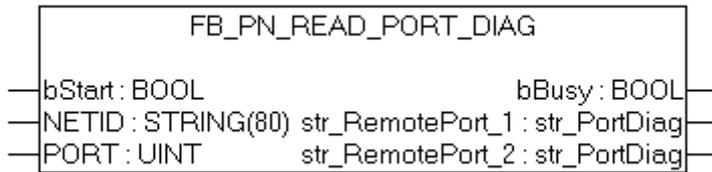
**bPort1**: Ist TRUE, wenn der Port einen Link hat.

**bPort2**: Ist TRUE, wenn der Port einen Link hat.

### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v2.11.0 R3	PC or CX (x86, ARM)	TcProfinetDiag.Lib

### 6.1.11 FB\_PN\_READ\_PORT\_DIAG



Dieser Baustein ruft die Diagnose Informationen der Ports eines PROFINET-Gerätes ab.

#### VAR\_INPUT

```
VAR_INPUT
  bStart : BOOL;
  NETID  : T_AmsNetId; (* AMS Net ID from Controller *)
  PORT   : T_AmsPort; (* Port used by Controller to communicate with Device *)
END_VAR
```

**bStart** : Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert

**NETID** : AMS Net ID des Controllers

**PORT** : Port über den der Controller mit dem Gerät kommuniziert ( Port = Device ID + 1000hex )

#### VAR\_OUTPUT

```
VAR_OUTPUT
  bBusy          : BOOL;
  str_RemotePort_1 : str_PortDiag;
  str_RemotePort_2 : str_PortDiag;
END_VAR
```

**bBusy** : Bei der Aktivierung des Funktionsbausteins wird dieser Ausgang gesetzt und bleibt gesetzt, bis eine Rückmeldung erfolgt. Während Busy = TRUE wird an den Eingängen kein neuer Befehl angenommen.

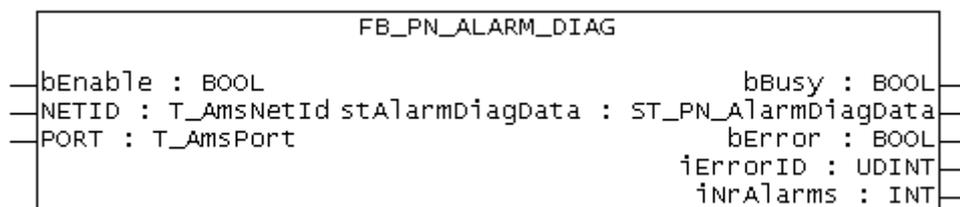
**str\_RemotePort\_1** : Diese Struktur [▶ 69] enthält die Diagnose Informationen zu Port 1 .

**str\_RemotePort\_2** : Diese Struktur enthält die Diagnose Informationen zu Port 2 .

#### Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v2.11.0 R3	PC or CX (x86, ARM)	TcPROFINETDiag.Lib

### 6.1.12 FB\_PN\_ALARM\_DIAG



Mit diesem Funktionsbaustein können Diagnose Alarmer ausgelesen werden. Jede Instanz dieses Bausteins stellt einen PLC-Eingang ("PnIoBoxDiag") zur Verfügung. Dieser Eingang ist mit dem "PnIoBoxDiag" Eingang des Gerätes zu Verknüpfen, welches ausgewertet werden soll. Nach erfolgreichem Auslesen der

Diagnose Alarme / Warnungen, wird der Alarm-Status des Gerätes wieder zurückgesetzt. Der Baustein muss für jedes PROFINET-Gerät einmal aufgerufen werden. Ein Laufindex (iNrAlarms) gibt an, wie viele Diagnosealarme aus dem Puffer gelesen wurden.

### VAR\_INPUT

```
VAR_INPUT
  bEnable : BOOL;
  NETID   : T_AmsNetId; (* AMS Net ID from Controller *)
  PORT    : T_AmsPort; (* Port used by Controller to communicate with Device *)
END_VAR
```

**bEnable** : Aktivierung des Bausteins

**NETID** : AMS Net ID des Controllers

**PORT** : Port, über den der Controller mit dem Gerät kommuniziert ( Port = Device ID + 1000hex )

### VAR\_OUTPUT

```
VAR_OUTPUT
  bBusy          : BOOL;
  stAlarmDiagData : ST_PN_AlarmDiagData;
  bError         : BOOL;
  iErrorID       : UDINT;
  iNrAlarms      : INT;
END_VAR
```

**bBusy** : Bei der Aktivierung des Funktionsbausteins wird dieser Ausgang gesetzt und bleibt gesetzt, bis eine Rückmeldung erfolgt. Während Busy = TRUE wird an den Eingängen kein neuer Befehl angenommen.

**stAlarmDiagData** : Über diese Struktur werden Diagnose Messages ausgegeben. Solange das StausBit [0x0010 = At least one AlarmCR got a diagnosis alarm] am PLC Eingang ansteht, wird in jedem Zyklus ein Alarm über die Struktur ausgegeben.

**bError** : Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der *bBusy*-Ausgang zurückgesetzt wurde.

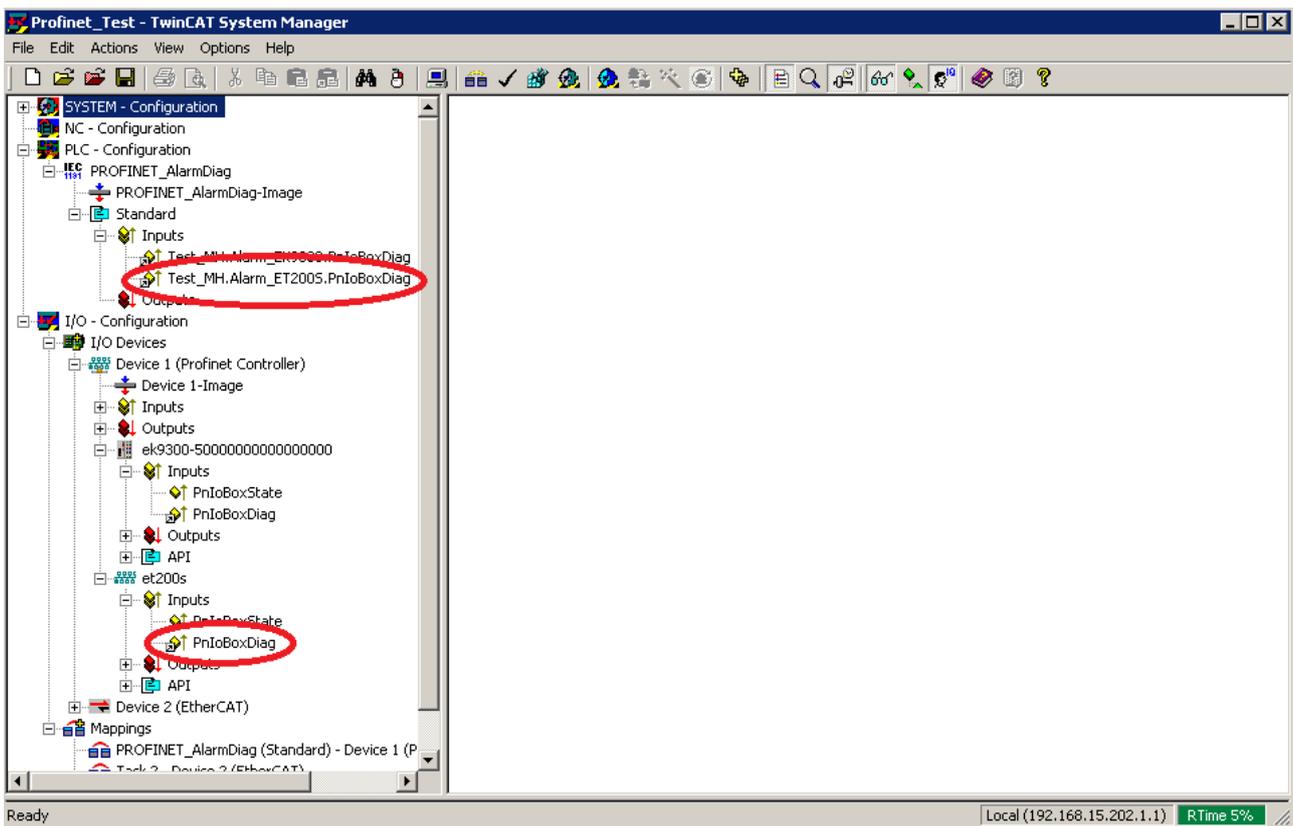
**iErrorID** : Liefert bei einem gesetzten *bError*-Ausgang eine ADS Fehlernummer.

**iNrAlarms** : Anzahl der zuletzt ausgelesenen Alarme.

### VAR

```
VAR
  PnIoBoxDiag AT %I* : WORD; (*Hardware Input*)
END_VAR
```

**PnIoBoxDiag** : Hardware-Eingang. Diese Variable ist mit dem PROFINET-Gerät zu verknüpfen. Über eine Statusänderung dieser Variable wird dem SPS Programm mitgeteilt, dass neue Alarm Diagnose in dem verknüpften PROFINET-Gerät anstehen.



Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v2.11.0 R3	PC or CX (x86, ARM)	TcProfinetDiag.Lib

## 6.2 Datenstrukturen

### 6.2.1 str\_SW\_Rec

Die Datenstruktur **str\_IM\_0xAFF0** bildet die Struktur des I&M0 Frames in der PLC ab. Der Informationen enthält, die in PROFINET-Geräten fest hinterlegt sind.

```

TYPE str_IM_0xAFF0 :
STRUCT
  nBlockTyp      : WORD;
  nBlockLen     : WORD;
  nBlockVersion  : WORD;
  nVendorID     : WORD;
  cOrderID      : STRING(21);
  cSerialNumber  : STRING(17);
  nHW_Rev       : WORD;
  strSW_Rev     : str_SW_Rec;
  nRevCount     : WORD;
  nProfileID    : WORD;
  nProfileSpecType : WORD;
  arIM_Version  : ARRAY[0..1] OF BYTE;
  nSupport      : WORD;
END_STRUCT
END_TYPE
    
```

Die Datenstruktur **str\_SW\_REC** enthält die Software Version des PROFINET Gerätes.

```

TYPE str_SW_Rec :
STRUCT
  cSWRevPrefix   : STRING(2);
  nSWRevFuncEnhance : BYTE;
  nSWRevBugFix   : BYTE;
END_STRUCT
    
```

```
nSWRevIntCha      :BYTE;
END_STRUCT
END_TYPE
```

## 6.2.2 str\_IM\_0xAFF1

Die Datenstruktur **str\_IM\_0xAFF1** bildet die Struktur des I&M1 Frames in der PLC ab. Diese Struktur wird sowohl zum Schreiben als auch zum Lesen von einem PROFINET-Gerät verwendet.

```
TYPE str_IM_0xAFF1 :
STRUCT
  nBlockTyp      : WORD;
  nBlockLen      : WORD;
  nBlockVersion  : WORD;
  st_IM_TagFunction : STRING(32);
  st_IM_TagLocation : STRING(22);
END_STRUCT
END_TYPE
```

## 6.2.3 str\_IM\_0xAFF2

Die Datenstruktur **str\_IM\_0xAFF2** bildet die Struktur des I&M2 Frames in der PLC ab. Diese Struktur wird sowohl zum Schreiben als auch zum Lesen von einem PROFINET-Gerät verwendet.

```
TYPE str_IM_0xAFF1 :
STRUCT
  nBlockTyp      : WORD;
  nBlockLen      : WORD;
  nBlockVersion  : WORD;
  st_IM_Date     : STRING(16);
END_STRUCT
END_TYPE
```

## 6.2.4 str\_IM\_0xAFF3

Die Datenstruktur **str\_IM\_0xAFF3** bildet die Struktur des I&M3 Frames in der PLC ab. Diese Struktur wird sowohl zum Schreiben als auch zum Lesen von einem PROFINET-Gerät verwendet.

```
TYPE str_IM_0xAFF3 :
STRUCT
  nBlockTyp      : WORD;
  nBlockLen      : WORD;
  nBlockVersion  : WORD;
  st_IM_Descriptor : STRING(54)
END_STRUCT
END_TYPE
```

## 6.2.5 str\_IM\_0xAFF4

Die Datenstruktur **str\_IM\_0xAFF4** bildet die Struktur des I&M4 Frames in der PLC ab. Diese Struktur wird sowohl zum Schreiben als auch zum Lesen von einem PROFINET-Gerät verwendet.

```
TYPE str_IM_0xAFF3 :
STRUCT
  nBlockTyp      : WORD;
  nBlockLen      : WORD;
  nBlockVersion  : WORD;
  st_IM_Signatur : STRING(54)
END_STRUCT
END_TYPE
```

## 6.2.6 str\_GetPortStatistic

In der Datenstruktur **str\_GetPortStatistic** werden alle statistischen Informationen eines Gerätes dargestellt.

```
TYPE str_GetPortStatistic :
STRUCT
  Speed          : DWORD;
  PhyMAC         : STRING(50);
  OperatingStatus : STRING(16);
  RxOctets      : DWORD;
```

```

RxUniCastPackets : DWORD;
RxBadPackets     : DWORD;
RxDroppedFrames  : DWORD;
RxUnknownProtocol : DWORD;
TxOctets         : DWORD;
TxUniCastPackets : DWORD;
TxBadPackets     : DWORD;
TxDroppedPackets : DWORD;
END_STRUCT
END_TYPE

```

## 6.2.7 str\_PortDiag

In der Datenstruktur **str\_PortDiag** werden alle Port Diagnose Informationen eines Gerätes dargestellt.

```

TYPE str_PortDiag :
STRUCT
    PortId           : STRING(128);
    PortDescription  : STRING(128);
    SystemName       : STRING(128);
    SystemDescription: STRING(128);
    ChassisId        : STRING(128);
END_STRUCT
END_TYPE

```

## 6.2.8 ST\_PN\_DiagMessage

Die Datenstruktur **ST\_PN\_DiagMessage** enthält den kompletten Datenstream, einer Diagnose-Message, die von einem PROFINET bei Anfrage gesendet wird. Dieser Datenstream wird im Baustein **FB\_PN\_ALARM\_DIAG** ausgewertet und auf eine lesbare Struktur kopiert.

```

TYPE ST_PN_DiagMessage :
STRUCT
    nFlags      : WORD;
    nTextID     : WORD;
    TimeStamp   : ARRAY[0..7] OF BYTE;
    nData       : ARRAY[0..299] OF BYTE;
END_STRUCT
END_TYPE

```

## 6.2.9 ST\_PN\_Diag

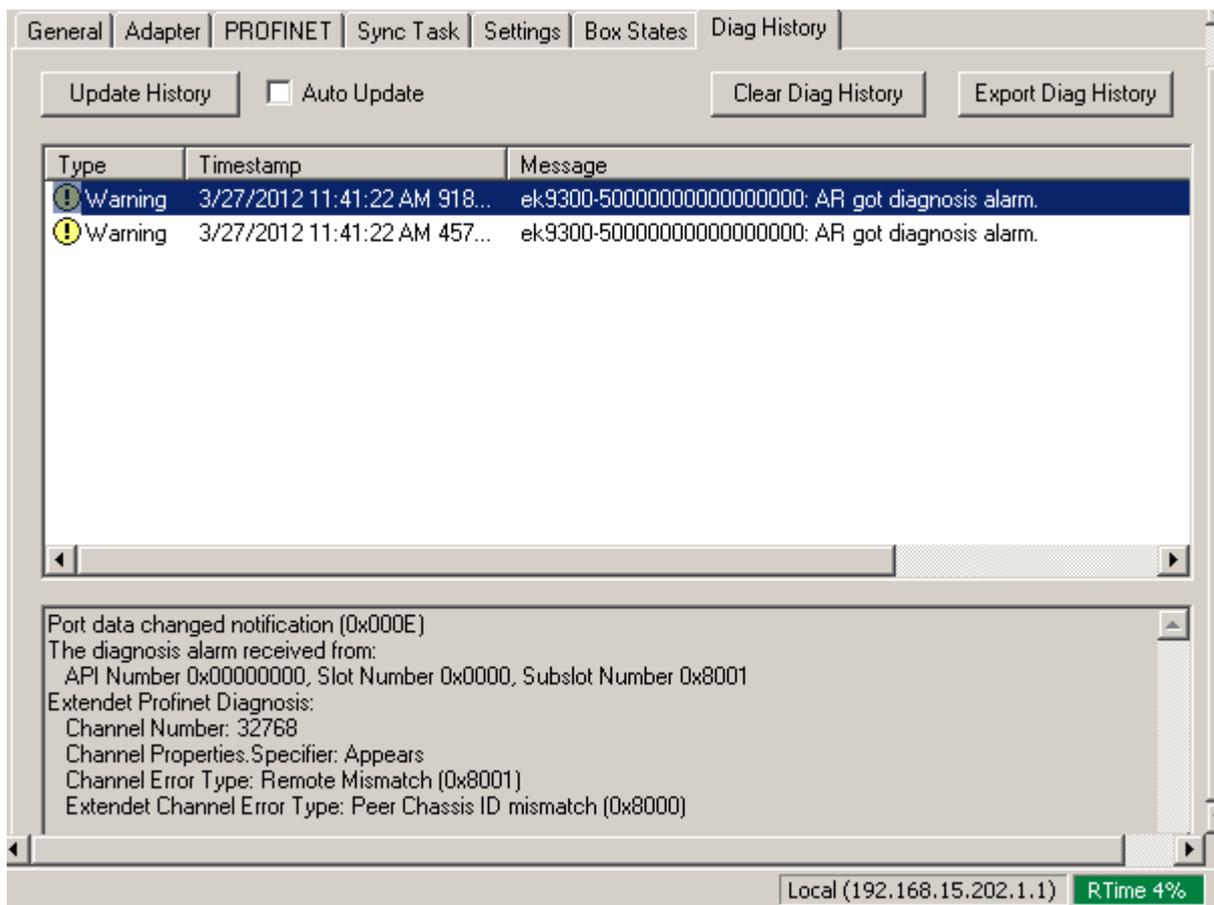
Die Datenstruktur **ST\_PN\_Diag** enthält eine Diagnose-Meldung einer Klemme, die über ein PN-Gerät and einem Controller angeschlossen ist.

```

TYPE str_PortDiag :
STRUCT
    strTimeStamp      : ARRAY[0..7] OF BYTE;
    nAPI              : DWORD;
    nSlot             : WORD;
    nSubSlot          : WORD;
    nAlarmType        : WORD;
    nAlarmSpecifier   : WORD;
    nUserStructIdentifier : WORD;
    nChannelNumber    : WORD;
    nChannelErrorTyp  : WORD;
    nChannelProperties : WORD;
    nExtChannelErrorTyp : WORD;
    arSpare           : ARRAY [1..9] OF WORD;
    arUserSpecificData : ARRAY [0..19] OF BYTE;
END_STRUCT
END_TYPE

```

Der Informationsgehalt der Struktur entspricht dem der Diag History, die im System Manager angezeigt wird.



## 6.2.10 ST\_PN\_AlarmDiagData

Die Datenstruktur **ST\_PN\_AlarmDiagData** enthält den von einem Gerät ausgelesenen Alarmdiagnose-Datensatz einschließlich einem Zeitstempel, wann das Event aufgetreten ist und einem Flag, welches anzeigt, dass "User spezifische" Daten vorhanden sind.

```

TYPE ST_PN_AlarmDiagData :
STRUCT
  ST_TimeStamp      : TIMESTRUCT;
  sNameOfStation    : STRING(20);
  ST_Diag           : ST_PN_Diag;
  bUserSpecData     : BOOL;
END_STRUCT
END_TYPE

```

## 6.2.11 E\_PN\_ALARM\_TYP

### E\_PN\_ALARM\_TYP

Der Aufzählungstyp **E\_PN\_ALARM\_TYP** listet alle Alarme der PROFINET-Kommunikation auf.

```

TYPE E_PN_ALARM_TYP :
(
  PN_ALARM_RESERVE           :=0,
  PN_ALARM_DIAGNOSE_APPEARS :=1,
  PN_ALARM_PROCESS          :=2,
  PN_ALARM_PULL             :=3,
  PN_ALARM_PLUG             :=4,
  PN_ALARM_STATUS           :=5,
  PN_ALARM_UPDATE           :=6,
  PN_ALARM_REDUNDANCY       :=7,
  PN_ALARM_Controlled_by_Supervisor :=8,
  PN_ALARM_Released         :=9,
  PN_ALARM_Plug_Wrong_Submodule :=16#A,
  PN_ALARM_Diagnosis_Disappears :=16#B,
  PN_ALARM_Multicast_Communication_Mismatch :=16#C,
  PN_ALARM_Multicast        :=16#D,
)

```

```
PN_ALARM_STATUS_           :=16#E,  
PN_ALARM_Sync_             :=16#F,  
PN_ALARM_Isochronous_Mode_Problem_Notification :=16#10  
);  
END_TYPE
```

## 7 Anhang

### 7.1 UL Hinweise



#### Application

The modules are intended for use with Beckhoff's UL Listed EtherCAT System only.

#### Examination

For cULus examination, the Beckhoff I/O System has only been investigated for risk of fire and electrical shock (in accordance with UL508 and CSA C22.2 No. 142).

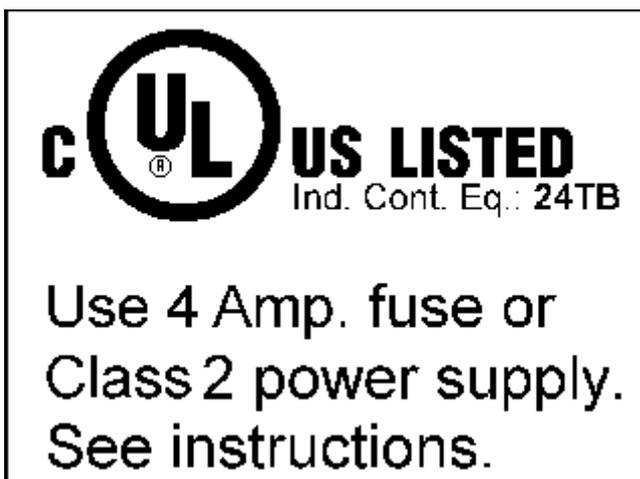
#### For devices with Ethernet connectors

Not for connection to telecommunication circuits.

#### Grundlagen

Im Beckhoff EtherCAT Produktbereich sind je nach Komponente zwei UL-Zertifikate anzutreffen:

- UL-Zertifizierung nach UL508 mit eingeschränkter Leistungsaufnahme  
Die Stromaufnahme durch das Gerät wird begrenzt auf eine max. mögliche Stromaufnahme von 4 A.  
Solcherart zertifizierte Geräte sind gekennzeichnet durch das Zeichen



Annähernd alle aktuellen EtherCAT Produkte (Stand 2010/05) sind uneingeschränkt UL zertifiziert.

#### Anwendung

Werden *eingeschränkt* zertifizierte Klemmen verwendet, ist die Stromaufnahme bei 24 V<sub>DC</sub> entsprechend zu beschränken durch Versorgung

- von einer isolierten, mit einer Sicherung (entsprechend UL248) von maximal 4 A geschützten Quelle, oder
- von einer Spannungsquelle die *NEC class 2* entspricht.  
Eine Spannungsquelle entsprechend *NEC class 2* darf nicht seriell oder parallel mit einer anderen *NEC class 2* entsprechenden Spannungsquelle verbunden werden!

Diese Anforderungen gelten für die Versorgung aller EtherCAT Buskoppler, Netzteilklemmen, Busklemmen und deren Power-Kontakte.

## 7.2 Firmware Kompatibilität

Beckhoff EtherCAT Geräte werden mit dem aktuell verfügbaren letzten Firmware-Stand ausgeliefert. Dabei bestehen zwingende Abhängigkeiten zwischen Firmware und Hardware; eine Kompatibilität ist nicht in jeder Kombination gegeben. Die unten angegebene Übersicht zeigt auf welchem Hardware-Stand eine Firmware betrieben werden kann.

- Es wird empfohlen, die für die jeweilige Hardware letztmögliche Firmware einzusetzen
- Ein Anspruch auf ein kostenfreies Firmware-Update bei ausgelieferten Produkten durch Beckhoff gegenüber dem Kunden besteht nicht.

### HINWEIS

#### Beschädigung des Gerätes möglich!

Beachten Sie die Hinweise zum Firmware Update auf der gesonderten Seite. Wird ein Gerät in den BOOTSTRAP-Mode zum Firmware-Update versetzt, prüft es u.U. beim Download nicht, ob die neue Firmware geeignet ist. Dadurch kann es zur Beschädigung des Gerätes kommen! Vergewissern Sie sich daher immer, ob die Firmware für den Hardware-Stand des Gerätes geeignet ist!

#### EL6631-0010

Hardware (HW)	Firmware	Revision Nr.	Releasedatum
03 - 04*	01*	EL6631-0010-0016	2011/05

\*) Zum Zeitpunkt der Erstellung dieser Dokumentation ist dies der aktuelle kompatible Firmware-/Hardware-Stand. Überprüfen Sie auf der Beckhoff Webseite, ob eine aktuellere [Dokumentation](#) vorliegt.

## 7.3 Firmware EL/ES/EM/EPxxxx

In diesem Kapitel wird das Geräteupdate für Beckhoff EtherCAT Slaves der Serien EL/ES, EM, EK und EP beschrieben. Ein FW-Update sollte nur nach Rücksprache mit dem Beckhoff Support durchgeführt werden.

#### Inhaltsverzeichnis

- [Gerätebeschreibung ESI-File/XML](#) [▶ 74]
- [Versionsbestimmung Firmware](#) [▶ 77]
- [Update Controller-Firmware \\*.efw](#) [▶ 78]
- [Update FPGA-Firmware \\*.rbf](#) [▶ 79]
- [Mehrfaches Update](#) [▶ 82]

#### Speicherorte

In einem EtherCAT-Slave werden an bis zu 3 Orten Daten für den Betrieb vorgehalten:

- Je nach Funktionsumfang und Performance besitzen EtherCAT Slaves einen oder mehrere lokale Controller zur Verarbeitung von IO-Daten. Das darauf laufende Programm ist die sog. **Firmware** im Format \*.efw.
- In bestimmten EtherCAT Slaves kann auch die EtherCAT Kommunikation in diesen Controller integriert sein. Dann ist der Controller meist ein so genannter **FPGA-Chip** mit der \*.rbf-Firmware.

- Darüber hinaus besitzt jeder EtherCAT Slave einen Speicherchip, um seine eigene Gerätebeschreibung zu speichern, in einem sog. **EEPROM**. Beim Einschalten wird diese Beschreibung geladen und u.a. die EtherCAT Kommunikation entsprechend eingerichtet. Die Gerätebeschreibung kann von der Beckhoff Website (<http://www.beckhoff.de>) im Downloadbereich heruntergeladen werden. Dort sind alle ESI-Dateien (**E**therCAT **S**lave **I**nformation) als Zip-Datei zugänglich.

Kundenseitig zugänglich sind diese Daten nur über den Feldbus EtherCAT und seine Kommunikationsmechanismen. Beim Update oder Auslesen dieser Daten ist insbesondere die azyklische Mailbox-Kommunikation oder der Registerzugriff auf den ESC in Benutzung.

Der TwinCAT Systemmanager bietet Mechanismen, um alle 3 Teile mit neuen Daten programmieren zu können, wenn der Slave dafür vorgesehen ist. Es findet üblicherweise keine Kontrolle durch den Slave statt, ob die neuen Daten für ihn geeignet sind, ggf. ist ein Weiterbetrieb nicht mehr möglich.

### HINWEIS

#### Beschädigung des Gerätes möglich!

- Das Herunterladen der Firmware auf ein EtherCAT-Gerät darf nicht unterbrochen werden.
- Eine einwandfreie EtherCAT-Kommunikation muss sichergestellt sein, CRC-Fehler oder LostFrames dürfen nicht auftreten.
- Die Spannungsversorgung muss ausreichend dimensioniert, die Pegel entsprechend der Vorgabe sein.

#### Gerätebeschreibung ESI-File/XML

### HINWEIS

#### Update der ESI-Beschreibung/EEPROM

Manche Slaves haben Abgleich- und Konfigurationsdaten aus der Produktion im EEPROM abgelegt. Diese werden bei einem Update unwiederbringlich überschrieben.

Die Gerätebeschreibung ESI wird auf dem Slave lokal gespeichert und beim Start geladen. Jede Gerätebeschreibung hat eine eindeutige Kennung aus Slave name (9-stellig) und Revision-Nummer (4-stellig). Jeder im System-Manager konfigurierte Slave zeigt seine Kennung im EtherCAT-Reiter (Geräteerkennung aus Name EL3204-0000 und Revision -0016

):

Die konfigurierte Kennung muss kompatibel sein mit der tatsächlich als Hardware eingesetzten Gerätebeschreibung, d.h. der Beschreibung die der Slave (hier: EL3204) beim Start geladen hat. Üblicherweise muss dazu die konfigurierte Revision gleich oder niedriger der tatsächlich im Klemmenverbund befindlichen sein.

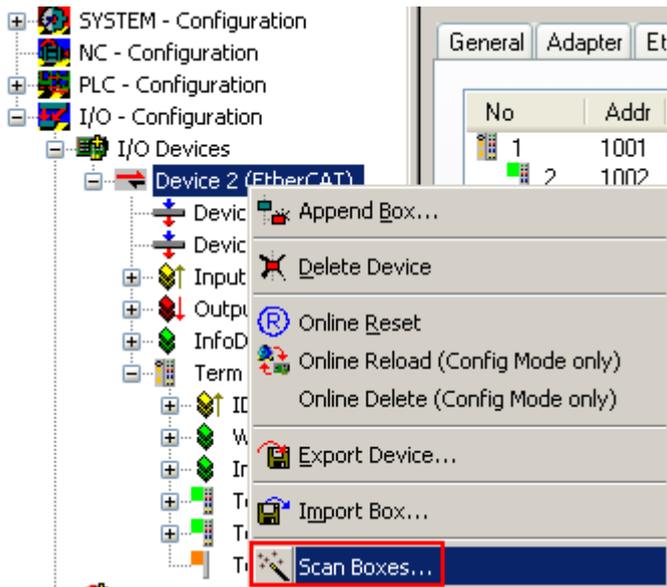
Weitere Hinweise hierzu entnehmen Sie bitte der [EtherCAT System-Dokumentation](#).

**i Update von XML/ESI-Beschreibung**

Die Geräteversion steht in engem Zusammenhang mit der verwendeten Firmware bzw. Hardware. Nicht kompatible Kombinationen führen mindestens zu Fehlfunktionen oder sogar zur endgültigen Außerbetriebsetzung des Gerätes. Ein entsprechendes Update sollte nur in Rücksprache mit dem Beckhoff Support ausgeführt werden.

**Anzeige der Slave-Kennung ESI**

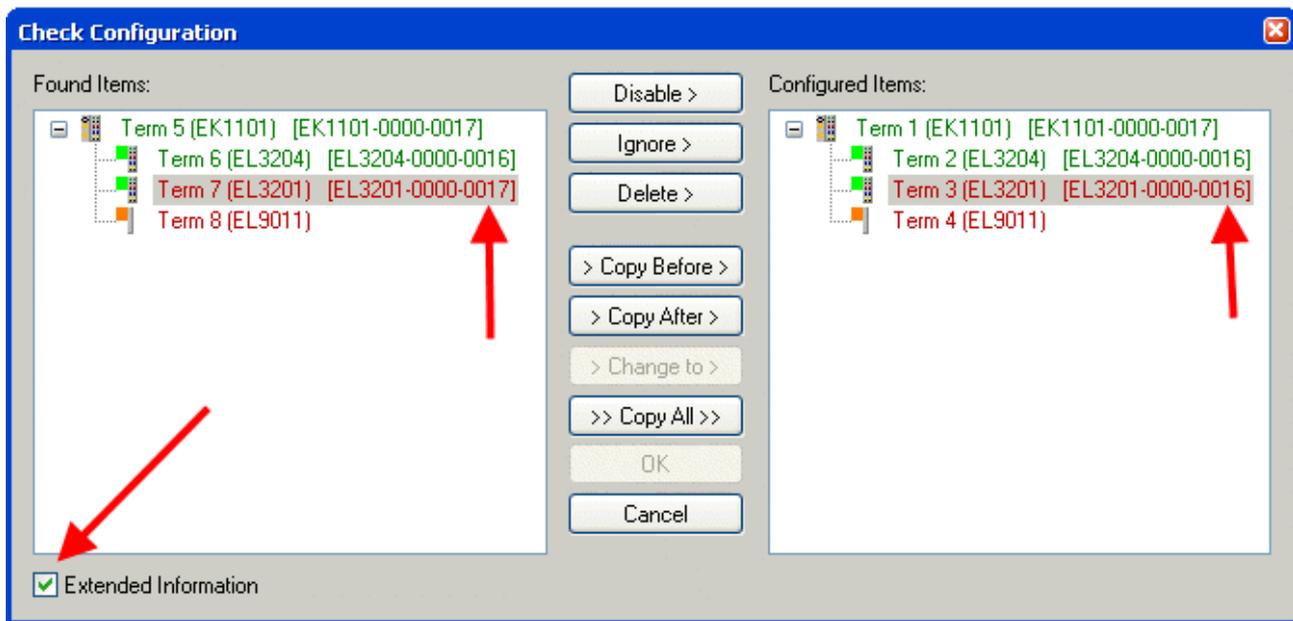
Der einfachste Weg, die Übereinstimmung von konfigurierter und tatsächlicher Gerätebeschreibung festzustellen, ist im TwinCAT Modus Config/FreeRun das Scannen der EtherCAT-Boxen auszuführen, ein Rechtsklick auf das EtherCAT Gerät bewirkt im Config/FreeRun-Mode das Scannen des unterlagerten Feldes:



Wenn das gefundene Feld mit dem konfigurierten übereinstimmt, erscheint: Konfiguration identisch,



ansonsten ein Änderungsdialog, um die realen Angaben in die Konfiguration zu übernehmen:



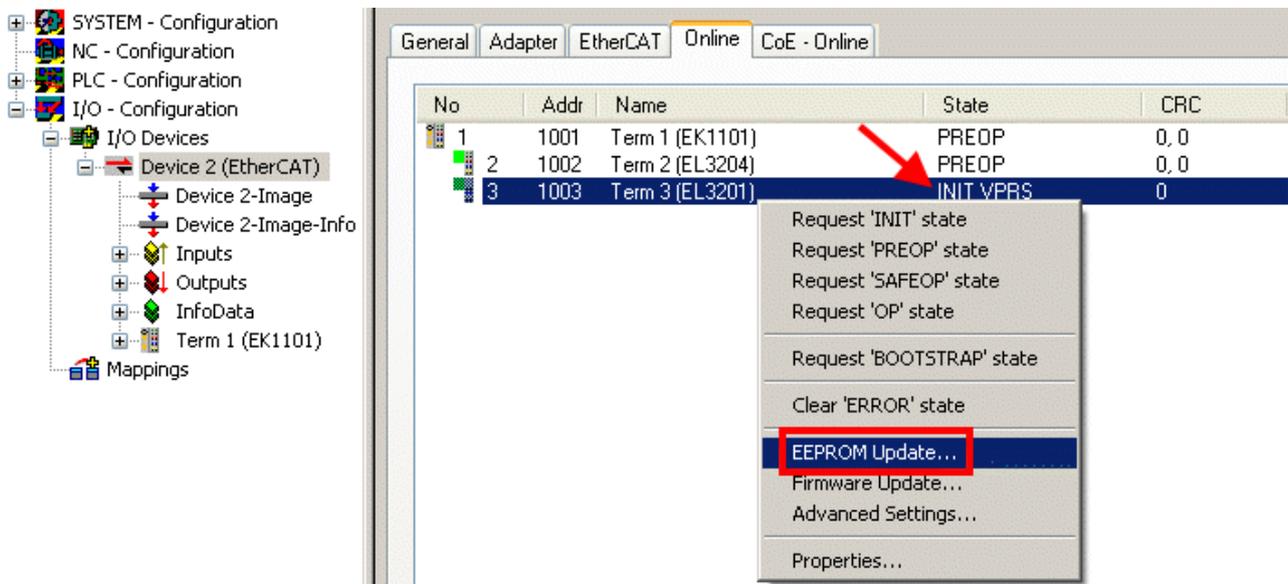
In diesem Beispiel wurde eine EL3201-0000-0017 vorgefunden, während eine EL3201-0000-0016 konfiguriert wurde. In diesem Fall bietet es sich an, mit dem *Copy Before*-Button die Konfiguration anzupassen. Die Checkbox *Extended Information* muss gesetzt werden, um die Revision angezeigt zu bekommen.

## Änderung der Slave-Kennung ESI

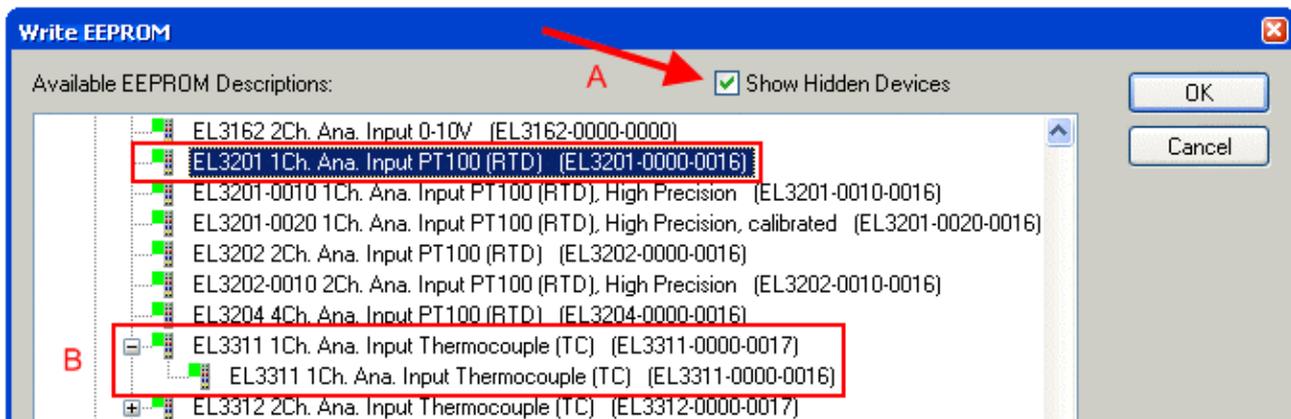
### Änderung der Slave-Kennung ESI

Die ESI/EEPROM-Kennung kann unter TwinCAT wie folgt aktualisiert werden:

- Einwandfreie EtherCAT-Kommunikation muss zum Slave hergestellt werden
- Der State des Slave ist unerheblich
- Rechtsklick auf den Slave in der Online-Anzeige führt zum Dialog *EEPROM Update*:



Im folgenden Dialog wird die neue ESI-Beschreibung ausgewählt. Die CheckBox *Show Hidden Devices* zeigt auch ältere, normalerweise ausgeblendete Ausgaben eines Slave':



Ein Laufbalken im Systemmanager zeigt den Fortschritt - erst erfolgt das Schreiben, dann das Verifying.

### ● Änderung erst nach Neustart wirksam

**i** Die meisten EtherCAT-Geräte lesen eine geänderte ESI-Beschreibung umgehend bzw. nach dem Aufstarten aus dem INIT ein. Einige Kommunikationseinstellungen wie z.B. Distributed Clocks werden jedoch erst bei PowerOn gelesen. Deshalb ist ein kurzes Abschalten des EtherCAT Slave nötig, damit die Änderung wirksam wird.

### Versionsbestimmung der Firmware

#### Versionsbestimmung nach Laseraufdruck

Auf einem Beckhoff EtherCAT Slave ist eine Seriennummer aufgelasert. Der Aufbau der Seriennummer lautet: **KK YY FF HH**

KK - Produktionswoche (Kalenderwoche)  
 YY - Produktionsjahr  
 FF - Firmware-Stand  
 HH - Hardware-Stand

Beispiel mit Ser. Nr.: 12 10 03 02:

12 - Produktionswoche 12  
 10 - Produktionsjahr 2010  
 03 - Firmware-Stand 03  
 02 - Hardware-Stand 02

#### Versionsbestimmung mit dem System-Manager

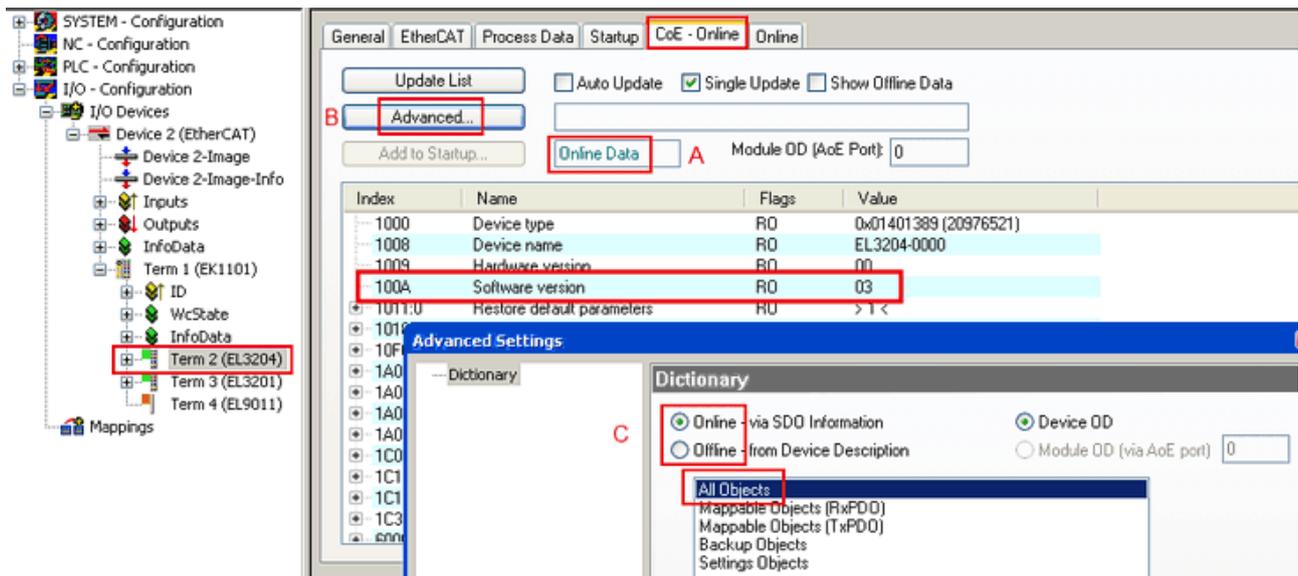
Der TwinCAT System-Manager zeigt die Version der Controller-Firmware an, wenn der Slave online für den Master zugänglich ist. Klicken Sie hierzu auf die E-Bus-Klemme deren Controller-Firmware Sie überprüfen möchten (im Beispiel Klemme 2 (EL3204) und wählen Sie den Karteireiter *CoE-Online* (CAN over EtherCAT).

### ● CoE-Online und Offline-CoE

**i**

- **online:** es wird im EtherCAT Slave vom Controller angeboten, wenn der EtherCAT Slave dies unterstützt.  
Dieses CoE-Verzeichnis kann nur bei angeschlossenem und betriebsbereitem Slave angezeigt werden.
- **offline:** in der EtherCAT Slave Information ESI/XML kann der Default-Inhalt des CoE enthalten sein.  
Dieses CoE-Verzeichnis kann nur angezeigt werden, wenn es in der ESI (z.B. "Beckhoff EL5xxx.xml") enthalten ist.

Im folgenden Dialog wird der FW-Stand der markierten EL3204 in CoE-Eintrag x100A mit 03 angezeigt:



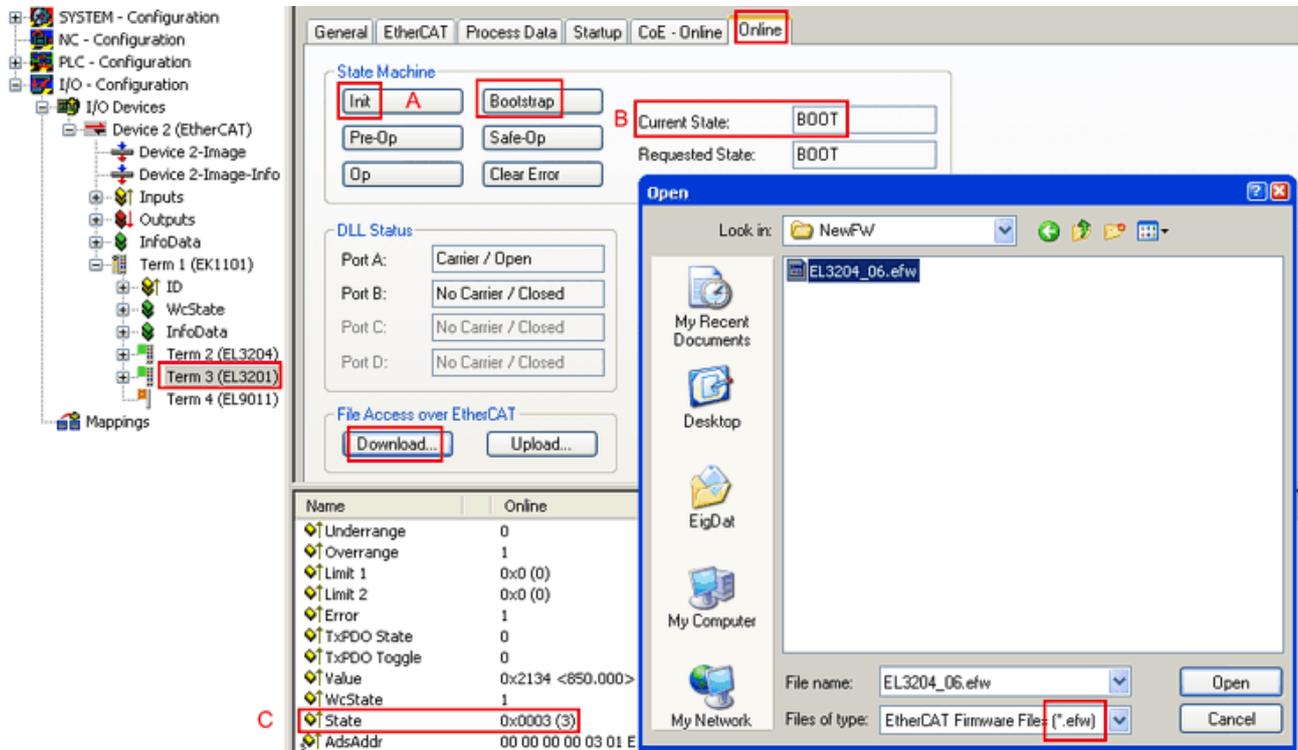
TwinCAT 2.11 zeigt in (A) an, dass aktuell das Online-CoE-Verzeichnis angezeigt wird. Ist dies nicht der Fall, kann durch die erweiterten Einstellungen (B) durch *Online* und Doppelklick auf *All Objects* das Online-Verzeichnis geladen werden.

**Update Controller-Firmware \*.efw**

**i CoE-Verzeichnis**

Das Online-CoE-Verzeichnis wird vom Controller verwaltet und in einem eigenen EEPROM gespeichert. Es wird durch ein FW-Update i.allg. nicht verändert.

Um die Controller-Firmware eines Slave zu aktualisieren, wechseln Sie zum Karteireiter *Online* :



Es ist folgender Ablauf einzuhalten, wenn keine anderen Angaben z.B. durch den Beckhoff Support vorliegen.

- Slave in INIT schalten (A)
- Slave in BOOTSTRAP schalten
- Kontrolle des aktuellen Status (B, C)

- Download der neuen \*efw-Datei
- Nach Beendigung des Download in INIT schalten, dann in OP
- Slave kurz stromlos schalten

**FPGA-Firmware \*.rbf**

Falls ein FPGA-Chip die EtherCAT Kommunikation übernimmt, kann ggf. mit einer \*.rbf-Datei ein Update durchgeführt werden.

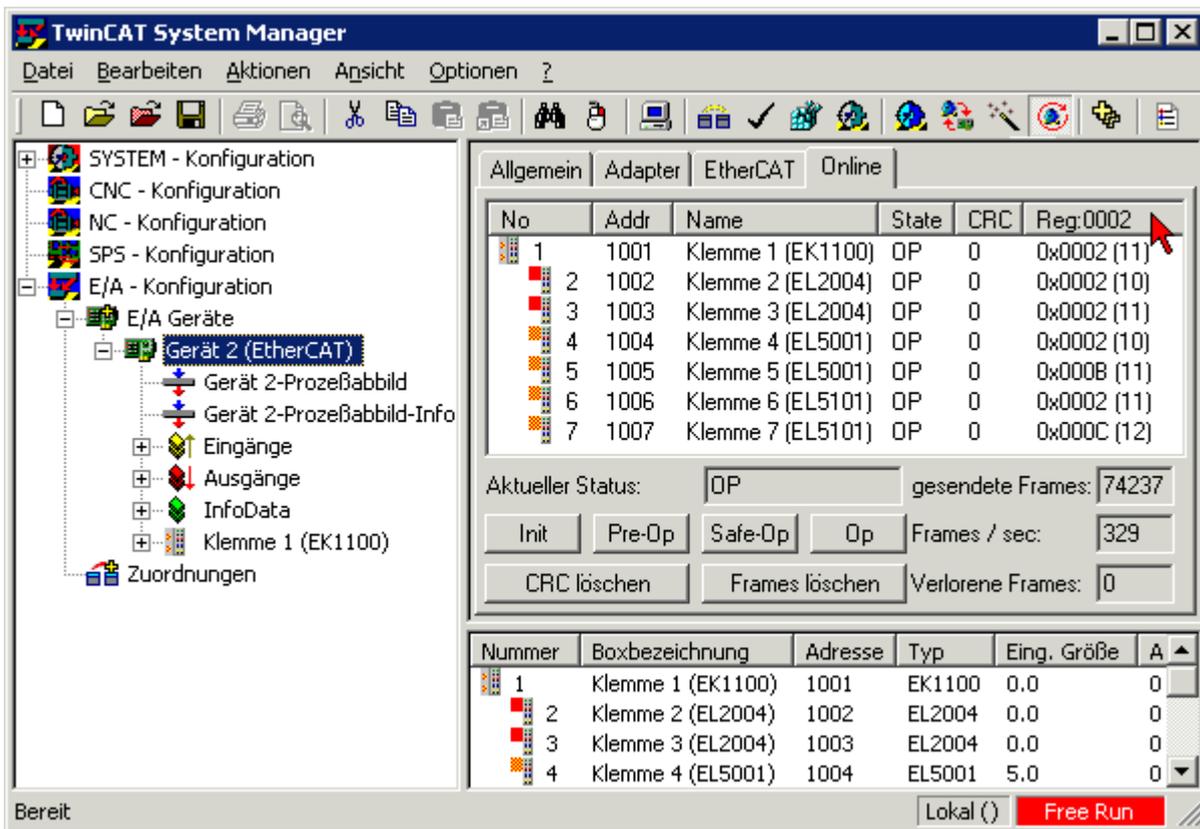
- [Controller-Firmware \[► 78\]](#) für die Aufbereitung der E/A-Signale
- [FPGA-Firmware \[► 79\]](#) für die EtherCAT-Kommunikation (nur für Klemmen mit FPGA)

Die in der Seriennummer der Klemme enthaltene Firmware-Versionsnummer beinhaltet beide Firmware-Teile. Wenn auch nur eine dieser Firmwarekomponenten verändert wird, dann wird diese Versionsnummer fortgeschrieben.

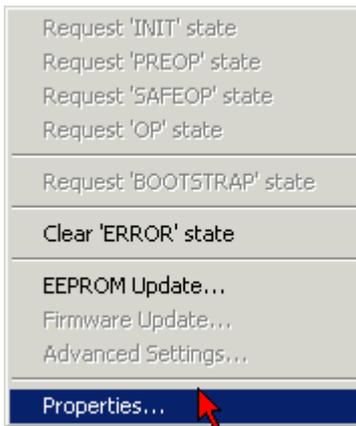
**Versionsbestimmung mit dem System-Manager**

Der TwinCAT System-Manager zeigt die Version der FPGA-Firmware an. Klicken Sie hierzu auf die Ethernet-Karte Ihres EtherCAT-Stranges (im Beispiel Gerät 2) und wählen Sie den Karteireiter *Online*.

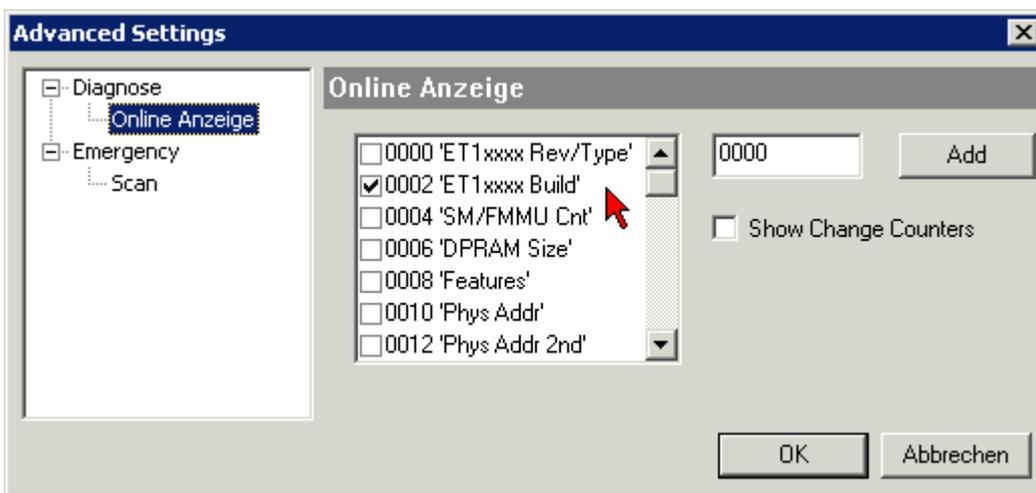
Die Spalte *Reg:0002* zeigt die Firmware-Version der einzelnen EtherCAT-Geräte in hexadezimaler und dezimaler Darstellung an.



Falls die Spalte *Reg:0002* nicht angezeigt wird, klicken sie mit der rechten Maustaste auf den Tabellenkopf und wählen im erscheinenden Kontextmenü, den Menüpunkt *Properties*.



In dem folgenden Dialog *Advanced Settings* können Sie festlegen, welche Spalten angezeigt werden sollen. Markieren Sie dort unter *Diagnose/Online Anzeige* das Kontrollkästchen vor *'0002 ETxxx Build'* um die Anzeige der FPGA-Firmware-Version zu aktivieren.



## Update

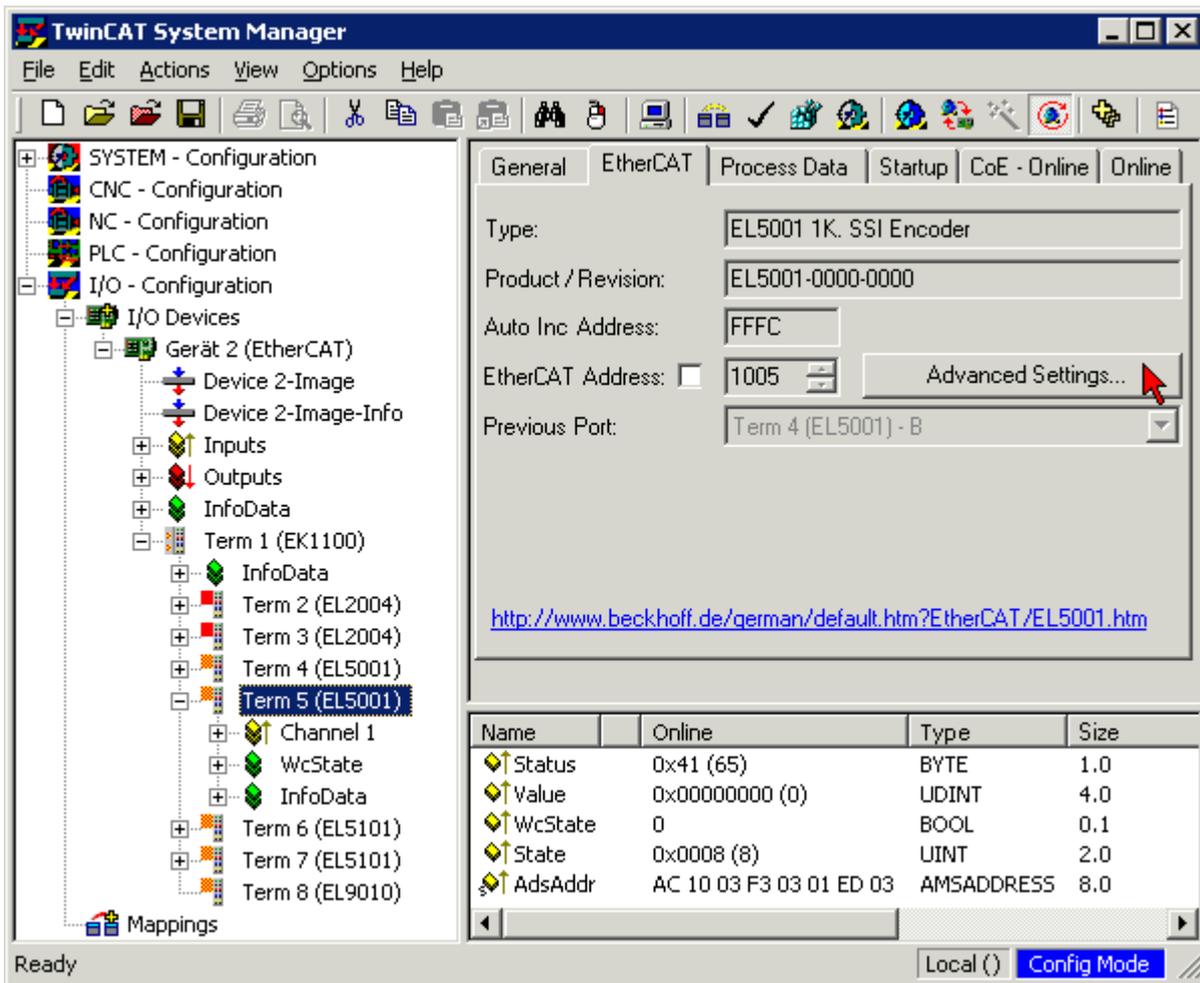
Für das Update der FPGA-Firmware

- eines EtherCAT-Kopplers, muss auf auf diesem Koppler mindestens die FPGA-Firmware-Version 11 vorhanden sein.
- einer E-Bus-Klemme, muss auf auf dieser Klemme mindestens die FPGA-Firmware-Version 10 vorhanden sein.

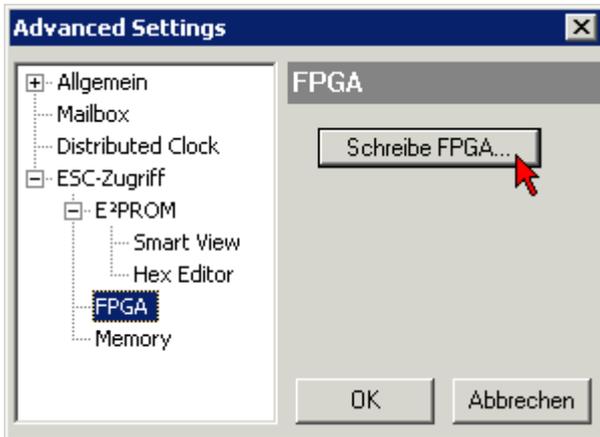
Ältere Firmwarestände können nur vom Hersteller aktualisiert werden!

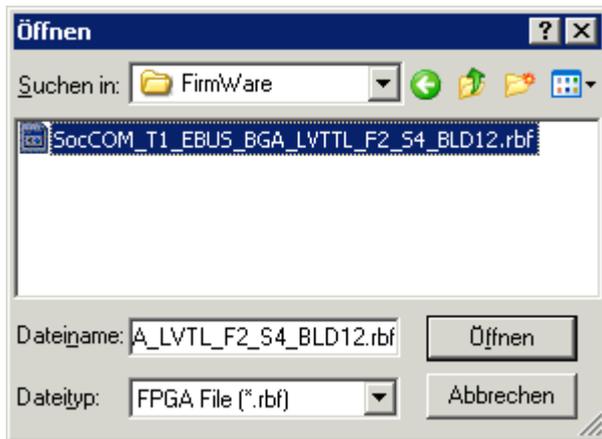
## Update eines EtherCAT-Geräts

Wählen Sie im TwinCAT System-Manager die Klemme an, deren FPGA-Firmware Sie aktualisieren möchten (im Beispiel: Klemme 5: EL5001) und klicken Sie auf dem Karteireiter *EtherCAT* auf die Schaltfläche *Weitere Einstellungen*.



Im folgenden Dialog *Advanced Settings* klicken Sie im Menüpunkt *ESC-Zugriff/E<sup>2</sup>PROM/FPGA* auf die Schaltfläche *Schreibe FPGA*,





Wählen Sie die Datei (\*.rbf) mit der neuen FPGA-Firmware aus und übertragen Sie diese zum EtherCAT-Gerät.

### HINWEIS

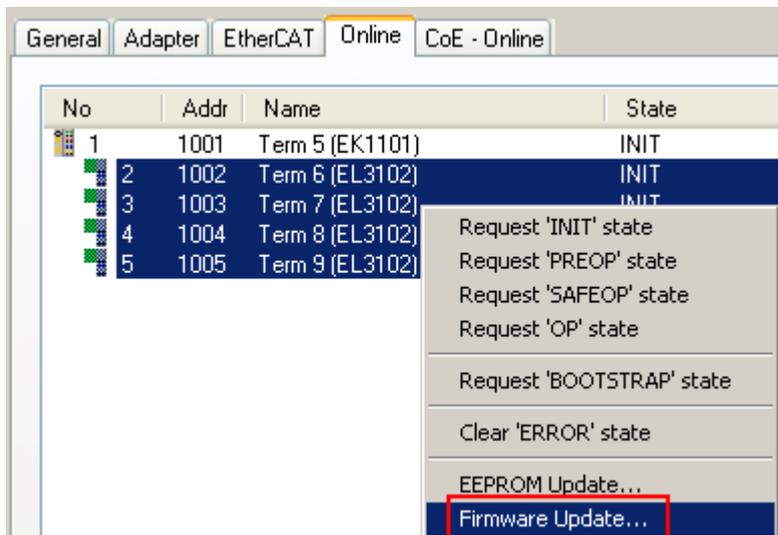
#### Beschädigung des Gerätes möglich!

Das Herunterladen der Firmware auf ein EtherCAT-Gerät dürfen Sie auf keinen Fall unterbrechen! Wenn Sie diesen Vorgang abbrechen, dabei die Versorgungsspannung ausschalten oder die Ethernet-Verbindung unterbrechen, kann das EtherCAT-Gerät nur vom Hersteller wieder in Betrieb genommen werden!

Um die neue FPGA-Firmware zu aktivieren ist ein Neustart (Aus- und Wiedereinschalten der Spannungsversorgung) des EtherCAT-Geräts erforderlich.

#### Gleichzeitiges Update mehrerer EtherCAT-Geräte

Die Firmware von mehreren Geräten kann gleichzeitig aktualisiert werden, ebenso wie die ESI-Beschreibung. Voraussetzung hierfür ist, dass für diese Geräte die gleiche Firmware-Datei/ESI gilt.



Wählen Sie dazu die betreffenden Slaves aus und führen Sie den FW-Update im BOOTSTRAP Modus wie o.a. aus.

## 7.4 ATEX-Dokumentation



### Hinweise zum Einsatz des Busklemmensystems in explosionsgefährdeten Bereichen (ATEX)

Beachten Sie auch die weiterführende Dokumentation Hinweise zum Einsatz des Busklemmensystems in explosionsgefährdeten Bereichen (ATEX) die Ihnen auf der Beckhoff-Homepage <http://www.beckhoff.de> im Bereich Download zur Verfügung steht!

## 7.5 Support und Service

Beckhoff und seine weltweiten Partnerfirmen bieten einen umfassenden Support und Service, der eine schnelle und kompetente Unterstützung bei allen Fragen zu Beckhoff Produkten und Systemlösungen zur Verfügung stellt.

### Downloadfinder

Unser [Downloadfinder](#) beinhaltet alle Dateien, die wir Ihnen zum Herunterladen anbieten. Sie finden dort Applikationsberichte, technische Dokumentationen, technische Zeichnungen, Konfigurationsdateien und vieles mehr.

Die Downloads sind in verschiedenen Formaten erhältlich.

### Beckhoff Niederlassungen und Vertretungen

Wenden Sie sich bitte an Ihre Beckhoff Niederlassung oder Ihre Vertretung für den [lokalen Support und Service](#) zu Beckhoff Produkten!

Die Adressen der weltweiten Beckhoff Niederlassungen und Vertretungen entnehmen Sie bitte unserer Internetseite: [www.beckhoff.com](http://www.beckhoff.com)

Dort finden Sie auch weitere Dokumentationen zu Beckhoff Komponenten.

### Beckhoff Support

Der Support bietet Ihnen einen umfangreichen technischen Support, der Sie nicht nur bei dem Einsatz einzelner Beckhoff Produkte, sondern auch bei weiteren umfassenden Dienstleistungen unterstützt:

- Support
- Planung, Programmierung und Inbetriebnahme komplexer Automatisierungssysteme
- umfangreiches Schulungsprogramm für Beckhoff Systemkomponenten

Hotline: +49 5246 963-157  
E-Mail: [support@beckhoff.com](mailto:support@beckhoff.com)

### Beckhoff Service

Das Beckhoff Service-Center unterstützt Sie rund um den After-Sales-Service:

- Vor-Ort-Service
- Reparaturservice
- Ersatzteilservice
- Hotline-Service

Hotline: +49 5246 963-460  
E-Mail: [service@beckhoff.com](mailto:service@beckhoff.com)

### Beckhoff Unternehmenszentrale

Beckhoff Automation GmbH & Co. KG

Hülshorstweg 20  
33415 Verl  
Deutschland

Telefon: +49 5246 963-0  
E-Mail: [info@beckhoff.com](mailto:info@beckhoff.com)  
Internet: [www.beckhoff.com](http://www.beckhoff.com)



Mehr Informationen:  
**[www.beckhoff.de/ts6271](http://www.beckhoff.de/ts6271)**

Beckhoff Automation GmbH & Co. KG  
Hülshorstweg 20  
33415 Verl  
Deutschland  
Telefon: +49 5246 9630  
[info@beckhoff.com](mailto:info@beckhoff.com)  
[www.beckhoff.com](http://www.beckhoff.com)

