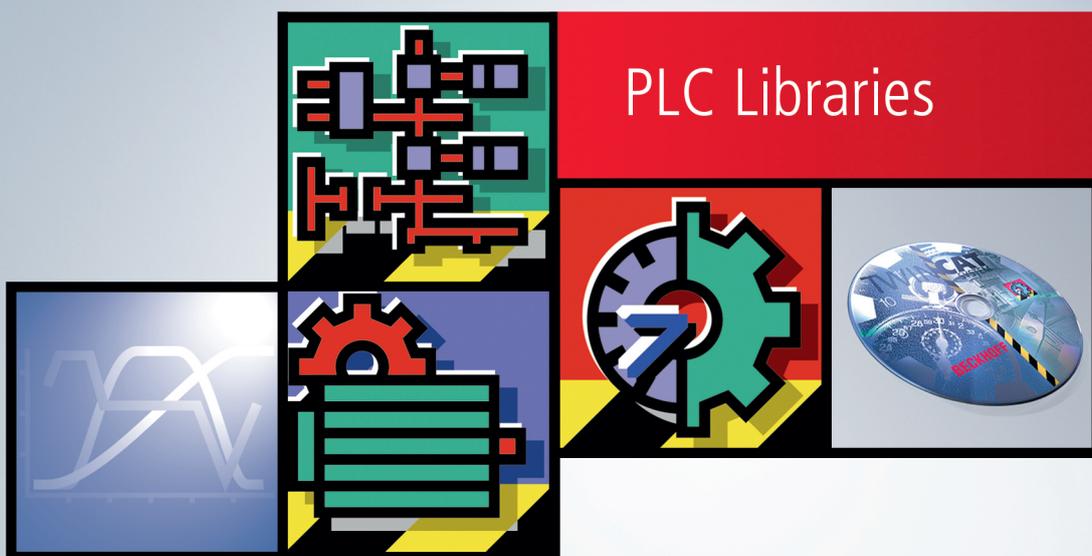


Handbuch | DE

TX1200

TwinCAT 2 | PLC-Bibliothek: TcDataExchange



Inhaltsverzeichnis

1	Vorwort	5
1.1	Hinweise zur Dokumentation	5
1.2	Sicherheitshinweise	6
1.3	Hinweise zur Informationssicherheit	7
2	Einleitung	8
3	Programmierung	9
3.1	Schreib-/Lesebausteine	9
3.1.1	FB_ReadAdsSymByName.....	9
3.1.2	FB_WriteAdsSymByName	10
3.1.3	FB_WriteBoolOnDelta	11
3.1.4	FB_WriteByteOnDelta	13
3.1.5	FB_WriteWordOnDelta	14
3.1.6	FB_WriteDWordOnDelta.....	15
3.1.7	FB_WriteRealOnDelta.....	16
3.1.8	FB_WriteLRealOnDelta.....	18
3.2	Überwachungsbausteine.....	19
3.2.1	FB_CheckWatchdog	19
3.2.2	FB_WriteWatchdog	20
3.3	Datentypen.....	21
3.3.1	E_AdsComMode	21
4	Anhang	22
4.1	Support und Service.....	22

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, für jede Installation und Inbetriebnahme die zu dem betreffenden Zeitpunkt veröffentlichte 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 weiter entwickelt.

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.

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

1.2 Sicherheitshinweise

Sicherheitsbestimmungen

Beachten Sie die folgenden Sicherheitshinweise und Erklärungen!
Produktspezifische Sicherheitshinweise finden Sie auf den folgenden Seiten oder in den Bereichen Montage, Verdrahtung, Inbetriebnahme usw.

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.

Erklärung der Symbole

In der vorliegenden Dokumentation werden die folgenden Symbole mit einem nebenstehenden Sicherheitshinweis oder Hinweistext verwendet. Die Sicherheitshinweise sind aufmerksam zu lesen und unbedingt zu befolgen!

GEFAHR

Akute Verletzungsgefahr!

Wenn der Sicherheitshinweis neben diesem Symbol nicht beachtet wird, besteht unmittelbare Gefahr für Leben und Gesundheit von Personen!

WARNUNG

Verletzungsgefahr!

Wenn der Sicherheitshinweis neben diesem Symbol nicht beachtet wird, besteht Gefahr für Leben und Gesundheit von Personen!

VORSICHT

Schädigung von Personen!

Wenn der Sicherheitshinweis neben diesem Symbol nicht beachtet wird, können Personen geschädigt werden!

HINWEIS

Schädigung von Umwelt oder Geräten

Wenn der Hinweis neben diesem Symbol nicht beachtet wird, können Umwelt oder Geräte geschädigt werden.



Tipps oder Fingerzeige

Dieses Symbol kennzeichnet Informationen, die zum besseren Verständnis beitragen.

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 Einleitung

Die vorliegenden Funktionsbausteine vereinfachen den Datenaustausch zwischen TwinCAT SPS-Laufzeitsystemen und/oder anderen ADS-Geräten (TwinCAT NC, Busklemmen Controller, ...).

Die Funktionsbausteine *FB_WriteXXXOnDelta()* führen einen Schreibvorgang aus, wenn das Eingangssignal einen bestimmten Grenzwert über-/ bzw. unterschritten hat. Wie oft das Eingangssignal überprüft werden soll, kann eingestellt werden. Durch das ereignisgesteuerte Schreiben der Daten wird die Belastung der Feldbusses minimiert. Tritt bei der Übertragung ein Fehler auf, so wird der Vorgang solange wiederholt, bis die Verbindung wieder vorhanden ist. Als Quell- und Zielvariablen sind alle, in der TwinCAT PLC unterstützten Datentypen zugelassen. Symbolnamen werden ebenfalls unterstützt.

Die Funktionsbausteine *FB_XXXAdsSymByName()* führen einen Schreib- bzw. Lesevorgang über den Namen der SPS-Variablen aus.

Zur Überwachung einzelner Kommunikationspartner stehen Watchdog-Bausteine zur Verfügung. Der zu überwachende Teilnehmer versendet zyklisch einen inkrementierenden Zähler. Bei dem Empfänger wird überprüft, ob sich der Zählerstand innerhalb einer bestimmten Zeit ändert.

3 Programmierung

3.1 Schreib-/Lesebausteine

Name	Beschreibung
FB_ReadAdsSymByName [▶ 9]	Liest eine Variable eines beliebigen Datentyps per Variablenname
FB_WriteAdsSymByName [▶ 10]	Schreibt eine Variable eines beliebigen Datentyps per Variablenname
FB_WriteBoolOnDelta [▶ 11]	Schreibt ereignisgesteuert eine Variable vom Typ BOOLEAN
FB_WriteByteOnDelta [▶ 13]	Schreibt ereignisgesteuert eine Variable vom Typ BYTE
FB_WriteWordOnDelta [▶ 14]	Schreibt ereignisgesteuert eine Variable vom Typ WORD
FB_WriteDWordOnDelta [▶ 15]	Schreibt ereignisgesteuert eine Variable vom Typ DWORD
FB_WriteRealOnDelta [▶ 16]	Schreibt ereignisgesteuert eine Variable vom Typ REAL
FB_WriteLRealOnDelta [▶ 18]	Schreibt ereignisgesteuert eine Variable vom Typ LREAL

3.1.1 FB_ReadAdsSymByName



Der Baustein ermöglicht das Lesen eines beliebigen Wertes von einer anderen Steuerung, mit Hilfe des Symbolnamens.

Bei einer positiven Flanke am Eingang *bRead* liest der Baustein den Wert der Variablen *sVarName* aus von dem ausgewählten ADS-Gerätes (z.B. SPS). Das ADS-Gerät wird durch die AMS-NetId (*sNetId*) und der AMS-Portnummer (*nPort*) angegeben. Der Wert wird in die Variable geschrieben, auf der *nDestAddr* zeigt.

Mit Hilfe des Eingangs *eComMode* kann die interne Arbeitsweise des Bausteins verändert werden:

eComMode := eAdsComModeSecureCom: Nach jedem Lesevorgang wird der Handle der SPS-Variablen wieder freigegeben. Dieser Modus sollte verwendet werden, wenn sehr langsam Werte ausgetauscht werden.

eComMode := eAdsComModeFastCom: Solange sich die Eingänge *sVarName*, *sNetID* und *nPort* nicht ändern, wird der Handle der SPS-Variablen nicht nach jedem Lesevorgang freigegeben. Dieser Modus sollte verwendet werden, wenn sehr häufig Werte ausgetauscht werden.

VAR_INPUT

```

bRead      : BOOL;
sNetId     : T_AmsNetId;
nPort      : T_AmsPort := AMSPORT_R0_PLC_RTSL1;
sVarName   : STRING;
nDestAddr  : DWORD;
nLen       : UDINT;
tTimeout   : TIME := DEFAULT_ADS_TIMEOUT;
eComMode   : E_AdsComMode := eAdsComModeSecureCom;
    
```

bRead: Baustein liest den Inhalt der Variablen <i>sVarName</i> des ausgewählten ADS-Gerätes und schreibt diese in die Variable auf die der Zeiger <i>nDestAddr</i> zeigt.	sNetId: AMS-NetId des ADS-Gerätes, von dem der Wert gelesen werden soll.	nPort: AMS-Portnummer des ADS-Gerätes, von dem der Wert gelesen werden soll.	sVarName: Symbolname der zu lesenden Variablen auf dem ausgewählten ADS-Gerät.	nDestAddr: Adresse der Variablen, in der der gelesene Wert geschrieben wird.	nLen: Länge der zu lesenden Variable in Byte.	tTimeout: Zeit bis zum Abbruch der Abarbeitung.	eComMode : Enum, mit dem bestimmt wird ob der Handle der SPS-Variablen nach jedem Lesen wieder freigegeben wird.
---	--	--	--	--	---	---	--

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Übertragung ist aktiv.	bError: Bei der Übertragung ist ein Fehler aufgetreten.	nErrorId: ADS-Fehlernummer, falls ein Fehler aufgetreten ist.
--------------------------------------	--	--

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2247	PC/CX	TcDataExchange-Bibliothek ab V1.1.0

Sehen Sie dazu auch

📖 E_AdsComMode [▶ 21]

3.1.2 FB_WriteAdsSymByName



Schreiben eines beliebigen Wertes auf eine andere Steuerung mit Hilfe des Symbolnamens.

Bei einer positiven Flanke am Eingang *bWrite* schreibt der Baustein den Wert auf den der Zeiger *nSrcAddr* zeigt, in die Variable *sVarName* des ausgewählten ADS-Gerätes (z.B. SPS). Das ADS-Gerät wird durch die AMS-NetId (*sNetId*) und der AMS-Portnummer (*nPort*) angegeben.

Mit Hilfe des Eingangs *eComMode* kann die interne Arbeitsweise des Bausteins verändert werden:

eComMode := eAdsComModeSecureCom: Nach jedem Schreibvorgang wird der Handle der SPS-Variablen wieder freigegeben. Dieser Modus sollte verwendet werden, wenn sehr langsam Werte ausgetauscht werden.

eComMode := eAdsComModeFastCom: Solange sich die Eingänge *sVarName*, *sNetID* und *nPort* nicht ändern, wird der Handle der SPS-Variablen nicht nach jedem Schreibvorgang freigegeben. Dieser Modus sollte verwendet werden, wenn sehr häufig Werte ausgetauscht werden.

VAR_INPUT

```
bWrite      : BOOL;
sNetId      : T_AmsNetId;
nPort       : T_AmsPort := AMSPORT_R0_PLC_RTSL1;
sVarName    : STRING;
nSrcAddr    : DWORD;
nLen        : UDINT;
tTimeout    : TIME := DEFAULT_ADS_TIMEOUT;
eComMode    : E_AdsComMode := eAdsComModeSecureCom;
```

bWrite: Baustein schreibt den Inhalt der Variablen, auf den der Zeiger <i>nSrcAddr</i> zeigt, in die Variable <i>sVarName</i> des ausgewählten ADS-Gerätes.	sNetId: AMS-NetId des ADS-Gerätes, zu dem der Wert übertragen werden soll.	nPort: AMS-Portnummer des ADS-Gerätes, zu dem der Wert übertragen werden soll.	sVarName: Symbolname der zu beschreibenden Variablen auf dem ausgewählten ADS-Gerät.	nSrcAddr: Adresse der Variablen, in der der zu schreibende Wert steht.	nLen: Länge der zu schreibenden Variablen in Byte.	tTimeout: Zeit bis zum Abbruch der Abarbeitung.	eComMode : Enum, mit dem bestimmt wird ob der Handle der SPS-Variablen nach jedem Schreiben wieder freigegeben wird.
---	--	--	--	--	--	---	--

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
nErrorId   : UDINT;
```

bBusy: Übertragung ist aktiv.	bError: Bei der Übertragung ist ein Fehler aufgetreten.	nErrorId: ADS-Fehlernummer, falls ein Fehler aufgetreten ist.
--------------------------------------	--	--

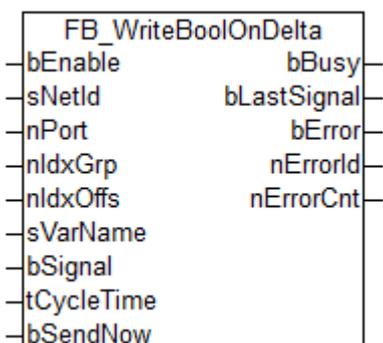
Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64 ab Build 2247	PC/CX	TcDataExchange-Bibliothek ab V1.1.0

Sehen Sie dazu auch

 E_AdsComMode [▶ 21]

3.1.3 FB_WriteBoolOnDelta



Ereignisgesteuertes Schreiben einer Variablen vom Typ BOOLEAN.

Der Funktionsbaustein *FB_WriteBoolOnDelta()* überprüft zyklisch, ob sich der Wert am Eingang *bSignal* geändert hat. Die Zykluszeit, für die Überprüfung wird durch den Parameter *tCycleTime* bestimmt. Wird für *tCycleTime* 0 sec angegeben, so wird in jedem SPS-Zyklus das Eingangssignal überprüft. Wurde eine Änderung erkannt, so wird der Wert des Signals an das angegebende ADS-Gerät verschickt. Adressiert wird der Empfänger durch die AMS-NetId und durch die Portnummer (siehe auch Identifikation ADS-Gerät). Die Position innerhalb des Empfängers wird durch die Index-Group/Index-Offset oder dem Symbolnamen angegeben. Üblicherweise ist dieses das Eingangsabbild oder der Merkerbereich.

Wird der Eingang *bEnable* auf FALSE gesetzt, so erfolgt keine weitere Übertragung des Signals.

VAR_INPUT

```
bEnable      : BOOL := FALSE;
sNetId       : T_AmsNetId;
nPort        : T_AmsPort := AMSPORT_R0_PLC_RTSl;
nIdxGrp      : UDINT;
nIdxOffs     : UDINT;
sVarName     : STRING;
bSignal      : BOOL;
tCycleTime   : TIME := t#0s;
bSendNow     : BOOL;
```

bEnable: Baustein Freischalten.	sNetId: AMS-NetId des ADS-Gerätes, nach dem der Wert übertragen werden soll.	nPort: AMS-Portnummer des ADS-Gerätes, nach dem der Wert übertragen werden soll.	nIdxGrp: Index-Group innerhalb des ADS-Gerätes, in der der Wert übertragen werden soll.	nIdxOffs: Index-Offset innerhalb des ADS-Gerätes, in der der Wert übertragen werden soll.	sVarName: Symbolname innerhalb des ADS-Gerätes, in der der Wert übertragen werden soll.	bSignal: Variable, dessen Wert übertragen werden soll.	tCycleTime: Zykluszeit, in der überprüft wird, ob sich das Eingangssignal geändert hat.	bSendNow: Durch eine positive Flanke wird der Wert unmittelbar übertragen.
---	--	--	---	---	---	--	---	--

VAR_OUTPUT

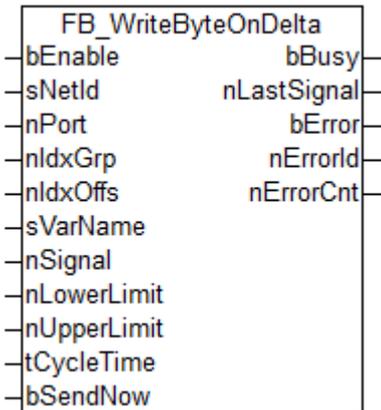
```
bBusy        : BOOL;
bLastSignal  : BOOL;
bError       : BOOL;
nErrorId     : UDINT;
nErrorCnt    : UDINT;
```

bBusy: Übertragung ist aktiv.	bLastSignal: zuletzt übertragener Wert.	bError: Bei der Übertragung ist ein Fehler aufgetreten.	nErrorId: Fehlernummer, falls ein Fehler aufgetreten ist.	nErrorCnt: Anzahl der fehlerhaften Übertragungsversuche.
--------------------------------------	--	--	--	---

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64	PC/CX, BX oder BC	TcDataExchange-Bibliothek ab V1.0.0

3.1.4 FB_WriteByteOnDelta



Ereignisgesteuertes Schreiben einer Variablen vom Typ BYTE.

Der Funktionsbaustein *FB_WriteByteOnDelta()* überprüft zyklisch, ob sich der Wert am Eingang *nSignal* geändert hat. Die Zykluszeit, für die Überprüfung wird durch den Parameter *tCycleTime* bestimmt. Wird für *tCycleTime* 0 sec angegeben, so wird in jedem SPS-Zyklus das Eingangssignal überprüft. Wird bei dem Vergleich festgestellt, dass der aktuelle Wert um *nUpperLimit* größer ist oder um den Wert *nLowerLimit* kleiner ist, so wird der Wert des Signals an das angegebene ADS-Gerät verschickt. Adressiert wird der Empfänger durch die AMS-NetId und durch die Portnummer (siehe auch Identifikation ADS-Gerät). Die Position innerhalb des Empfängers wird durch die Index-Group/Index-Offset oder dem Symbolnamen angegeben. Üblicherweise ist dieses das Eingangsabbild oder der Merkerbereich.

Wird der Eingang *bEnable* auf FALSE gesetzt, so erfolgt keine weitere Übertragung des Signals.

VAR_INPUT

```

bEnable      : BOOL := FALSE;
sNetId       : T_AmsNetId;
nPort        : T_AmsPort := AMSPORT_R0_PLC_RTS1;
nIdxGrp      : UDINT;
nIdxOffs     : UDINT;
sVarName     : STRING;
nSignal      : BYTE;
nLowerLimit  : BYTE;
nUpperLimit  : BYTE;
tCycleTime   : TIME := t#0s;
bSendNow     : BOOL;
  
```

bEnable	sNetId:	nPort:	nIdxGrp	nIdxOffs	sVarName:	nSignal:	nLower Limit:	nUpper Limit:	tCycleTime:	bSendNow:
: Baustein Freischalten.	AMS-NetId des ADS-Gerätes, nach dem der Wert übertragen werden soll.	AMS-Portnummer des ADS-Gerätes, nach dem der Wert übertragen werden soll.	: Index-Group innerhalb b des ADS-Gerätes, in der der Wert übertragen werden soll.	: Index-Offset innerhalb b des ADS-Gerätes, in der der Wert übertragen werden soll.	Symbolname innerhalb b des ADS-Gerätes, in der der Wert übertragen werden soll.	Variable, dessen Wert übertragen werden soll.	unterer Grenzwert.	oberer Grenzwert.	Zykluszeit, in der überprüft wird, ob das Eingangssignal die Grenzwerte überschritten hat.	Durch eine positive Flanke wird der Wert unmittelbar übertragen.

VAR OUTPUT

```

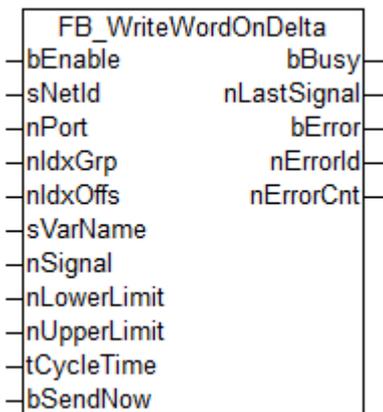
bBusy        : BOOL;
nLastSignal  : BYTE;
bError       : BOOL;
nErrorId     : UDINT;
nErrorCnt    : UDINT;
  
```

bBusy: Übertragung ist aktiv.	nLastSignal: zuletzt übertragender Wert.	bError: Bei der Übertragung ist ein Fehler aufgetreten.	nErrorId: Fehlernummer, falls ein Fehler aufgetreten ist.	nErrorCnt: Anzahl der fehlerhaften Übertragungsversuche.
--------------------------------------	---	--	--	---

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64	PC/CX, BX oder BC	TcDataExchange-Bibliothek ab V1.0.0

3.1.5 FB_WriteWordOnDelta



Ereignisgesteuertes Schreiben einer Variablen vom Typ WORD.

Der Funktionsbaustein *FB_WriteWordOnDelta()* überprüft zyklisch, ob sich der Wert am Eingang *nSignal* geändert hat. Die Zykluszeit, für die Überprüfung wird durch den Parameter *tCycleTime* bestimmt. Wird für *tCycleTime* 0 sec angegeben, so wird in jedem SPS-Zyklus das Eingangssignal überprüft. Wird bei dem Vergleich festgestellt, das der aktuelle Wert um *nUpperLimit* größer ist oder um den Wert *nLowerLimit* kleiner ist, so wird der Wert des Signal an das angegebende ADS-Gerät verschickt. Adressiert wird der Empfänger durch die AMS-NetId und durch die Portnummer (siehe auch Identifikation ADS-Gerät). Die Position innerhalb des Empfängers wird durch die Index-Group/Index-Offset oder dem Symbolnamen angegeben. Üblicherweise ist dieses das Eingangsabbild oder der Merkerbereich.

Wird der Eingang *bEnable* auf FALSE gesetzt, so erfolgt keine weitere Übertragung des Signal.

VAR INPUT

```

bEnable      : BOOL := FALSE;
sNetId       : T_AmsNetId;
nPort        : T_AmsPort := AMSPORT_R0_PLC_RTSL1;
nIdxGrp      : UDINT;
nIdxOffs     : UDINT;
sVarName     : STRING;
nSignal      : WORD;
nLowerLimit  : WORD;
nUpperLimit  : WORD;
tCycleTime   : TIME := t#0s;
bSendNow     : BOOL;
    
```

bEnable	sNetId:	nPort:	nIdxGrp	nIdxOffs	sVarName:	nSignal:	nLowerLimit:	nUpperLimit:	tCycleTime:	bSendNow:
: Baustein Freischalten.	AMS-NetId des ADS-Gerätes, nach dem der	AMS-Portnummer des ADS-Gerätes, nach dem der	: Index-Group innerhalb des ADS-Gerätes, in der	: Index-Offset innerhalb des ADS-Gerätes, in der	Symbolname innerhalb des ADS-Gerätes,	Variable, dessen Wert übertragen werden soll.	unterer Grenzwert.	oberer Grenzwert.	Zykluszeit, in der überprüft wird, ob das Eingang	Durch eine positive Flanke wird der Wert

	Wert übertragen werden soll.	Wert übertragen werden soll.	der Wert übertragen werden soll.	der Wert übertragen werden soll.	in der Wert übertragen werden soll.				ssignal die Grenzwe rte überschr itten hat.	unmittelb ar übertragen.
--	---------------------------------------	---------------------------------------	---	---	---	--	--	--	--	--------------------------------

VAR_OUTPUT

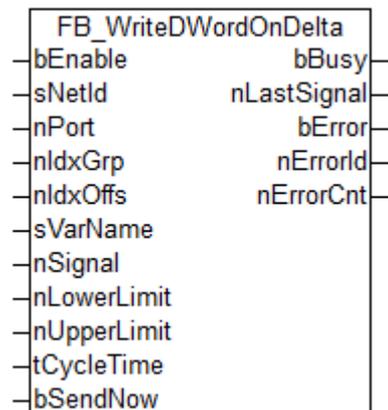
```
bBusy      : BOOL;
nLastSignal : WORD;
bError     : BOOL;
nErrorId   : UDINT;
nErrorCnt  : UDINT;
```

bBusy: Übertragung ist aktiv.	nLastSignal: zuletzt übertragender Wert.	bError: Bei der Übertragung ist ein Fehler aufgetreten.	nErrorId: Fehlernummer, falls ein Fehler aufgetreten ist.	nErrorCnt: Anzahl der fehlerhaften Übertragungsversuche.
--------------------------------------	---	--	--	---

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64	PC/CX, BX oder BC	TcDataExchange-Bibliothek ab V1.0.0

3.1.6 FB_WriteDWordOnDelta



Ereignisgesteuertes Schreiben einer Variablen vom Typ DWORD.

Der Funktionsbaustein *FB_WriteDWordOnDelta()* überprüft zyklisch, ob sich der Wert am Eingang *nSignal* geändert hat. Die Zykluszeit, für die Überprüfung wird durch den Parameter *tCycleTime* bestimmt. Wird für *tCycleTime* 0 sec angegeben, so wird in jedem SPS-Zyklus das Eingangssignal überprüft. Wird bei dem Vergleich festgestellt, dass der aktuelle Wert um *nUpperLimit* größer ist oder um den Wert *nLowerLimit* kleiner ist, so wird der Wert des Signals an das angegebene ADS-Gerät verschickt. Adressiert wird der Empfänger durch die AMS-NetId und durch die Portnummer (siehe auch Identifikation ADS-Gerät). Die Position innerhalb des Empfängers wird durch die Index-Group/Index-Offset oder dem Symbolnamen angegeben. Üblicherweise ist dieses das Eingangsabbild oder der Merkerbereich.

Wird der Eingang *bEnable* auf FALSE gesetzt, so erfolgt keine weitere Übertragung des Signals.

VAR_INPUT

```
bEnable    : BOOL := FALSE;
sNetId     : T_AmsNetId;
nPort      : T_AmsPort := AMSPORT_R0_PLC_RTS1;
nIdxGrp    : UDINT;
nIdxOffs   : UDINT;
sVarName   : STRING;
```

```
nSignal      : DWORD;
nLowerLimit  : DWORD;
nUpperLimit  : DWORD;
tCycleTime   : TIME := t#0s;
bSendNow     : BOOL;
```

bEnable : Baustein Freischalten.	sNetId : AMS-NetId des ADS-Gerätes, nach dem Wert übertragen werden soll.	nPort : AMS-Portnummer des ADS-Gerätes, nach dem Wert übertragen werden soll.	nIdxGrp : Index-Group innerhalb des ADS-Gerätes, in der Wert übertragen werden soll.	nIdxOffs : Index-Offset innerhalb des ADS-Gerätes, in der Wert übertragen werden soll.	sVarName : Symbolname innerhalb des ADS-Gerätes, in der Wert übertragen werden soll.	nSignal : Variable, dessen Wert übertragen werden soll.	nLowerLimit : unterer Grenzwert.	nUpperLimit : oberer Grenzwert.	tCycleTime : Zykluszeit, in der überprüft wird, ob das Eingangssignal die Grenzwerte überschritten hat.	bSendNow : Durch eine positive Flanke wird der Wert unmittelbar übertragen.
--	--	--	---	---	---	--	---	--	--	--

VAR_OUTPUT

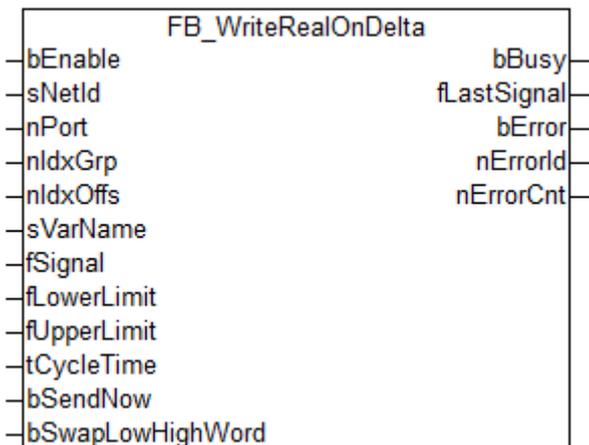
```
bBusy        : BOOL;
nLastSignal  : DWORD;
bError       : BOOL;
nErrorId     : UDINT;
nErrorCnt    : UDINT;
```

bBusy : Übertragung ist aktiv.	nLastSignal : zuletzt übertragender Wert.	bError : Bei der Übertragung ist ein Fehler aufgetreten.	nErrorId : Fehlernummer, falls ein Fehler aufgetreten ist.	nErrorCnt : Anzahl der fehlerhaften Übertragungsversuche.
---------------------------------------	--	---	---	--

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64	PC/CX, BX oder BC	TcDataExchange-Bibliothek ab V1.0.0

3.1.7 FB_WriteRealOnDelta



Ereignisgesteuertes Schreiben einer Variablen vom Typ REAL.

Der Funktionsbaustein *FB_WriteRealOnDelta()* überprüft zyklisch, ob sich der Wert am Eingang *fSignal* geändert hat. Die Zykluszeit, für die Überprüfung wird durch den Parameter *tCycleTime* bestimmt. Wird für *tCycleTime* 0 sec angegeben, so wird in jedem SPS-Zyklus das Eingangssignal überprüft. Wird bei dem

Vergleich festgestellt, dass der aktuelle Wert um *fUpperLimit* größer ist oder um den Wert *fLowerLimit* kleiner ist, so wird der Wert des Signals an das angegebene ADS-Gerät verschickt. Adressiert wird der Empfänger durch die AMS-NetId und durch die Portnummer (siehe auch Identifikation ADS-Gerät). Die Position innerhalb des Empfängers wird durch die Index-Group/Index-Offset oder dem Symbolnamen angegeben. Üblicherweise ist dieses das Eingangsabbild oder der Merkerbereich.

Je nach verwendeter Hardware ist die interne Darstellung von Fließkommazahlen unterschiedlich. Während Intel das 'little endian'-Format benutzt, wird bei Motorola-basierter Hardware das 'big-endian'-Format angewendet. Um einen Austausch von Fließkommazahlen zu ermöglichen, kann durch die Eingangsvariable *bSwapLowHighWord* eine Anpassung durchgeführt werden. Dieses ist z.B. nötig, wenn zwischen der TwinCAT SPS auf einem PC und einen BC9000 Fließkommazahlen ausgetauscht werden sollen.

Wird der Eingang *bEnable* auf FALSE gesetzt, so erfolgt keine weitere Übertragung des Signals.

VAR_INPUT

```
bEnable      : BOOL := FALSE;
sNetId       : T_AmsNetId;
nPort        : T_AmsPort := AMSPORT_RO_PLC_RTS1;
nIdxGrp      : UDINT;
nIdxOffs     : UDINT;
sVarName     : STRING;
fSignal      : REAL;
fLowerLimit  : REAL;
fUpperLimit  : REAL;
tCycleTime   : TIME := t#0s;
bSendNow     : BOOL;
bSwapLowHighWord : BOOL := FALSE;
```

bEnable: Baustein Freischalten.	sNetId: AMS-NetId des ADS-Gerätes, nach dem Wert übertragen werden soll.	nPort: AMS-Portnummer des ADS-Gerätes, nach dem Wert übertragen werden soll.	nIdxGrp: Index-Group innerhalb des ADS-Gerätes, in der Wert übertragen werden soll.	nIdxOffs: Index-Offset innerhalb des ADS-Gerätes, in der Wert übertragen werden soll.	sVarName: Symbolname innerhalb des ADS-Gerätes, in der Wert übertragen werden soll.	fSignal: Variable, dessen Wert übertragen werden soll.	fLowerLimit: unterer Grenzwert.	fUpperLimit: oberer Grenzwert.	tCycleTime: Zykluszeit, in der überprüft wird, ob das Eingangssignal die Grenzwerte überschritten hat.	bSendNow: Durch eine positive Flanke wird der Wert unmittelbar übertragen.	bSwapLowHighWord: Das niederwertige und das höherwertige WORD werden gedreht.
---	--	--	---	---	---	--	---	--	--	--	---

VAR_OUTPUT

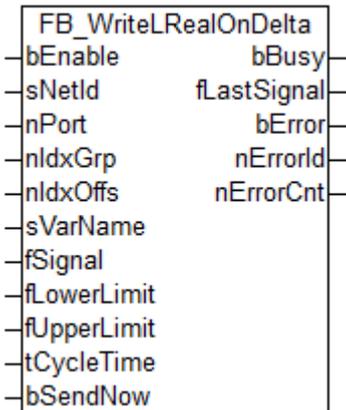
```
bBusy        : BOOL;
fLastSignal  : REAL;
bError       : BOOL;
nErrorId     : UDINT;
nErrorCnt    : UDINT;
```

bBusy: Übertragung ist aktiv.	fLastSignal: zuletzt übertragender Wert.	bError: Bei der Übertragung ist ein Fehler aufgetreten.	nErrorId: Fehlernummer, falls ein Fehler aufgetreten ist.	nErrorCnt: Anzahl der fehlerhaften Übertragungsversuche.
--------------------------------------	---	--	--	---

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64	PC/CX, BX oder BC	TcDataExchange-Bibliothek ab V1.0.0

3.1.8 FB_WriteLRealOnDelta



Ereignisgesteuertes Schreiben einer Variablen vom Typ LREAL.

Der Funktionsbaustein *FB_WriteLRealOnDelta()* überprüft zyklisch, ob sich der Wert am Eingang *fSignal* geändert hat. Die Zykluszeit, für die Überprüfung wird durch den Parameter *tCycleTime* bestimmt. Wird für *tCycleTime* 0 sec angegeben, so wird in jedem SPS-Zyklus das Eingangssignal überprüft. Wird bei dem Vergleich festgestellt, das der aktuelle Wert um *fUpperLimit* größer ist oder um den Wert *fLowerLimit* kleiner ist, so wird der Wert des Signals an das angegebende ADS-Gerät verschickt. Adressiert wird der Empfänger durch die AMS-NetId und durch die Portnummer (siehe auch [Identifikation ADS-Gerät](#)). Die Position innerhalb des Empfängers wird durch die Index-Group/Index-Offset oder dem Symbolnamen angegeben. Üblicherweise ist dieses das Eingangsabbild oder der Merkerbereich.

Wird der Eingang *bEnable* auf FALSE gesetzt, so erfolgt keine weitere Übertragung des Signals.

VAR_INPUT

```

bEnable      : BOOL := FALSE;
sNetId       : T_AmsNetId;
nPort        : T_AmsPort := AMSPORT_R0_PLC_RTS1;
nIdxGrp      : UDINT;
nIdxOffs     : UDINT;
sVarName     : STRING;
fSignal      : LREAL;
fLowerLimit  : LREAL;
fUpperLimit  : LREAL;
tCycleTime   : TIME := t#0s;
bSendNow     : BOOL;
    
```

bEnable	sNetId:	nPort:	nIdxGrp	nIdxOffs	sVarName:	fSignal:	fLowerLimit:	fUpperLimit:	tCycleTime:	bSendNow:
: Baustein Freischalten.	AMS-NetId des ADS-Gerätes, nach dem der Wert übertragen werden soll.	AMS-Portnummer des ADS-Gerätes, nach dem der Wert übertragen werden soll.	: Index-Group innerhalb b des ADS-Gerätes, in der der Wert übertragen werden soll.	: Index-Offset innerhalb b des ADS-Gerätes, in der der Wert übertragen werden soll.	Symbolname innerhalb b des ADS-Gerätes, in der der Wert übertragen werden soll.	Variable, dessen Wert übertragen werden soll.	unterer Grenzwert.	oberer Grenzwert.	Zykluszeit, in der überprüft wird, ob das Eingangssignal die Grenzwerte überschritten hat.	Durch eine positive Flanke wird der Wert unmittelbar übertragen.

VAR_OUTPUT

```

bBusy        : BOOL;
fLastSignal  : LREAL;
bError       : BOOL;
nErrorId     : UDINT;
nErrorCnt    : UDINT;
    
```

bBusy: Übertragung ist aktiv.	fLastSignal: zuletzt übertragender Wert.	bError: Bei der Übertragung ist ein Fehler aufgetreten.	nErrorId: Fehlernummer, falls ein Fehler aufgetreten ist.	nErrorCnt: Anzahl der fehlerhaften Übertragungsversuche.
--------------------------------------	---	--	--	---

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64	PC/CX	TcDataExchange-Bibliothek ab V1.0.0

3.2 Überwachungsbausteine

Name	Beschreibung
FB_CheckWatchdog [▶ 19]	Überwacht das empfangende Watchdogsignal
FB_WriteWatchdog [▶ 20]	Schreibt zyklisch ein Watchdogsignal (inkrementierender Zähler)

3.2.1 FB_CheckWatchdog



Überwachen eines Watchdogsignals, welches mit dem Baustein [FB_WriteWatchdog\(\) \[▶ 20\]](#) übertragen wird.

Das zu überwachende Gerät verschickt regelmäßig einen sich ändernden Zählerstand an das Gerät, welches die Übertragung kontrollieren soll. Dort wird mit Hilfe des Funktionsbausteins *FB_CheckWatchdog()* der Zählerstand überwacht.

Ändert sich dieser innerhalb einer bestimmten Zeit nicht, so wird der Ausgang *bWatchdog* auf TRUE gesetzt. Wird für *tWatchdogTime* ein Wert von 0 sec angegeben, so wird das Signal *bWatchdog* auf FALSE gesetzt. Die Zeit *tWatchdogTime* sollte ein vielfaches (5-10fach) der Zeit sein, in der das Überwachungssignal zugeschickt wird.

VAR_INPUT

```

bEnable      : BOOL := FALSE;
tWatchdogTime : TIME := t#0s;
nCnt         : UDINT;
  
```

bEnable: Baustein Freischalten.	tWatchdogTime: Zeitdauer, in der sich <i>nCnt</i> ändern muß.	nCnt: aktueller Zählerstand des Watchdogsignals
--	--	--

VAR_OUTPUT

```

bWatchdog    : BOOL;
nLastCnt     : UDINT;
  
```

bWatchdog: FALSE signalisiert ein gültiges Überwachungssignal. Wird für die Zeitdauer <i>tWatchdogTime</i> keine Änderung von <i>nCnt</i> erkannt, so geht der Ausgang auf TRUE.	nLastCnt: letzter, erfolgreich übertragender Zählerstand des Überwachungssignals.
---	--

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64	PC/CX, BX oder BC	TcDataExchange-Bibliothek ab V1.0.0

3.2.2 FB_WriteWatchdog



Schreiben eines Watchdogsignals in ein anderes ADS-Gerät (TwinCAT PLC, Busklemmen Controller, ...).

Der Funktionsbaustein *FB_WriteWatchdog()* schreibt zyklisch einen 32-Bit Zähler in einen anderes ADS-Gerät. Bei jeder erfolgreichen Übertragung wird der Zähler inkrementiert. Beim Empfänger kann mit Hilfe des Bausteine *FB_CheckWatchdog()* [► 19] dieses Signal ausgewertet werden. Adressiert wird der Empfänger durch die AMS-NetId und durch die Portnummer (siehe auch *Identifikation ADS-Gerät*). Die Position innerhalb des Empfängers wird durch die Index-Group/Index-Offset oder dem Symbolnamen angegeben. Üblicherweise ist dieses das Eingangsabbild oder der Merkerbereich.

Damit der Zählerstand nicht zu häufig übertragen wird, sollte die Zeit *tWatchdogTime* nicht kleiner als 1 sec sein. Wird für *tWatchdogTime* 0 sec angegeben, so wird die Übertragung des Signals gesperrt. Beachten sie bitte auch die Beschreibung vom Baustein *FB_CheckWatchdog()* [► 19].

Wird der Eingang *bEnable* auf FALSE gesetzt, so erfolgt keine weitere Übertragung des Watchdogsignals.

VAR_INPUT

```

bEnable      : BOOL := FALSE;
sNetId       : T_AmsNetId;
nPort        : T_AmsPort := AMSPORT_R0_PLC_RTSl;
nIdxGrp      : UDINT;
nIdxOffs     : UDINT;
sVarName     : STRING;
tWatchdogTime: TIME := t#0s;
bSendNow     : BOOL;
    
```

bEnable:	sNetId:	nPort:	nIdxGrp:	nIdxOffs:	sVarName:	tWatchdog Time:	bSendNow:
Baustein Freischalten	AMS-NetId des ADS-Gerätes, nach dem der Wert übertragen werden soll.	AMS-Portnummer des ADS-Gerätes, nach dem der Wert übertragen werden soll.	Index-Group innerhalb des ADS-Gerätes, in der der Wert übertragen werden soll.	Index-Offset innerhalb des ADS-Gerätes, in der der Wert übertragen werden soll.	Symbolname innerhalb des ADS-Gerätes, in der der Wert übertragen werden soll.	Zykluszeit, in der das Watchdogsignal übertragen wird.	Durch eine positive Flanke wird der Wert unmittelbar übertragen.

VAR_OUTPUT

```

bBusy        : BOOL;
nLastCnt     : UDINT;
bError       : BOOL;
nErrorId     : UDINT;
    
```

bBusy: Übertragung ist aktiv.	nLastCnt: zuletzt übertragender Zählerstand.	bError: Bei der Übertragung ist ein Fehler aufgetreten.	nErrorId: Fehlernummer, falls ein Fehler aufgetreten ist.
--------------------------------------	---	--	--

Voraussetzungen

Entwicklungsumgebung	Zielsystem	erforderliche Bibliotheken
TwinCAT 2.11 R3/x64	PC/CX, BX oder BC	TcDataExchange-Bibliothek ab V1.0.0

3.3 Datentypen

3.3.1 E_AdsComMode

```

TYPE E_AdsComMode :
(
  eAdsComModeSecureCom := 0,
  eAdsComModeFastCom   := 1
);
END_TYPE
    
```

4 Anhang

4.1 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.

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 unseren Internetseiten: <https://www.beckhoff.de>

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(0)5246 963 157
Fax: +49(0)5246 963 9157
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(0)5246 963 460
Fax: +49(0)5246 963 479
E-Mail: service@beckhoff.com

Beckhoff Firmenzentrale

Beckhoff Automation GmbH & Co. KG

Hülshorstweg 20
33415 Verl
Deutschland

Telefon: +49(0)5246 963 0
Fax: +49(0)5246 963 198
E-Mail: info@beckhoff.com
Internet: <https://www.beckhoff.de>

Mehr Informationen:
www.beckhoff.de/tx1200

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

