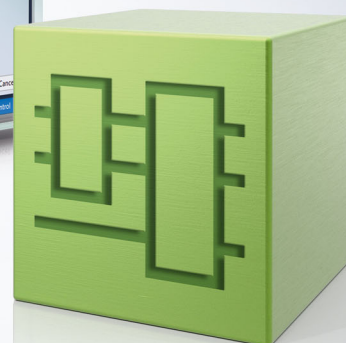
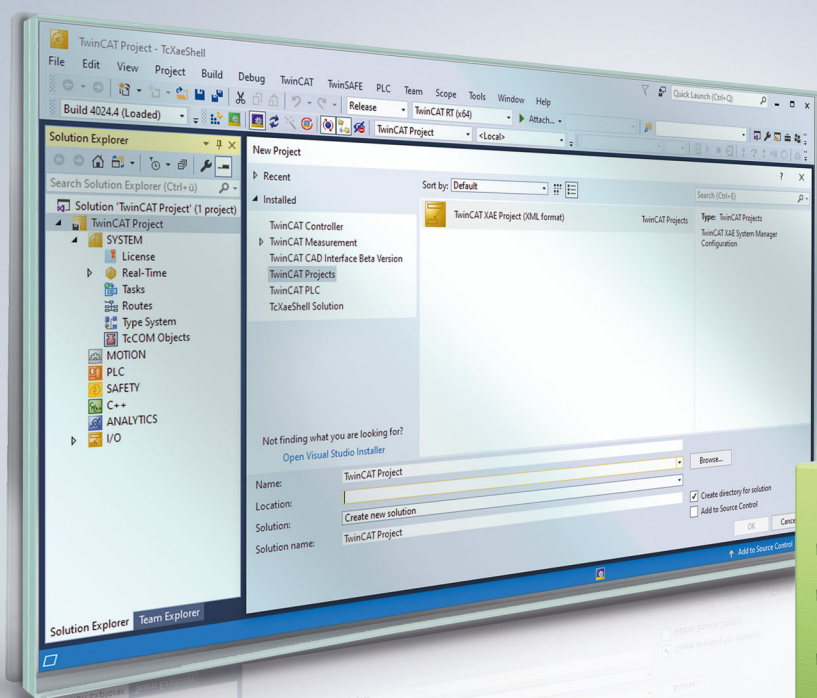


BECKHOFF New Automation Technology

Handbuch | DE

TE1000

TwinCAT 3 | PLC-Bibliothek: Tc2_ProfinetDiag



Inhaltsverzeichnis

1	Vorwort.....	5
1.1	Hinweise zur Dokumentation	5
1.2	Zu Ihrer Sicherheit.....	6
1.3	Hinweise zur Informationssicherheit	7
2	Übersicht.....	8
3	Funktionsbausteine	9
3.1	Controller.....	10
3.1.1	AlarmDiag	10
3.1.2	Identification and Maintenance	12
3.1.3	PROFINET RT Controller.....	21
3.1.4	FB_SET_PN_NAME	23
3.1.5	FB_RESET_PN_TO_FACTORY_SETTINGS	24
3.1.6	FB_PN_SCAN.....	25
3.1.7	FB_PN_SCAN_UpTo255.....	26
3.2	Device	27
3.2.1	EL6631-0010.....	27
3.2.2	Device via CCAT (CX-B930,TF6270)	30
3.2.3	FB_PN_SEND_ALARM	35
3.3	FB_PN_GET_PORT_STATISTIC.....	36
3.4	FB_PN_READ_PORT_DIAG.....	37
4	Datenstrukturen	39
4.1	Controller.....	39
4.1.1	IM	39
4.1.2	str_PN_Scan	40
4.1.3	AlarmDiag	41
4.1.4	ST_PN_DeviceInfo.....	42
4.2	Device	42
4.2.1	str_Diag_PN_Settings.....	42
4.2.2	str_luM_Data.....	42
4.3	EL6631-0010.....	43
4.3.1	E_PN_ALARM_TYP	43
4.4	IM	43
4.4.1	RecStruct	43
4.5	Port Diagnose	43
4.5.1	str_GetPortStatistic	43
4.5.2	str_PortDiag	44
5	Anhang.....	45
5.1	ADS Return Codes.....	45
5.2	Support und Service.....	50

1 Vorwort

1.1 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.

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

1.2 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.

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>.

2 Übersicht

Die SPS-Bibliothek enthält für den Einsatz am PROFINET Controller und am Device fertige Funktionsbausteine, welche bei der Handhabung, der Diagnose und mehr unterstützen. Die Bezeichnungen der verschiedenen Funktionsbausteine können sich ähneln, daher muss der Nutzer vor der Verwendung beachten, dass zwischen „Controller“ und „Device“ unterschieden wird.

Systemvoraussetzung

Target System	IPC or CX, (x86, x64, ARM)
Min. TwinCAT-Version	3.1.4018
Min. TwinCAT-Level	TC1200 TC3 PLC

3 Funktionsbausteine

Controller

Controller

Bausteine Controller	Bedeutung	Beschreibung
FB_SET_PN_NAME [▶ 23]	Vergibt dem jeweiligen PROFINET Device einen Namen	TF6271 und EL663x
FB_RESET_PN_TO_FACTORY_SETTINGS [▶ 24]	Setzt das jeweilige PROFINET Device auf Herstellereinstellungen	TF6271 und EL663x
FB_PN_SCAN [▶ 25]	Scannt das PROFINET Netzwerk und liefert die Anzahl sowie eine Liste mit den gefundenen PROFINET Devices	TF6271 und EL663x

AlarmDiag

Bausteine AlarmDiag	Bedeutung	Beschreibung
FB_PN_ALARM_DIAG [▶ 10]	Lesen der Diagnose Alarme	TF6271 und EL663x

I&M (Identification & Maintenance)

Bausteine I&M Funktionen	Bedeutung	Beschreibung
FB_PN_IM0_READ [▶ 12]	Lesen der I&M Funktion 0	TF6271 und EL663x
FB_PN_IM1_READ [▶ 13]	Lesen der I&M Funktion 1	TF6271 und EL663x
FB_PN_IM2_READ [▶ 15]	Lesen der I&M Funktion 2	TF6271 und EL663x
FB_PN_IM3_READ [▶ 17]	Lesen der I&M Funktion 3	TF6271 und EL663x
FB_PN_IM4_READ [▶ 19]	Lesen der I&M Funktion 4	TF6271 und EL663x
FB_PN_IM1_WRITE [▶ 14]	Schreiben der I&M Funktion 1	TF6271 und EL663x
FB_PN_IM2_WRITE [▶ 16]	Schreiben der I&M Funktion 2	TF6271 und EL663x
FB_PN_IM3_WRITE [▶ 18]	Schreiben der I&M Funktion 3	TF6271 und EL663x
FB_PN_IM4_WRITE [▶ 20]	Schreiben der I&M Funktion 4	TF6271 und EL663x

PROFINET RT Controller

Bausteine PROFINET RT Controller	Bedeutung	Beschreibung
FB_PN_ReadStateOfDevices [▶ 21]	Auslesen der Anzahl der konfigurierten, fehlenden Teilnehmer und die mit einem Diagnoseinformation	TF6271, CCAT-M930 und EL663x
FB_PN_ReadCompleteInfoOfDevices [▶ 22]	Erstellt eine Liste mit den vollständigen Informationen der konfigurierten Teilnehmer	TF6271, CCAT-M930 und EL663x

Device

Device

Bausteine Device	Bedeutung	Beschreibung
FB_PN_SEND_ALARM [▶ 35]	Sendet an den Controller einen Alarm	TF6270

Device via CCAT (CX-B930, TF6270)

Bausteine Device via CCAT	Bedeutung	Beschreibung
FB_PROFINET_READ_IM [▶ 30]	Liefert die I&M-Daten des jeweiligen Gerätes	TF6270 und CCAT-B930
FB_PROFINET_READ_NAME [▶ 31]	Liefert den PROFINET-Namen des jeweiligen PROFINET Device	TF6270 und CCAT-B930
FB_PROFINET_READ_PRM [▶ 32]	Scannt das PROFINET Netzwerk und liefert die Anzahl sowie eine Liste mit den gefundenen PROFINET Devices	TF6270 und CCAT-B930
FB_PROFINET_WRITE_IM [▶ 33]	Ermöglicht das Beschreiben der I&M-Daten des jeweiligen PROFINET Device	TF6270 und CCAT-B930
FB_PROFINET_SET_NAME [▶ 34]	Ermöglicht das Verändern des PROFINET-Namens des jeweiligen PROFINET Device	TF6270 und CCAT-B930

EL6631-0010

Bausteine EL6631-0010	Bedeutung	Beschreibung
FB_READ_PROFINET_NAME [▶ 27]	Liest den PROFINET Namen einer EL6631-0010 und falls konfiguriert einer virtuellen EL6631-0010 aus	EL6631-0010
FB_Write_IuM_EL6631_0010 [▶ 28]	Schreibt I&M1, I&M2, I&M3 und I&M4 Daten auf das PROFINET-Device.	EL6631-0010
FB_Read_IuM_EL6631_0010 [▶ 29]	Liest I&M1, I&M2, I&M3 und I&M4 Daten aus dem PROFINET-Device aus.	EL6631-0010

Port Diagnose

Statistik und Diagnose Informationen

Bausteine Statistik und Diagnose Informationen	Bedeutung	Beschreibung
FB_PN_GET_PORT_STATISTIC [▶ 36]	Lesen der Port Statistik	TF6271 und EL663x
FB_PN_READ_PORT_DIAG [▶ 37]	Lesen der Port Diagnose	TF6271 und EL663x

3.1 Controller

3.1.1 AlarmDiag

3.1.1.1 FB_PN_ALARM_DIAG



Mit dem Funktionsbaustein FB_PN_ALARM_DIAG können Diagnose Alarme 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 Diagnose Alarme aus dem Puffer gelesen wurden.

 **Eingänge**

```
VAR_INPUT
  bEnable : BOOL;
  NETID   : T_AmsNetId;
  PORT    : T_AmsPort;
END_VAR
```

Name	Typ	Beschreibung
bEnable	BOOL	Aktivierung des Bausteins.
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T_AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).

 **Ausgänge**

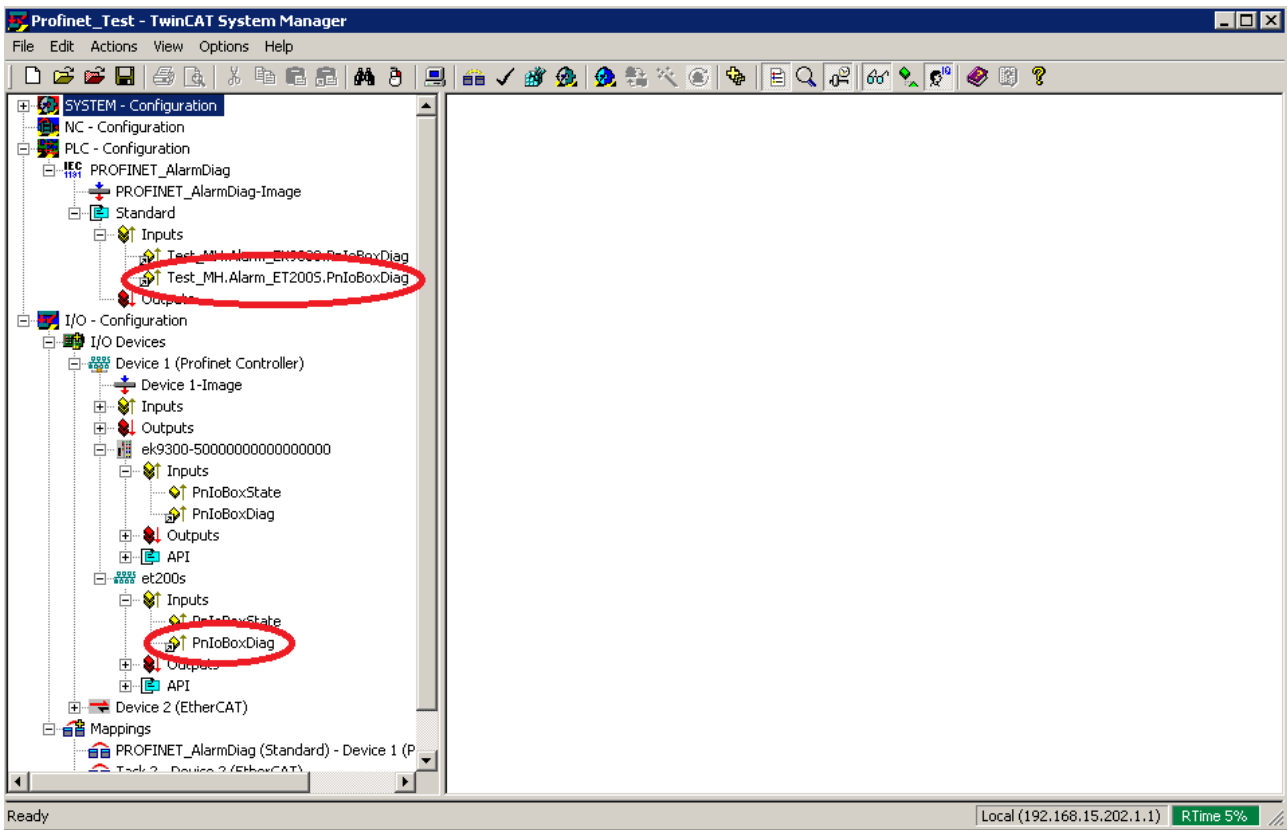
```
VAR_OUTPUT
  bBusy           : BOOL;
  stAlarmDiagData : ST_PN_AlarmDiagData;
  bError          : BOOL;
  iErrorID        : UDINT;
  iNrAlarms       : INT;
END_VAR
```

Name	Typ	Beschreibung
bBusy	BOOL	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	ST_PN_AlarmDiagData ▶ 42	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
iErrorID	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS-Fehlernummer</u> ▶ 45 .
iNrAlarms	INT	Anzahl der zuletzt ausgelesenen Alarme.

VAR

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

Name	Typ	Beschreibung
PnIoBoxDiag	WORD	Hardware-Eingang. Diese Variable ist mit dem PROFINET-Gerät zu verknüpfen. Über eine Statusänderung dieser Variable wird dem SPS-Programm mitgeteilt, dass neue Alarm Diagnosen in dem verknüpften PROFINET-Gerät anstehen.



Voraussetzungen

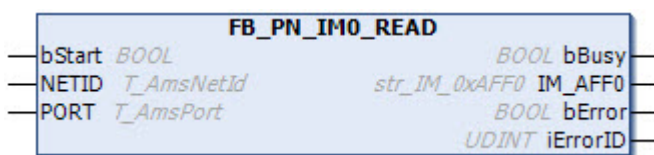
Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.1.2 Identification and Maintenance

I&M (Identification & Maintenance)

Bausteine I&M Funktionen	Bedeutung	Beschreibung
FB_PN_IM0_READ [► 12]	Lesen der I&M Funktion 0	TF6271 und EL663x
FB_PN_IM1_READ [► 13]	Lesen der I&M Funktion 1	TF6271 und EL663x
FB_PN_IM2_READ [► 15]	Lesen der I&M Funktion 2	TF6271 und EL663x
FB_PN_IM3_READ [► 17]	Lesen der I&M Funktion 3	TF6271 und EL663x
FB_PN_IM4_READ [► 19]	Lesen der I&M Funktion 4	TF6271 und EL663x
FB_PN_IM1_WRITE [► 14]	Schreiben der I&M Funktion 1	TF6271 und EL663x
FB_PN_IM2_WRITE [► 16]	Schreiben der I&M Funktion 2	TF6271 und EL663x
FB_PN_IM3_WRITE [► 18]	Schreiben der I&M Funktion 3	TF6271 und EL663x
FB_PN_IM4_WRITE [► 20]	Schreiben der I&M Funktion [► 4	TF6271 und EL663x

3.1.2.1 FB_PN_IM0_READ



Mit dem Funktionsbaustein FB_PN_Im0_READ 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 nach PROFINET Standard.

Eingänge

```
VAR_INPUT
    bStart : BOOL;
    NETID : T_AmsNetId;
    PORT : T_AmsPort;
END_VAR
```

Name	Typ	Beschreibung
bStart	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T_AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).

Ausgänge

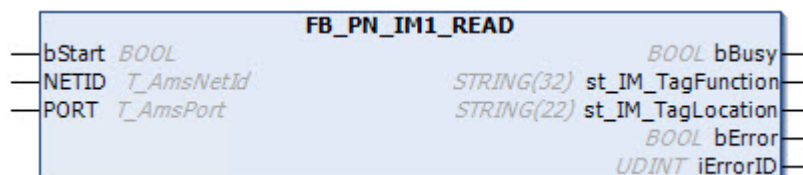
```
VAR_OUTPUT
    bBusy : BOOL;
    IM_AFF0 : str_IM_0xAFF0;
    bError : BOOL;
    iErrorID : UDINT;
END_VAR
```

Name	Typ	Beschreibung
bBusy	BOOL	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	str_IM_0xAFF0 [▶ 39]	IM_AFF0: Ausgabe des vom Gerät gelieferten I&M0 Frames in einer Struktur.
bError	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
iErrorID	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS</u> -Fehlernummer [▶ 45].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.1.2.2 FB_PN_IM1_READ



Mit dem Funktionsbaustein FB_PN_IM1_READ 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 nach PROFINET Standard.

Eingänge

```
VAR_INPUT
  bStart      : BOOL;
  NETID       : T_AmsNetId;
  PORT        : T_AmsPort;
END_VAR
```

Name	Typ	Beschreibung
bStart	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T_AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).

Ausgänge

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

Name	Typ	Beschreibung
bBusy	BOOL	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	STRING	Ausgelesenes Label für Function des Gerätes.
st_IM_TagLocation	STRING	Ausgelesenes Label für Einbauort des Gerätes.
bError	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
iErrorID	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS</u> -Fehlernummer [► 45].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.1.2.3 FB_PN_IM1_WRITE



Mit dem Funktionsbaustein FB_PN_IM1_WRITE 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 nach PROFINET Standard.

Eingänge

```
VAR_INPUT
  bStart      : BOOL;
  NETID       : T_AmsNetId;
  PORT        : T_AmsPort;
```

```

st_IM_TagFunction : STRING(32);
st_IM_TagLocation : STRING(22);
END_VAR
    
```

Name	Typ	Beschreibung
bStart	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T_AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).
st_IM_TagFunction	STRING	Mit diesem String wird die Funktionsbeschreibung auf dem Gerät abgelegt.
st_IM_TagLocation	STRING	Mit diesem String wird der Einbauort auf dem Gerät abgelegt.

Ausgänge

```

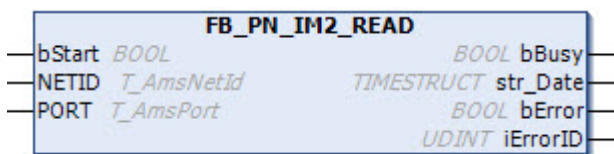
VAR_OUTPUT
bBusy : BOOL;
bError : BOOL;
iErrorID : UDINT;
END_VAR
    
```

Name	Typ	Beschreibung
bBusy	BOOL	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
iErrorID	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS</u> -Fehlernummer [► 45].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.1.2.4 FB_PN_IM2_READ



Mit dem Funktionsbaustein FB_PN_IM2_READ 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 nach PROFINET Standard.

Eingänge

```

VAR_INPUT
bStart : BOOL;
NETID : T_AmsNetId;
PORT : T_AmsPort;
END_VAR
    
```

Name	Typ	Beschreibung
bStart	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T_AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).

Ausgänge

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

Name	Typ	Beschreibung
bBusy	BOOL	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	TIMESTRUCT	Liefert das Datum des Geräteeinbaus in dem Format < YYYY-MM-DD HH:MM > zurück.
bError	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
iErrorID	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS-Fehlernummer</u> [► 45].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.1.2.5 FB_PN_IM2_WRITE



Mit dem Funktionsbaustein FB_PN_IM2_WRITE 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 nach PROFINET Standard.

Eingänge

```
VAR_INPUT
  bStart : BOOL;
  NETID  : T_AmsNetId;
  PORT   : T_AmsPort;
  str_Date : TIMESTRUCT; (*YYYY-MM-DD HH:MM*)
END_VAR
```


Name	Typ	Beschreibung
bStart	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T_AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).
str_Date	TIMESTRUCT	Schreibt ein Datum (z. B. Einbaudatum des Gerätes) in dem Format < YYYY-MM-DD HH:MM > auf das Gerät.

Ausgänge

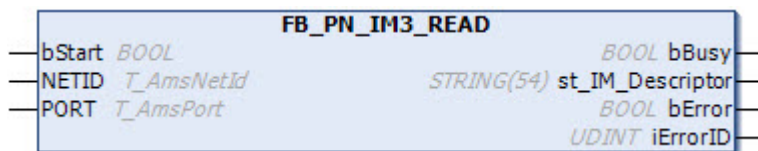
```
VAR_OUTPUT
  bBusy      : BOOL;
  bError     : BOOL;
  iErrorID   : UDINT;
END_VAR
```

Name	Typ	Beschreibung
bBusy	BOOL	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
iErrorID	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS-Fehlernummer</u> [► 45].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.1.2.6 FB_PN_IM3_READ



Mit dem Funktionsbaustein FB_PN_IM3_READ 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 nach PROFINET Standard.

Eingänge

```
VAR_INPUT
  bStart : BOOL;
  NETID  : T_AmsNetId;
  PORT   : T_AmsPort;
END_VAR
```

Name	Typ	Beschreibung
bStart	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T_AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).

Ausgänge

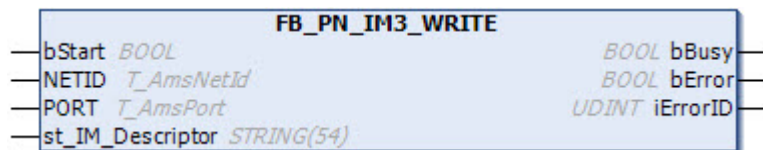
```
VAR_OUTPUT
  bBusy          : BOOL;
  st_IM_Descriptor : STRING(54);
  bError         : BOOL;
  iErrorID       : UDINT;
END_VAR
```

Name	Typ	Beschreibung
bBusy	BOOL	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	STRING	Liefert die für das Gerät hinterlegte Herstellerbeschreibung zurück.
bError	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
iErrorID	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS-</u> Fehlernummer [► 45].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.1.2.7 FB_PN_IM3_WRITE



Mit dem Funktionsbaustein FB_PN_IM3_WRITE 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 nach PROFINET Standard.

Eingänge

```
VAR_INPUT
  bStart          : BOOL;
  NETID           : T_AmsNetId;
  PORT            : T_AmsPort;
  st_IM_Descriptor : STRING(54);
END_VAR
```

Name	Typ	Beschreibung
bStart	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T_AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).
str_Date	TIMESTRUCT	Schreibt ein Datum (z. B. Einbaudatum des Gerätes) in dem Format < YYYY-MM-DD HH:MM > auf das Gerät.

Ausgänge

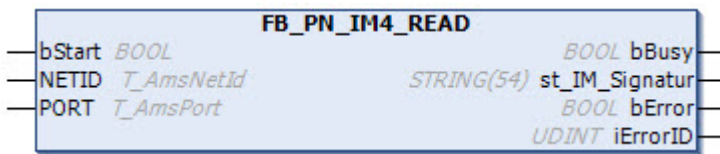
```
VAR_OUTPUT
  bBusy          : BOOL;
  bError         : BOOL;
  iErrorID       : UDINT;
END_VAR
```

Name	Typ	Beschreibung
bBusy	BOOL	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
iErrorID	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS</u> -Fehlernummer [► 45].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.1.2.8 FB_PN_IM4_Read



Mit dem Funktionsbaustein FB_PN_IM4_READ 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 nach PROFINET Standard.

Eingänge

```
VAR_INPUT
  bStart : BOOL;
  NETID  : T_AmsNetId;
  PORT   : T_AmsPort;
END_VAR
```

Name	Typ	Beschreibung
bStart	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T_AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).

Ausgänge

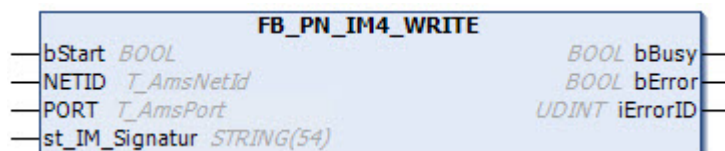
```
VAR_OUTPUT
  bBusy          : BOOL;
  st_IM_Signatur : STRING(54);
  bError         : BOOL;
  iErrorID       : UDINT;
END_VAR
```

Name	Typ	Beschreibung
bBusy	BOOL	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	STRING	Liefert die für das Gerät hinterlegte Herstellersignatur zurück.
bError	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
iErrorID	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS</u> -Fehlernummer [► 45].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.1.2.9 FB_PN_IM4_WRITE



Mit dem Funktionsbaustein FB_PN_IM4_WRITE 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 nach PROFINET Standard.

Eingänge

```
VAR_INPUT
    bStart      : BOOL;
    NETID       : T_AmsNetId;
    PORT        : T_AmsPort;
    st_IM_Signatur : STRING(54);
END_VAR
```

Name	Typ	Beschreibung
bStart	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T_AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).
st_IM_Signatur	STRING	Signatur des Herstellers, die auf das Gerät geschrieben wird.

Ausgänge

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

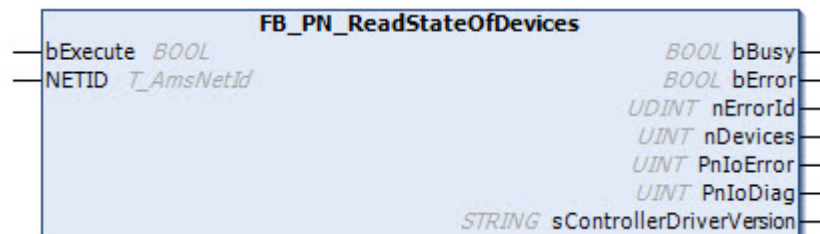
Name	Typ	Beschreibung
bBusy	BOOL	Bei der Aktivierung des Funktionsbausteins wird dieser Ausgang gesetzt und bleibt gesetzt, bis eine Rückmeldung erfolgt. Während Busy = TRUE wird an den Eingängen kein neuer Befehl angenommen.
bError	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
iErrorID	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS-Fehlernummer</u> [▶ 45].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.1.3 PROFINET RT Controller

3.1.3.1 FB_PN_ReadStateOfDevices



Der Funktionsbaustein FB_PN_ReadDtdateOfDevices liefert bei Aufruf die Anzahl der konfigurierten, der fehlenden Teilnehmer und derer mit Diagnoseinformationen.

PROFINET RT Controller Treiber Version

i Der Funktionsbaustein ist nur für PROFINET RT Controller Treiber Version v03(v0.21) oder höher verfügbar.
Im TwinCAT XAE lässt sich die Treiber Version auch beim IO Gerät PROFINET RT Controller finden.

Eingänge

```
VAR_INPUT
  bExecute          : BOOL;
  NETID             : T_AmsNetIdArr;
END_VAR
```

Name	Typ	Beschreibung
bExecute	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des PROFINET RT Controllers

Ausgänge

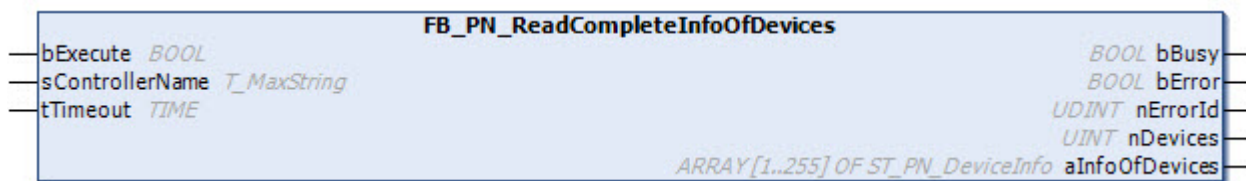
```
VAR_OUTPUT
  bBusy             : BOOL;
  bError            : BOOL;
  nErrorID          : UDINT;
  nDevices          : UINT;
  PnIoError         : UINT;
  PnIoDiag          : UINT;
  sControllerDriverVersion : STRING;
END_VAR
```

Name	Typ	Beschreibung
bBusy	BOOL	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
nErrorId	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS-Fehlernummer</u> [► 45].
nDevices	UINT	Anzahl der konfigurierten Geräte.(max.255)
PnIoError	UINT	Anzahl der Geräte mit Fehlerzustand oder Diagnose.
PnIoDiag	UINT	Anzahl der Geräte mit Diagnose.
sControllerDriverVersion	UINT	PROFINET Controller Version 03 (V00.21) oder höher notwendig.

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4024.54	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag (>= v1.4.1.0)

3.1.3.2 FB_PN_ReadCompleteInfoOfDevices



Der Funktionsbaustein FB_PN_ReadCompleteInfoOfDevices erstellt eine Liste mit den vollständigen Informationen der konfigurierten Teilnehmer (in TwinCAT). Für PROFINET RT Controller wie TF6271 oder CCAT M930 Schnittstelle oder EL6631 v11(v.024).

Eingänge

```
VAR_INPUT
  bExecute          : BOOL;
  sControllerName  : T_Maxstring
  tTimeout         : TIME;
END_VAR
```

Name	Typ	Beschreibung
bExecute	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
sControllerName	T_MaxString	Name des PROFINET RTControllers im TwinCAT Baum.
tTimeout	TIME	Timeout für den kompletten Lesevorgang

Ausgänge

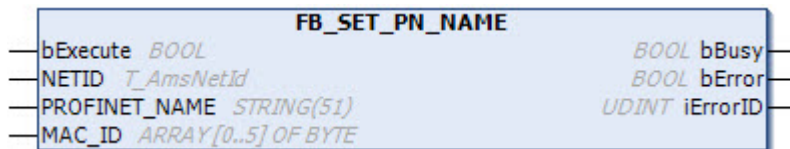
```
VAR_OUTPUT
  bBusy            : BOOL;
  bError           : BOOL;
  nErrorID        : UDINT;
  nDevices         : UINT;
  aInfoOfDevices  : ARRAY [1..255] OF st_PN_DeviceInfo;
END_VAR
```

Name	Typ	Beschreibung
bBusy	BOOL	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der bBusy-Ausgang zurückgesetzt wurde.
nErrorId	UDINT	Liefert bei einem gesetzten bError-Ausgang eine <u>ADS-Fehlernummer</u> [► 45].
nDevices	UINT	Anzahl der PROFINET-Geräte in der Konfiguration.
alInfoOfDevices	ST_PN_DeviceInfo	Einstellungen Informationen der konfigurierten PROFINET-Geräte.

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4024.54	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag (>= v1.4.1.0)

3.1.4 FB_SET_PN_NAME



Der Funktionsbaustein FB_SET_PN_NAME vergibt dem jeweiligen PROFINET Device einen Namen. Bei der Vergabe ist drauf zu achten, dass nur PROFINET-konforme Zeichen verwendet werden.

Eingänge

```
VAR_INPUT
  bExecute      : BOOL;
  NETID         : T_AmsNetId;
  PROFINET_NAME : STRING(51);
  MAC_ID        : ARRAY [0..5] OF BYTE;
END_VAR
```

Name	Typ	Beschreibung
bExecute	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers
PROFINET_NAME	STRING	Name welcher an das PROFINET Device vergeben werden soll. Max. 240 Zeichen und folgende Zeichen erlaubt „a..z“, „0..9“, „.“, „-“
MAC_ID	BYTE	MAC ID des Controllers

Ausgänge

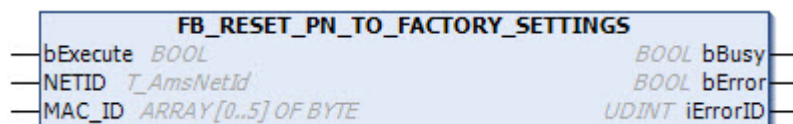
```
VAR_OUTPUT
  bBusy      : BOOL;
  bError     : BOOL;
  iErrorID   : UDINT;
END_VAR
```

Name	Typ	Beschreibung
bBusy	BOOL	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
iErrorID	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS-Fehlernummer</u> [► 45].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.1.5 FB_RESET_PN_TO_FACTORY_SETTINGS



Der Funktionsbaustein FB_RESET_PN_TO_FACTORY setzt das jeweilige PROFINET Device auf seine Herstellereinstellungen zurück.

Eingänge

```
VAR_INPUT
  bExecute      : BOOL;
  NETID         : T_AmsNetId;
  MAC_ID       : ARRAY [0..5] OF BYTE;
END_VAR
```

Name	Typ	Beschreibung
bExecute	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers
MAC_ID	BYTE	MAC ID des Controllers

Ausgänge

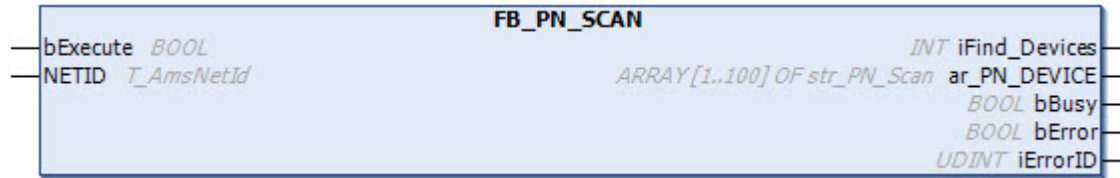
```
VAR_OUTPUT
  bBusy        : BOOL;
  bError       : BOOL;
  iErrorID     : UDINT;
END_VAR
```

Name	Typ	Beschreibung
bBusy	BOOL	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
iErrorID	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS-Fehlernummer</u> [► 45].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.1.6 FB_PN_SCAN



Der Funktionsbaustein FB_PN_SCAN scannt das PROFINET Netzwerk und liefert die Anzahl sowie eine Liste mit Informationen zu den gefundenen PROFINET Devices.

Eingänge

```
VAR_INPUT
  bExecute      : BOOL;
  NETID         : T_AmsNetId;
END_VAR
```

Name	Typ	Beschreibung
bExecute	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers

Ausgänge

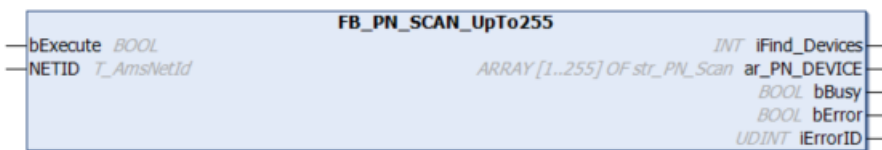
```
VAR_OUTPUT
  iFind_Devices : INT;
  ar_PN_DEVICE  : ARRAY [1..100] OF str_PN_SCAN;
  bBusy         : BOOL;
  bError        : BOOL;
  iErrorID      : UDINT;
END_VAR
```

Name	Typ	Beschreibung
iFind_Devices	int	Anzahl der PROFINET-Geräte in der Konfiguration.
ar_PN_DEVICE	str_PN_SCAN [▶ 40]	PROFINET-/IP-Einstellungen der PROFINET-Geräte.
bBusy	BOOL	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
iErrorID	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS-Fehlernummer</u> [▶ 45].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.1.7 FB_PN_SCAN_UpTo255



Der Funktionsbaustein FB_PN_SCAN_UpTo255 scannt das PROFINET Netzwerk und liefert die Anzahl sowie eine Liste mit Informationen zu den gefundenen PROFINET Devices.

Eingänge

```
VAR_INPUT
  bExecute      : BOOL;
  NETID         : T_AmsNetId;
END_VAR
```

Name	Typ	Beschreibung
bExecute	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers

Ausgänge

```
VAR_OUTPUT
  iFind_Devices : INT;
  ar_PN_DEVICE  : ARRAY [1..255] OF str_PN_SCAN;
  bBusy         : BOOL;
  bError        : BOOL;
  iErrorID      : UDINT;
END_VAR
```

Name	Typ	Beschreibung
iFind_Devices	int	Anzahl der PROFINET-Geräte in der Konfiguration.
ar_PN_DEVICE	str_PN_SCAN [▶ 40]	PROFINET-/IP-Einstellungen der PROFINET-Geräte.
bBusy	BOOL	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
iErrorID	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS-Fehlernummer</u> [▶ 45].

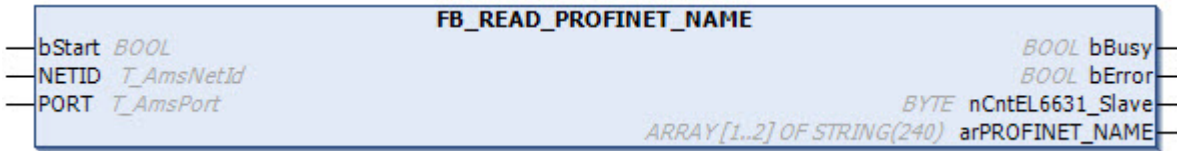
Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4024.57	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag (>= v1.5.2.0)

3.2 Device

3.2.1 EL6631-0010

3.2.1.1 FB_READ_PROFINET_NAME



Der Funktionsbaustein FB_READ_PROFINET_NAME liest den PROFINET Namen einer EL6631-0010 und falls konfiguriert einer virtuellen EL6631-0010 aus.

Eingänge

```
VAR_INPUT
  bStart   : BOOL;
  NETID    : T_AmsNetId;
  PORT     : T_AmsPort;
END_VAR
```

Name	Typ	Beschreibung
bStart	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T_AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).

Ausgänge

```
VAR_OUTPUT
  bBusy           : BOOL;
  bError          : BOOL;
  nCntEL6631_Slave : BYTE
  arPROFINET_NAME : ARRAY [1..2] OF STRING(240)
END_VAR
```

Name	Typ	Beschreibung
bBusy	BOOL	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
nCntEL6631_Slave	BYTE	Liefert die Informationen, wie sich die EL6631-0010 darstellt. „0“ = EL6631-0010 und „1“ = virtuelle EL6631-0010.
arPROFINET_NAME	STRING	Enthält die PROFINET Namen der EL6631-0010 und falls konfiguriert den der virtuellen EL6631-0010.

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.2.1.2 FB_Write_IuM_EL6631_0010



Der Funktionsbaustein FB_Write_IuM_EL6631_0010 schreibt I&M1, I&M2, I&M3 und I&M4 (Identification & Maintenance) Daten nach PROFINET-Spezifikation über EtherCAT als String auf das PROFINET-Gerät.

Eingänge

```
VAR_INPUT
  bWrite      : BOOL;
  NETID       : T_AmsNetId;
  PORT        : T_AmsPort;
  byState     : BYTE;
  iNumber     : INT:=0;
  st_IM_TagFunction : STRING;
  st_IM_TagLocation : STRING;
  st_IM_Date  : STRING;
  st_IM_Descriptor : STRING;
  st_IM_Signature : STRING;
END_VAR
```

Name	Typ	Beschreibung
bWrite	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert und die I&M Daten in das ausgewählte ProfiNET-Gerät geschrieben.
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T-AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).
byState	BYTE	Über dieses Byte kann ausgewählt werden, welche I&M Daten geschrieben werden sollen. Bit 0 -> I&M1 Bit 1 -> I&M2 Bit 2 -> I&M3 Bit 3 -> I&M4
iNumber	INT	Mit einer Klemme können 2 ProfiNET-Geräte abgebildet werden. Über iNumber ("0" oder "1") wird das Gerät ausgewählt, für das die I&M Daten geschrieben werden sollen.
st_IM_TagFunction	STRING	Label für die Funktion des Gerätes wird auf das Gerät geschrieben. I&M1 byState.0=TRUE
st_IM_Taglocation	STRING	Label für den Einbauort des Gerätes wird auf das Gerät geschrieben. I&M1 byState.0=TRUE
st_IM_Date	STRING	Datum des Geräteeinbaus wird auf des Gerät geschrieben. I&M2 byState.1=TRUE
st_IM_Descriptor	STRING	Beschreibung des Herstellers wird auf das Gerät geschrieben. I&M3 byState.2=TRUE
st_IM_Signature	STRING	Signatur des Herstellers wird auf das Gerät geschrieben. I&M4 byState.3=TRUE

Ausgänge

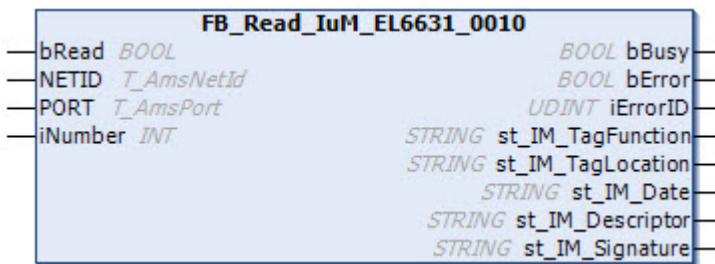
```
VAR_OUTPUT
  bBusy          : BOOL;
  bError         : BOOL;
  iErrorID       : UDINT;
END_VAR
```

Name	Typ	Beschreibung
bBusy	BOOL	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
iErrorID	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS-Fehlernummer</u> [► 45].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.2.1.3 FB_Read_IuM_EL6631_0010



Der Funktionsbaustein FB_Read_IuM_EL6631_0010 liest I&M1, I&M2, I&M3 und I&M4 (Identification & Maintenance) Daten über EtherCAT als String aus einem PROFINET-Gerät. Das Auslesen der I&M0 Daten erfolgt bei einem PROFINET-Gerät mit EtherCAT über CoE (CAN over EtherCAT).

Eingänge

```
VAR_INPUT
  bRead          : BOOL;;
  NETID          : T_AmsNetId;
  PORT           : T_AmsPort;
  iNumber        : INT:=0;
END_VAR
```

Name	Typ	Beschreibung
bRead	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert und die I&M Daten aus dem PROFINET-Gerät ausgelesen.
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T-AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).
iNumber	INT	Mit einer Klemme können 2 PROFINET-Geräte abgebildet werden. Über iNumber ("0" oder "1") wird das Gerät ausgewählt, für das die I&M Daten ausgelesen werden sollen.

Ausgänge

```
VAR_OUTPUT
  bBusy          : BOOL;
  bError         : BOOL;
  iErrorID       : UDINT;
  st_IM_TagFunction : STRING; (* I&M1 *)
  st_IM_TagLocation : STRING; (* I&M1 *)
  st_IM_Date      : STRING; (* I&M2 *)
  st_IM_Descriptor : STRING; (* I&M3 *)
  st_IM_Signature  : STRING; (* I&M4 *)
END_VAR
```

Name	Typ	Beschreibung
bBusy	BOOL	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
iErrorID	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS-Fehlernummer</u> [► 45].
st_IM_TagFunction	STRING	Ausgelesenes Label für Funktion des Gerätes. I&M1
st_IM_Taglocation	STRING	Ausgelesenes Label für den Einbauort des Gerätes. I&M1
st_IM_Date	STRING	Liefert das Datum des Geräteeinbaus in dem Format. I&M2
st_IM_Descriptor	STRING	Liefert die für das Gerät hinterlegte Herstellerbeschreibung zurück. I&M3
st_IM_Signature	STRING	Liefert die für das Gerät hinterlegte Herstellersignatur zurück. I&M4

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.2.2 Device via CCAT (CX-B930,TF6270)

3.2.2.1 FB_PROFINET_READ_IM



Der Funktionsbaustein FB_PROFINET_READ_IM liefert die I&M (Identification & Maintenance)-Daten des jeweiligen PROFINET Device.

Eingänge

```
VAR_INPUT
  bStart : BOOL;
  NETID  : T_AmsNetId;
  PORT   : T_AmsPort;
END_VAR
```

Name	Typ	Beschreibung
bStart	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T_AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).

Ausgänge

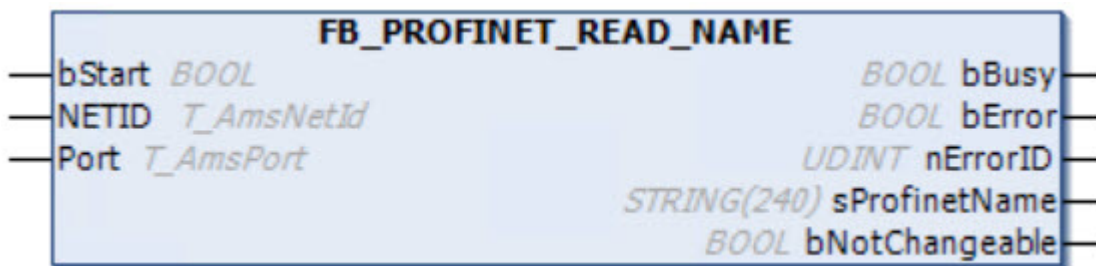
```
VAR_OUTPUT
  str_IuM_Data      : str_IuM_Data
  bBusy             : BOOL;
  bError            : BOOL;
  nErrorID          : UDINT;
END_VAR
```

Name	Typ	Beschreibung
str_IuM_Data	str_IuM_Data [► 42]	Strukturelement mit I&M-Daten
bBusy	BOOL	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
nErrorId	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS-Fehlernummer</u> [► 45].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.2.2.2 FB_PROFINET_READ_NAME



Der Funktionsbaustein FB_PROFINET_READ_NAME liefert den PROFINET-Namen des jeweiligen PROFINET Device und die Information, ob der PROFINET-Name veränderbar ist seitens des PROFINET Controller.

Eingänge

```
VAR_INPUT
  bStart : BOOL;
  NETID  : T_AmsNetId;
  PORT   : T_AmsPort;
END_VAR
```

Name	Typ	Beschreibung
bStart	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des PROFINET Device
PORT	T_AmsPort	ADS Port Nummer des PROFINET Device; Default = 0xFFFF

Ausgänge

```
VAR_OUTPUT
  bBusy          : BOOL;
  bError         : BOOL;
  nErrorID       : UDINT;
  sProfinetName  : STRING(240);
  bNotChangeable : BOOL;
END_VAR
```

Name	Typ	Beschreibung
bBusy	BOOL	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
nErrorId	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS-Fehlernummer</u> [► 45].
sProfinetName	STRING(240)	PROFINET Device Name. Max. 240 Zeichen und folgende Zeichen erlaubt „a..z“, „0..9“, „.“, „-“
bNotChangeable	BOOL	Wenn „TRUE“, kann der PROFINET Controller den PROFINET Namen des Device nicht ändern.

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4024.55	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag (>= v1.5.1.0)

3.2.2.3 FB_PROFINET_READ_PRM



Der Funktionsbaustein FB_PROFINET_READ_PRM scannt das PROFINET Netzwerk und liefert die Anzahl sowie eine Liste mit Informationen zu den gefundenen PROFINET Devices.

Eingänge

```
VAR_INPUT
  NETID : T_AmsNetId;
  PORT  : T_AmsPort;
  bStart : BOOL;
END_VAR
```


Name	Typ	Beschreibung
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T-AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).
bStart	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.

Ausgänge

```
VAR_OUTPUT
  str_Diag_PN_Settings : str_Diag_PN_Settings;
  bBusy                : BOOL;
  bError               : BOOL;
  nErrorID             : UDINT;
END_VAR
```

Name	Typ	Beschreibung
str_Diag_PN_Settings	str_Diag_PN_Settings [▶ 42]	PROFINET-/IP-Einstellungen
bBusy	BOOL	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
nErrorId	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS-Fehlernummer</u> [▶ 45].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.2.2.4 FB_PROFINET_WRITE_IM



Mit dem Funktionsbaustein FB_PROFINET_WRITE_IM können die I&M(Identification & Maintenance)-Daten des jeweiligen PROFINET Device beschrieben werden.

Eingänge

```
VAR_INPUT
  bStart      : BOOL;
  NETID       : T_AmsNetId;
  PORT        : T_AmsPort;
  str_IuM_Data : str_IuM_Data
END_VAR
```

Name	Typ	Beschreibung
bStart	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T_AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).
str_luM_Data	str_luM_Data [► 42]	Strukturelement mit I&M-Daten

Ausgänge

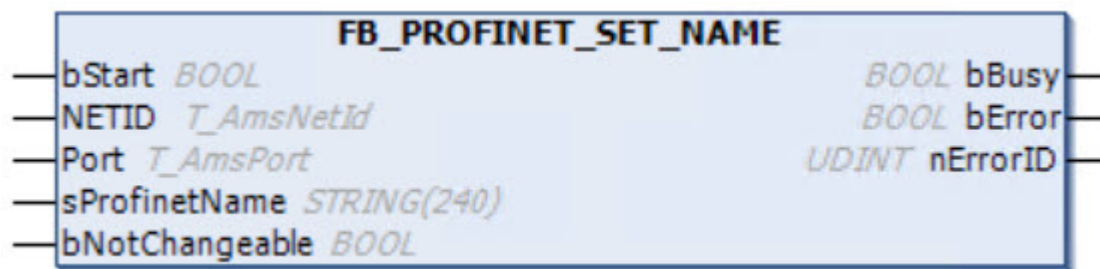
```
VAR_OUTPUT
  bBusy          : BOOL;
  bError         : BOOL;
  nErrorID       : UDINT;
END_VAR
```

Name	Typ	Beschreibung
bBusy	BOOL	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
nErrorId	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS</u> -Fehlernummer [► 45].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.2.2.5 FB_PROFINET_SET_NAME



Mit dem Funktionsbaustein FB_PROFINET_SET_NAME kann der PROFINET-Name des jeweiligen PROFINET Device verändert werden. PROFINET Treiber Version 06 (V00.34) oder höher, TF6270, CCAT PN Interface(B930) erforderlich

Eingänge

```
VAR_INPUT
  bStart          : BOOL;
  NETID           : T_AmsNetId;
  PORT            : T_AmsPort;
  sProfinetName   : STRING(240);
  bNotChangeable : BOOL;
END_VAR
```

Name	Typ	Beschreibung
bStart	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des PROFINET Device
PORT	T_AmsPort	ADS Port Nummer des PROFINET Device; default = 0xFFFF
sProfinetName	STRING(240)	PROFINET Device Name. Max. 240 Zeichen und folgende Zeichen erlaubt „a..z“, „0..9“, „.“, „-“
bNotChangeable	BOOL	Wenn „TRUE“, kann der PROFINET Controller den PROFINET Namen nicht ändern.

Ausgänge

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

Name	Typ	Beschreibung
bBusy	BOOL	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
nErrorId	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS-Fehlernummer</u> [► 45].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4024.55	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag (>= v1.5.1.0)

3.2.3 FB_PN_SEND_ALARM



Der Funktionsbaustein FB_PN_SEND_ALARM sendet an den Controller einen Alarm.

Eingänge

```
VAR_INPUT
  bStart          : BOOL;
  NETID          : T_AmsNetId;
  PORT           : T_AmsPort;
  PN_ALARM_Typ  : E_PN_ALARM_TYP;
  PN_slotNumber : WORD;
  PN_SubSlotNumber : WORD;
END_VAR
```

Name	Typ	Beschreibung
bStart	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T_AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).
PN_ALARM_Typ	E_PN_ALARM_TYP [▶ 43]	Enthält vordefinierte Alarmtypen
PN_SlotNumber	WORD	Slot Nummer
PN_SubSlotNumber	WORD	SubSlot Nummer

Ausgänge

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

Name	Typ	Beschreibung
bBusy	BOOL	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	BOOL	Sollte ein Fehler bei der Übertragung des Kommandos erfolgen, dann wird dieser Ausgang gesetzt, nachdem der <i>bBusy</i> -Ausgang zurückgesetzt wurde.
iErrorID	UDINT	Liefert bei einem gesetzten <i>bError</i> -Ausgang eine <u>ADS-Fehlernummer</u> [▶ 45].

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.3 FB_PN_GET_PORT_STATISTIC



Der Funktionsbaustein FB_PN_GET_PORT_STATISTIC liefert bei Aufruf die statistischen Daten zu den Ports eines PROFINET-Gerätes.

Eingänge

```
VAR_INPUT
  bStart      : BOOL;
  NETID       : T_AmsNetId;
  PORT        : T_AmsPort;
END_VAR
```

Name	Typ	Beschreibung
bStart	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T_AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).

Ausgänge

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

Name	Typ	Beschreibung
bBusy	BOOL	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	str_GetPortStatistic [▶ 43]	Diese Struktur enthält die statistischen Daten zu Port 1 .
str_RemotePort_2	str_GetPortStatistic [▶ 43]	Diese Struktur enthält die statistischen Daten zu Port 2.
bPort1	BOOL	Ist TRUE, wenn der Port einen Link hat.
bPort2	BOOL	Ist TRUE, wenn der Port einen Link hat.

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

3.4 FB_PN_READ_PORT_DIAG



Der Funktionsbaustein FB_PN_READ_PORT_DIAG ruft die Diagnose Informationen der Ports eines PROFINET-Gerätes ab.

Eingänge

```
VAR_INPUT
  bStart      : BOOL;
  NETID       : T_AmsNetId;
  PORT        : T_AmsPort;
END_VAR
```

Name	Typ	Beschreibung
bStart	BOOL	Über eine positive Flanke an diesem Eingang wird der Baustein aktiviert.
NETID	T_AmsNetId	AMS Net ID des Controllers
PORT	T_AmsPort	Port, über den der Controller mit dem Gerät kommuniziert (Port = Device ID + 1000hex).

Ausgänge

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

Name	Typ	Beschreibung
bBusy	BOOL	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	str_GetPortStatistic [► 43]	Diese Struktur enthält die statistischen Daten zu Port 1 .
str_RemotePort_2	str_GetPortStatistic [► 43]	Diese Struktur enthält die statistischen Daten zu Port 2.

Voraussetzungen

Entwicklungsumgebung	Zielplattform	Einzubindende SPS Bibliotheken
TwinCAT v3.1.4018	PC or CX (x64, x86, ARM)	Tc2_ProfinetDiag

4 Datenstrukturen

4.1 Controller

4.1.1 IM

4.1.1.1 str_SW_Rec

Die Datenstruktur **str_SW_REC** enthält die Software-Version des PROFINET-Gerätes.

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

Name	Beschreibung
cSWRevPrefix	Revision Prefix (V = officially released Version, R = Revision, P = Prototype, U = Under Test/Field Test, T = Test Device)
nSWRevFunctionEnhance	Funktionserweiterung
nSWRevBugFix	Bug Fix
nSWRevIntCha	Internal Change

4.1.1.2 str_IM_0xAFF0

Die Datenstruktur **str_IM_0xAFF0** bildet die Struktur des I&M0 Frames in der PLC ab. Diese Struktur wird zum Lesen von einem PROFINET-Gerät verwendet und enthält Herstellerinformationen.

```
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
```

4.1.1.3 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
```

4.1.1.4 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

```

4.1.1.5 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

```

4.1.1.6 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

```

4.1.2 str_PN_Scan

Die Datenstruktur **str_PN_Scan** enthält Informationen über den PROFINET-Teilnehmer.

```

TYPE str_PN_Scan :
STRUCT
  IP_Addr      : ARRAY [0..3] OF BYTE;
  SubNetMask   : ARRAY [0..3] OF BYTE;
  DefaultGateway : ARRAY [0..3] OF BYTE;
  MacID        : ARRAY [0..5] OF BYTE;
  VendorID     : WORD;
  DeviceID     : WORD;
  PN_NAME      : STRING(51);
END_STRUCT
END_TYPE

```

Name	Beschreibung
IP_Addr	IP Adresse
SubNetMask	Subnetzmaske
DefaultGateway	Default Gateway
MacId	MacId
VendorID	Hersteller ID
DeviceID	Geräte ID
PN_NAME	PROFINET Name des PROFINET-Teilnehmers

4.1.3 AlarmDiag

4.1.3.1 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
    
```

4.1.3.2 ST_PN_Diag

Die Datenstruktur **ST_PN_Diag** enthält eine Diagnose-Meldung einer Klemme, die über ein PN-Gerät und 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.

The screenshot shows the 'Diag History' window in the Beckhoff System Manager. It features a toolbar with 'Update History', 'Auto Update' (checkbox), 'Clear Diag History', and 'Export Diag History'. The main area contains a table of diagnostic events:

Type	Timestamp	Message	AddInfo	MessageID
Warning	20/12/2023 10:48:32 640 ms	ek9300.test: AR received diagnosis alarm (Alarmtype 0x0003, Slot 3, Subslot 2).	Yes	6
Warning	20/12/2023 10:48:32 636 ms	ek9300.test: AR received diagnosis alarm (Alarmtype 0x0003, Slot 3, Subslot 1).	Yes	5
Warning	20/12/2023 10:48:32 580 ms	ek9300.test: AR received diagnosis alarm (Alarmtype 0x0001, Slot 0, Subslot 1).	Yes	4
Info	20/12/2023 10:48:12 300 ms	ek9300.test: AR is established (got ApplReady).	No	3
Info	20/12/2023 10:48:10 001 ms	ek9300.test: Controller send PmEnd.	No	2
Info	20/12/2023 10:48:09 980 ms	ek9300.test: Controller start the parameterization.	No	1
Info	20/12/2023 10:48:09 969 ms	ek9300.test: Controller send ConnectReq to device.	No	0

Below the table, a detailed view of a selected alarm is shown:

```

Full alarm (0x0003)
The diagnosis alarm received from:
  API Number 0x00000000, Slot Number 0x0003, Subslot Number 0x0002
    
```

4.1.3.3 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

```

4.1.4 ST_PN_DeviceInfo

Die Datenstruktur **ST_PN_DeviceInfo** enthält Informationen über den PROFINET-Controller.

```

TYPE str_PN_DeviceInfo :
STRUCT
  nBoxAddr          : UINT;
  sBoxName          : STRING(240);
  sIP_Addr          : STRING(15);
  sSubNetMask       : STRING(15);
  sDefaultGateway   : STRING(15);
  PnIoState         : WORD;
  PnIoDiag          : WORD;
  nNrOfInputCR      : UINT;
  nNrOfOutputCR     : UINT;
  nCycleTime        : UDINT;
END_STRUCT
END_TYPE

```

Name	Beschreibung
nBoxAddr	BOX ID aus TwinCAT
sBoxName	PROFINET Box/Geräte Name
sIP_Addr	IP Adresse
sSubNetMask	Subnetzmaske
sDefaultGateway	Default Gateway
PnIoState	PROFINET Status
PnIoDiag	PROFINET Diagnose
nNrOfInputCR	Anzahl der input CR (Communication Relation)
nNrOfOutputCR	Anzahl der output CR (Communication Relation)
nCycleTime	PROFINET Zykluszeit in µs

4.2 Device

4.2.1 str_Diag_PN_Settings

Die Datenstruktur **str_Diag_PN_Settings** enthält IP-Informationen, die in PROFINET-Geräten fest hinterlegt sind.

```

TYPE str_Diag_PN_Settings :
STRUCT
  IP_Addr           : ARRAY [0..3] OF BYTE;
  Sub_Net_Mask     : ARRAY [0..3] OF BYTE;
  Default_Gateway  : ARRAY [0..3] OF BYTE;
  st_PN_Name       : STRING(255);
END_STRUCT
END_TYPE

```

4.2.2 str_IuM_Data

Die Datenstruktur **str_IuM_Data** enthält Informationen über den PROFINET-Teilnehmer.

```

TYPE str_IuM_Data :
STRUCT
  st_IM_TagFunction      : STRING(32);
  st_IM_TagLocation     : STRING(22);
  st_IM_Date            : STRING(16);
  st_IM_Description     : STRING(54);
  st_IM_Signature       : STRING(54);
END_STRUCT
END_TYPE

```

4.3 EL6631-0010

4.3.1 E_PN_ALARM_TYP

E_PN_ALARM_TYP

Der Aufzählungstyp **E_PN_ALARM_TYP** listet alle Alarmer 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

```

4.4 IM

4.4.1 RecStruct

Die Datenstruktur **RecStruct** bildet die PROFINET Record Daten, also die azyklischen Parameterdaten, in der PLC ab.

```

TYPE RecStruct :
STRUCT
  nRw      : WORD; (* 0 = read / 1 = write *)
  nNrOfAR  : WORD; (* Number of Application Relationship*)
  nAPI     : DWORD; (* Application Process Identifier *)
  nSlot    : WORD; (* Coupler = 0 / Terminal > 0*)
  nSubSlot : WORD; (* 1 *)
  nIndex   : WORD; (* Register e.g. 0xAFF0 = IM0 *)
  nHWnLength : WORD; (* Read = 0 *)
  nLenOfAlign : WORD; (* 0 *)
END_STRUCT
END_TYPE

```

4.5 Port Diagnose

4.5.1 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
```

4.5.2 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
```

5 Anhang

5.1 ADS Return Codes

Gruppierung der Fehlercodes:

Globale Fehlercodes: [0x0000 \[▶ 45\]](#)... (0x9811_0000 ...)

Router Fehlercodes: [0x0500 \[▶ 45\]](#)... (0x9811_0500 ...)

Allgemeine ADS Fehler: [0x0700 \[▶ 46\]](#)... (0x9811_0700 ...)

RTime Fehlercodes: [0x1000 \[▶ 48\]](#)... (0x9811_1000 ...)

Globale Fehlercodes

Hex	Dec	HRESULT	Name	Beschreibung
0x0	0	0x98110000	ERR_NOERROR	Kein Fehler.
0x1	1	0x98110001	ERR_INTERNAL	Interner Fehler.
0x2	2	0x98110002	ERR_NORTIME	Keine Echtzeit.
0x3	3	0x98110003	ERR_ALLOCLOCKEDMEM	Zuweisung gesperrt - Speicherfehler.
0x4	4	0x98110004	ERR_INSERTMAILBOX	Postfach voll – Es konnte die ADS Nachricht nicht versendet werden. Reduzieren der Anzahl der ADS Nachrichten pro Zyklus bringt Abhilfe.
0x5	5	0x98110005	ERR_WRONGRECEIVEHMSG	Falsches HMSG.
0x6	6	0x98110006	ERR_TARGETPORTNOTFOUND	Ziel-Port nicht gefunden – ADS Server ist nicht gestartet oder erreichbar.
0x7	7	0x98110007	ERR_TARGETMACHINENOTFOUND	Zielrechner nicht gefunden – AMS Route wurde nicht gefunden.
0x8	8	0x98110008	ERR_UNKNOWNCMDID	Unbekannte Befehl-ID.
0x9	9	0x98110009	ERR_BADTASKID	Ungültige Task-ID.
0xA	10	0x9811000A	ERR_NOIO	Kein IO.
0xB	11	0x9811000B	ERR_UNKNOWNAMSCMD	Unbekannter AMS-Befehl.
0xC	12	0x9811000C	ERR_WIN32ERROR	Win32 Fehler.
0xD	13	0x9811000D	ERR_PORTNOTCONNECTED	Port nicht verbunden.
0xE	14	0x9811000E	ERR_INVALIDAMSLLENGTH	Ungültige AMS-Länge.
0xF	15	0x9811000F	ERR_INVALIDAMSNETID	Ungültige AMS Net ID.
0x10	16	0x98110010	ERR_LOWINSTLEVEL	Installations-Level ist zu niedrig –TwinCAT 2 Lizenzfehler.
0x11	17	0x98110011	ERR_NODEBUGINTAVAILABLE	Kein Debugging verfügbar.
0x12	18	0x98110012	ERR_PORTDISABLED	Port deaktiviert – TwinCAT System Service nicht gestartet.
0x13	19	0x98110013	ERR_PORTALREADYCONNECTED	Port bereits verbunden.
0x14	20	0x98110014	ERR_AMSSYNC_W32ERROR	AMS Sync Win32 Fehler.
0x15	21	0x98110015	ERR_AMSSYNC_TIMEOUT	AMS Sync Timeout.
0x16	22	0x98110016	ERR_AMSSYNC_AMSERROR	AMS Sync Fehler.
0x17	23	0x98110017	ERR_AMSSYNC_NOINDEXINMAP	Keine Index-Map für AMS Sync vorhanden.
0x18	24	0x98110018	ERR_INVALIDAMSPORT	Ungültiger AMS-Port.
0x19	25	0x98110019	ERR_NOMEMORY	Kein Speicher.
0x1A	26	0x9811001A	ERR_TCPSEND	TCP Sendefehler.
0x1B	27	0x9811001B	ERR_HOSTUNREACHABLE	Host nicht erreichbar.
0x1C	28	0x9811001C	ERR_INVALIDAMSFRAGMENT	Ungültiges AMS Fragment.
0x1D	29	0x9811001D	ERR_TLSSSEND	TLS Sendefehler – Secure ADS Verbindung fehlgeschlagen.
0x1E	30	0x9811001E	ERR_ACCESSDENIED	Zugriff Verweigert – Secure ADS Zugriff verweigert.

Router Fehlercodes

Hex	Dec	HRESULT	Name	Beschreibung
0x500	1280	0x98110500	ROUTERERR_NOLOCKEDMEMORY	Lockierter Speicher kann nicht zugewiesen werden.
0x501	1281	0x98110501	ROUTERERR_RESIZEMEMORY	Die Größe des Routerspeichers konnte nicht geändert werden.
0x502	1282	0x98110502	ROUTERERR_MAILBOXFULL	Das Postfach hat die maximale Anzahl der möglichen Meldungen erreicht.
0x503	1283	0x98110503	ROUTERERR_DEBUGBOXFULL	Das Debug Postfach hat die maximale Anzahl der möglichen Meldungen erreicht.
0x504	1284	0x98110504	ROUTERERR_UNKNOWNPORTTYPE	Der Porttyp ist unbekannt.
0x505	1285	0x98110505	ROUTERERR_NOTINITIALIZED	Router ist nicht initialisiert.
0x506	1286	0x98110506	ROUTERERR_PORTALREADYINUSE	Die Portnummer ist bereits vergeben.
0x507	1287	0x98110507	ROUTERERR_NOTREGISTERED	Der Port ist nicht registriert.
0x508	1288	0x98110508	ROUTERERR_NOMOREQUEUES	Die maximale Portanzahl ist erreicht.
0x509	1289	0x98110509	ROUTERERR_INVALIDPORT	Der Port ist ungültig.
0x50A	1290	0x9811050A	ROUTERERR_NOTACTIVATED	Der Router ist nicht aktiv.
0x50B	1291	0x9811050B	ROUTERERR_FRAGMENTBOXFULL	Das Postfach hat die maximale Anzahl für fragmentierte Nachrichten erreicht.
0x50C	1292	0x9811050C	ROUTERERR_FRAGMENTTIMEOUT	Fragment Timeout aufgetreten.
0x50D	1293	0x9811050D	ROUTERERR_TOBEREMOVED	Port wird entfernt.

Allgemeine ADS Fehlercodes

Hex	Dec	HRESULT	Name	Beschreibung
0x700	1792	0x98110700	ADSERR_DEVICE_ERROR	Allgemeiner Gerätefehler.
0x701	1793	0x98110701	ADSERR_DEVICE_SRVNOTSUPP	Service wird vom Server nicht unterstützt.
0x702	1794	0x98110702	ADSERR_DEVICE_INVALIDGRP	Ungültige Index-Gruppe.
0x703	1795	0x98110703	ADSERR_DEVICE_INVALIDOFFSET	Ungültiger Index-Offset.
0x704	1796	0x98110704	ADSERR_DEVICE_INVALIDACCESS	Lesen oder Schreiben nicht gestattet.
0x705	1797	0x98110705	ADSERR_DEVICE_INVALIDSIZE	Parametergröße nicht korrekt.
0x706	1798	0x98110706	ADSERR_DEVICE_INVALIDDATA	Ungültige Daten-Werte.
0x707	1799	0x98110707	ADSERR_DEVICE_NOTREADY	Gerät nicht betriebsbereit.
0x708	1800	0x98110708	ADSERR_DEVICE_BUSY	Gerät beschäftigt.
0x709	1801	0x98110709	ADSERR_DEVICE_INVALIDCONTEXT	Ungültiger Kontext vom Betriebssystem - Kann durch Verwendung von ADS Bausteinen in unterschiedlichen Tasks auftreten. Abhilfe kann die Multitasking-Synchronisation in der SPS geben.
0x70A	1802	0x9811070A	ADSERR_DEVICE_NOMEMORY	Nicht genügend Speicher.
0x70B	1803	0x9811070B	ADSERR_DEVICE_INVALIDPARM	Ungültige Parameter-Werte.
0x70C	1804	0x9811070C	ADSERR_DEVICE_NOTFOUND	Nicht gefunden (Dateien,...).
0x70D	1805	0x9811070D	ADSERR_DEVICE_SYNTAX	Syntax-Fehler in Datei oder Befehl.
0x70E	1806	0x9811070E	ADSERR_DEVICE_INCOMPATIBLE	Objekte stimmen nicht überein.
0x70F	1807	0x9811070F	ADSERR_DEVICE_EXISTS	Objekt ist bereits vorhanden.
0x710	1808	0x98110710	ADSERR_DEVICE_SYMBOLNOTFOUND	Symbol nicht gefunden.
0x711	1809	0x98110711	ADSERR_DEVICE_SYMBOLVERSIONINVALID	Symbol-Version ungültig – Kann durch einen Online-Change auftreten. Erzeuge einen neuen Handle.
0x712	1810	0x98110712	ADSERR_DEVICE_INVALIDSTATE	Gerät (Server) ist im ungültigen Zustand.
0x713	1811	0x98110713	ADSERR_DEVICE_TRANSMODENOTSUPP	AdsTransMode nicht unterstützt.
0x714	1812	0x98110714	ADSERR_DEVICE_NOTIFYHANDINVALID	Notification Handle ist ungültig.
0x715	1813	0x98110715	ADSERR_DEVICE_CLIENTUNKNOWN	Notification-Client nicht registriert.
0x716	1814	0x98110716	ADSERR_DEVICE_NOMOREHDL	Keine weiteren Handles verfügbar.
0x717	1815	0x98110717	ADSERR_DEVICE_INVALIDWATCHSIZE	Größe der Notification zu groß.
0x718	1816	0x98110718	ADSERR_DEVICE_NOTINIT	Gerät nicht initialisiert.
0x719	1817	0x98110719	ADSERR_DEVICE_TIMEOUT	Gerät hat einen Timeout.
0x71A	1818	0x9811071A	ADSERR_DEVICE_NOINTERFACE	Interface Abfrage fehlgeschlagen.
0x71B	1819	0x9811071B	ADSERR_DEVICE_INVALIDINTERFACE	Falsches Interface angefordert.
0x71C	1820	0x9811071C	ADSERR_DEVICE_INVALIDCLSID	Class-ID ist ungültig.
0x71D	1821	0x9811071D	ADSERR_DEVICE_INVALIDOBJID	Object-ID ist ungültig.
0x71E	1822	0x9811071E	ADSERR_DEVICE_PENDING	Anforderung steht aus.
0x71F	1823	0x9811071F	ADSERR_DEVICE_ABORTED	Anforderung wird abgebrochen.
0x720	1824	0x98110720	ADSERR_DEVICE_WARNING	Signal-Warnung.
0x721	1825	0x98110721	ADSERR_DEVICE_INVALIDARRAYIDX	Ungültiger Array-Index.
0x722	1826	0x98110722	ADSERR_DEVICE_SYMBOLNOTACTIVE	Symbol nicht aktiv.
0x723	1827	0x98110723	ADSERR_DEVICE_ACCESSDENIED	Zugriff verweigert.
0x724	1828	0x98110724	ADSERR_DEVICE_LICENSENOTFOUND	Fehlende Lizenz.
0x725	1829	0x98110725	ADSERR_DEVICE_LICENSEEXPIRED	Lizenz abgelaufen.
0x726	1830	0x98110726	ADSERR_DEVICE_LICENSEEXCEEDED	Lizenz überschritten.
0x727	1831	0x98110727	ADSERR_DEVICE_LICENSEINVALID	Lizenz ungültig.
0x728	1832	0x98110728	ADSERR_DEVICE_LICENSESYSTEMID	Lizenzproblem: System-ID ist ungültig.
0x729	1833	0x98110729	ADSERR_DEVICE_LICENSENOTIMELIMIT	Lizenz nicht zeitlich begrenzt.
0x72A	1834	0x9811072A	ADSERR_DEVICE_LICENSEFUTUREISSUE	Lizenzproblem: Zeitpunkt in der Zukunft.
0x72B	1835	0x9811072B	ADSERR_DEVICE_LICENSETIMETOLONG	Lizenz-Zeitraum zu lang.
0x72C	1836	0x9811072C	ADSERR_DEVICE_EXCEPTION	Exception beim Systemstart.
0x72D	1837	0x9811072D	ADSERR_DEVICE_LICENSEDUPLICATED	Lizenz-Datei zweimal gelesen.
0x72E	1838	0x9811072E	ADSERR_DEVICE_SIGNATUREINVALID	Ungültige Signatur.
0x72F	1839	0x9811072F	ADSERR_DEVICE_CERTIFICATEINVALID	Zertifikat ungültig.
0x730	1840	0x98110730	ADSERR_DEVICE_LICENSEOEMNOTFOUND	Public Key vom OEM nicht bekannt.
0x731	1841	0x98110731	ADSERR_DEVICE_LICENSERESTRICTED	Lizenz nicht gültig für diese System.ID.
0x732	1842	0x98110732	ADSERR_DEVICE_LICENSEDEMODENIED	Demo-Lizenz untersagt.
0x733	1843	0x98110733	ADSERR_DEVICE_INVALIDFNCID	Funktions-ID ungültig.
0x734	1844	0x98110734	ADSERR_DEVICE_OUTOFRANGE	Außerhalb des gültigen Bereiches.
0x735	1845	0x98110735	ADSERR_DEVICE_INVALIDALIGNMENT	Ungültiges Alignment.

Hex	Dec	HRESULT	Name	Beschreibung
0x736	1846	0x98110736	ADSERR_DEVICE_LICENSEPLATFORM	Ungültiger Plattform Level.
0x737	1847	0x98110737	ADSERR_DEVICE_FORWARD_PL	Kontext – Weiterleitung zum Passiv-Level.
0x738	1848	0x98110738	ADSERR_DEVICE_FORWARD_DL	Kontext – Weiterleitung zum Dispatch-Level.
0x739	1849	0x98110739	ADSERR_DEVICE_FORWARD_RT	Kontext – Weiterleitung zur Echtzeit.
0x740	1856	0x98110740	ADSERR_CLIENT_ERROR	Clientfehler.
0x741	1857	0x98110741	ADSERR_CLIENT_INVALIDPARM	Dienst enthält einen ungültigen Parameter.
0x742	1858	0x98110742	ADSERR_CLIENT_LISTEMPTY	Polling-Liste ist leer.
0x743	1859	0x98110743	ADSERR_CLIENT_VARUSED	Var-Verbindung bereits im Einsatz.
0x744	1860	0x98110744	ADSERR_CLIENT_DUPLINVOKEID	Die aufgerufene ID ist bereits in Benutzung.
0x745	1861	0x98110745	ADSERR_CLIENT_SYNC TIMEOUT	Timeout ist aufgetreten – Die Gegenstelle antwortet nicht im vorgegebenen ADS Timeout. Die Routeneinstellung der Gegenstelle kann falsch konfiguriert sein.
0x746	1862	0x98110746	ADSERR_CLIENT_W32ERROR	Fehler im Win32 Subsystem.
0x747	1863	0x98110747	ADSERR_CLIENT_TIMEOUTINVALID	Ungültiger Client Timeout-Wert.
0x748	1864	0x98110748	ADSERR_CLIENT_PORTNOTOPEN	Port nicht geöffnet.
0x749	1865	0x98110749	ADSERR_CLIENT_NOAMSADDR	Keine AMS Adresse.
0x750	1872	0x98110750	ADSERR_CLIENT_SYNCINTERNAL	Interner Fehler in Ads-Sync.
0x751	1873	0x98110751	ADSERR_CLIENT_ADDHASH	Überlauf der Hash-Tabelle.
0x752	1874	0x98110752	ADSERR_CLIENT_REMOVEHASH	Schlüssel in der Tabelle nicht gefunden.
0x753	1875	0x98110753	ADSERR_CLIENT_NOMORESVM	Keine Symbole im Cache.
0x754	1876	0x98110754	ADSERR_CLIENT_SYNCRESINVALID	Ungültige Antwort erhalten.
0x755	1877	0x98110755	ADSERR_CLIENT_SYNCPORTLOCKED	Sync Port ist verriegelt.
0x756	1878	0x98110756	ADSERR_CLIENT_REQUESTCANCELLED	Die Anfrage wurde abgebrochen.

RTime Fehlercodes

Hex	Dec	HRESULT	Name	Beschreibung
0x1000	4096	0x98111000	RTERR_INTERNAL	Interner Fehler im Echtzeit-System.
0x1001	4097	0x98111001	RTERR_BADTIMERPERIODS	Timer-Wert nicht gültig.
0x1002	4098	0x98111002	RTERR_INVALIDTASKPTR	Task-Pointer hat den ungültigen Wert 0 (null).
0x1003	4099	0x98111003	RTERR_INVALIDSTACKPTR	Stack-Pointer hat den ungültigen Wert 0 (null).
0x1004	4100	0x98111004	RTERR_PrioEXISTS	Die Request Task Priority ist bereits vergeben.
0x1005	4101	0x98111005	RTERR_NOMORETCB	Kein freier TCB (Task Control Block) verfügbar. Maximale Anzahl von TCBs beträgt 64.
0x1006	4102	0x98111006	RTERR_NOMORESEMAS	Keine freien Semaphoren zur Verfügung. Maximale Anzahl der Semaphoren beträgt 64.
0x1007	4103	0x98111007	RTERR_NOMOREQUEUES	Kein freier Platz in der Warteschlange zur Verfügung. Maximale Anzahl der Plätze in der Warteschlange beträgt 64.
0x100D	4109	0x9811100D	RTERR_EXTIRQALREADYDEF	Ein externer Synchronisations-Interrupt wird bereits angewandt.
0x100E	4110	0x9811100E	RTERR_EXTIRQNOTDEF	Kein externer Sync-Interrupt angewandt.
0x100F	4111	0x9811100F	RTERR_EXTIRQINSTALLFAILED	Anwendung des externen Synchronisierungs-Interrupts ist fehlgeschlagen.
0x1010	4112	0x98111010	RTERR_IRQLNOTLESSOREQUAL	Aufruf einer Service-Funktion im falschen Kontext
0x1017	4119	0x98111017	RTERR_VMXNOTSUPPORTED	Intel VT-x Erweiterung wird nicht unterstützt.
0x1018	4120	0x98111018	RTERR_VMXDISABLED	Intel VT-x Erweiterung ist nicht aktiviert im BIOS.
0x1019	4121	0x98111019	RTERR_VMXCONTROLSMISSING	Fehlende Funktion in Intel VT-x Erweiterung.
0x101A	4122	0x9811101A	RTERR_VMXENABLEFAILS	Aktivieren von Intel VT-x schlägt fehl.

Spezifische positive HRESULT Return Codes:

HRESULT	Name	Beschreibung
0x0000_0000	S_OK	Kein Fehler.
0x0000_0001	S_FALSE	Kein Fehler. Bsp.: erfolgreiche Abarbeitung, bei der jedoch ein negatives oder unvollständiges Ergebnis erzielt wurde.
0x0000_0203	S_PENDING	Kein Fehler. Bsp.: erfolgreiche Abarbeitung, bei der jedoch noch kein Ergebnis vorliegt.
0x0000_0256	S_WATCHDOG_TIMEOUT	Kein Fehler. Bsp.: erfolgreiche Abarbeitung, bei der jedoch eine Zeitüberschreitung eintrat.

TCP Winsock-Fehlercodes

Hex	Dec	Name	Beschreibung
0x274C	10060	WSAETIMEDOUT	Verbindungs Timeout aufgetreten - Fehler beim Herstellen der Verbindung, da die Gegenstelle nach einer bestimmten Zeitspanne nicht ordnungsgemäß reagiert hat, oder die hergestellte Verbindung konnte nicht aufrecht erhalten werden, da der verbundene Host nicht reagiert hat.
0x274D	10061	WSAECONNREFUSED	Verbindung abgelehnt - Es konnte keine Verbindung hergestellt werden, da der Zielcomputer dies explizit abgelehnt hat. Dieser Fehler resultiert normalerweise aus dem Versuch, eine Verbindung mit einem Dienst herzustellen, der auf dem fremden Host inaktiv ist—das heißt, einem Dienst, für den keine Serveranwendung ausgeführt wird.
0x2751	10065	WSAEHOSTUNREACH	Keine Route zum Host - Ein Socketvorgang bezog sich auf einen nicht verfügbaren Host.
Weitere Winsock-Fehlercodes: Win32-Fehlercodes			

5.2 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

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

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

Beckhoff Unternehmenszentrale

Beckhoff Automation GmbH & Co. KG

Hülshorstweg 20
33415 Verl
Deutschland

Telefon: +49 5246 963-0
E-Mail: info@beckhoff.com
Internet: www.beckhoff.com

Mehr Informationen:
www.beckhoff.com/te1000

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

