

Handbuch | DE

TS8036

TwinCAT 2 | Crestron Server

Supplement |
Building Automation



Inhaltsverzeichnis

1	Vorwort	5
1.1	Hinweise zur Dokumentation	5
1.2	Sicherheitshinweise	6
1.3	Hinweise zur Informationssicherheit	7
2	Einführung	8
3	Hardwareanforderungen	9
4	Benutzeranforderungen	10
5	Funktionsweise	11
6	Performance Test	13
7	Installation	15
7.1	Installation unter Windows CE (WEC)	15
7.2	Installation unter Windows Embedded Standard (WES).....	15
7.3	Installation Crestron User Modules	16
8	Beckhoff SPS-Funktionsbausteine	17
8.1	FB_CrestronReadByte	17
8.2	FB_CrestronWriteByte	18
8.3	FB_CrestronWriteByteArray.....	19
8.4	FB_CrestronReadInt	19
8.5	FB_CrestronWriteInt	20
8.6	FB_CrestronWriteIntArray.....	21
8.7	FB_CrestronReadReal	22
8.8	FB_CrestronWriteReal	23
8.9	FB_CrestronGetClientConnectionState	23
9	Crestron SIMPL Windows User Modules	25
9.1	Beckhoff Communication	25
9.2	Beckhoff Communication 400	26
9.3	Beckhoff Read Bit	27
9.4	Beckhoff Read Bit By Name.....	28
9.5	Beckhoff Write Bit.....	29
9.6	Beckhoff Write Bit By Name	30
9.7	Beckhoff Read Byte	30
9.8	Beckhoff Read Byte By Name.....	31
9.9	Beckhoff Write Byte.....	32
9.10	Beckhoff Write Byte By Name.....	33
9.11	Beckhoff Read Int.....	34
9.12	Beckhoff Read Int By Name	35
9.13	Beckhoff Write Int.....	36
9.14	Beckhoff Write Int By Name	36
9.15	Beckhoff Switch.....	37
9.16	Beckhoff Switch By Name	38
10	Fehlercodes	39

11 Beispiele	40
11.1 Beckhoff SPS-Programmbeispiele	40
11.2 Crestron Programmierbeispiele	41

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.



Tipp oder Fingerzeig

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 Einführung

Voraussetzungen

Crestron Electronics, mit Hauptsitz in Rockleigh (www.crestron.de), New Jersey (USA), ist einer der Marktführer für Mediensteuerungssysteme. Neben der Steuerung von Audio- und Videoanlagen lassen sich auch gebäudetechnische Komponenten integrieren. Im Folgenden soll gezeigt werden, wie über Ethernet Daten zwischen Crestron Steuerungen und der Beckhoff Hardware ausgetauscht werden können.



Für die Crestron-Steuerungen stehen Bausteine (User Modules) zur Verfügung. Auf dem Beckhoff Controller muss der TwinCAT Crestron Server installiert werden. Ein Crestron Controller kann Schreib- und Lesebefehle aufrufen und ein Beckhoff Controller kann Daten in die Crestron-Steuerung schreiben oder herauslesen.

3 Hardwareanforderungen

Voraussetzung Hardware Beckhoff:

Der TwinCAT Crestron Server ist auf allen PC-Steuerungen, die kompatibel zu TwinCAT sind, einsetzbar (ausgenommen ist der CX9000).

Zur Installation des TwinCAT Crestron Server auf einer CX x86- oder CX ARM-Prozessor basierenden Hardware-Plattform wird folgendes benötigt / vorausgesetzt.

- TwinCAT PLC Runtime
- Microsoft .NET Framework / Compact .NET Framework.
- Ethernet-Schnittstelle

Bitte informieren Sie sich im Beckhoff Information System, welche Hardware mit welchem Image diese Anforderungen erfüllen.

Voraussetzung Hardware Crestron:

- Crestron Controller der 2-Serie mit Netzwerkschnittstelle (z.B. CP2E, MP2E etc.).
- Firmware Version 4.001.10.12 (Feb 17 2009) oder höher.

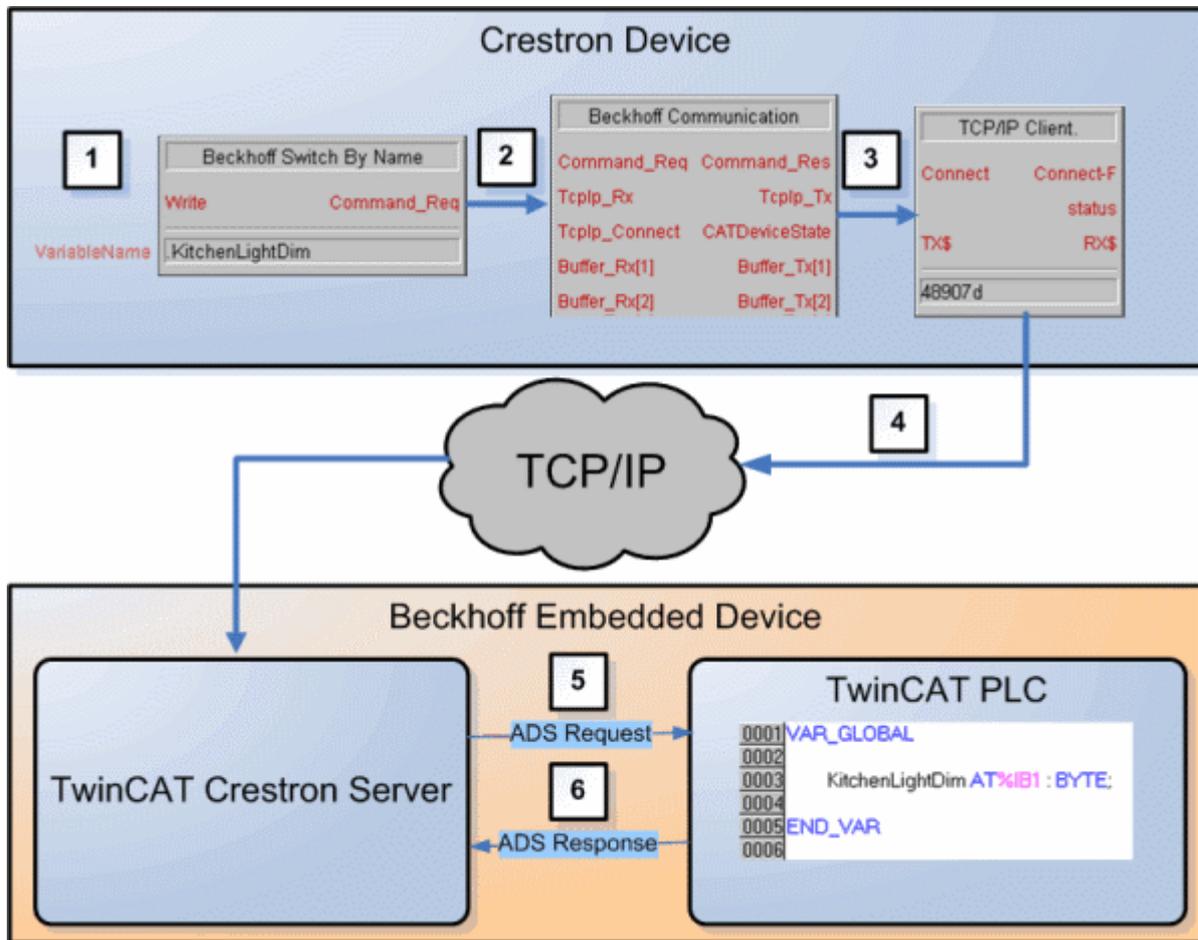
4 Benutzeranforderungen

Für den Nutzer dieses ADS Servers / dieser Bibliothek werden folgende Grundkenntnisse vorausgesetzt:

- Umgang mit dem TwinCAT PLC Control
- Umgang mit dem TwinCAT System Manager
- PC und Netzwerkkennntnisse
- Einschlägige Sicherheitsvorschriften der technischen Gebäudeausrüstung

5 Funktionsweise

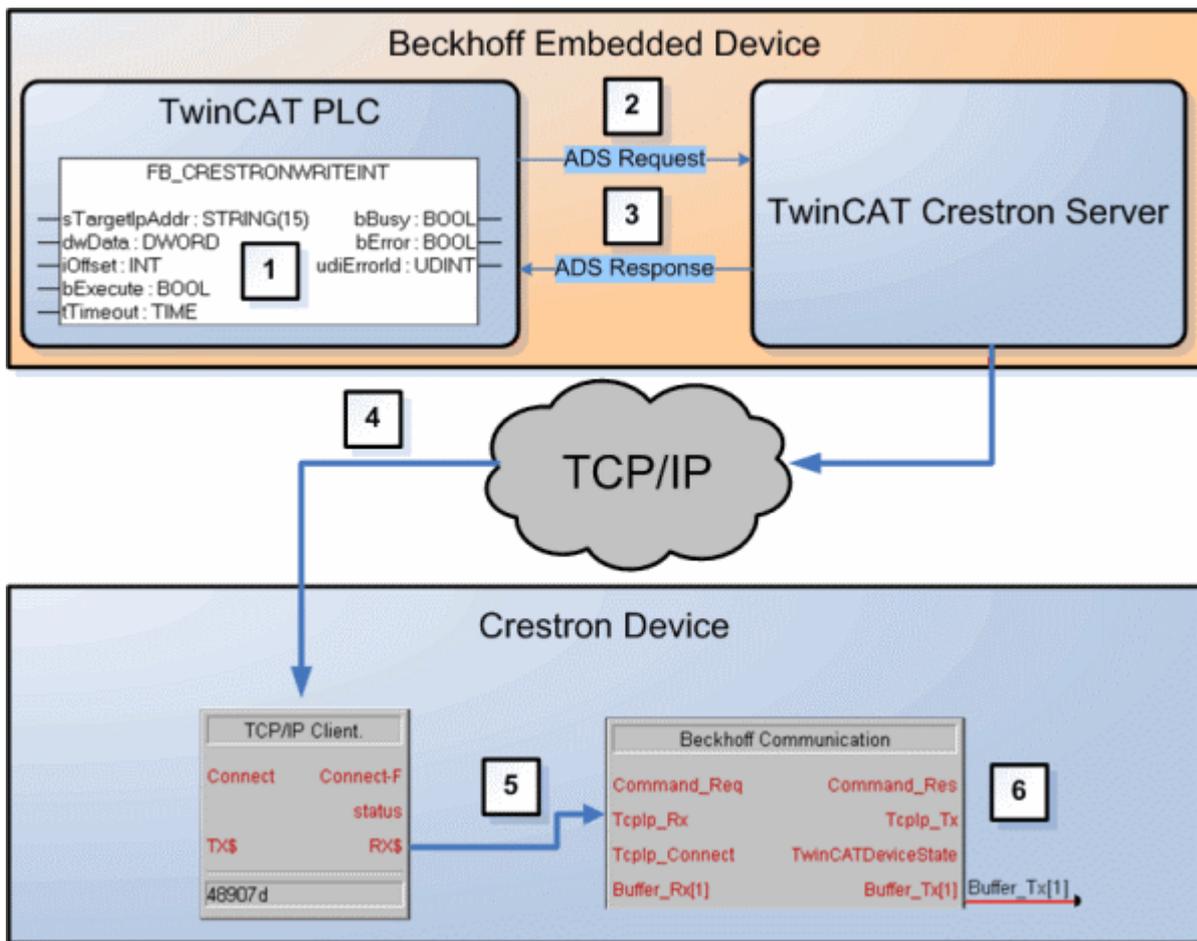
Im Ablaufdiagramm wird beschrieben, wie die einzelnen Arbeitsschritte eines Schreibbefehls von der Crestron Steuerung zu der TwinCAT SPS abgearbeitet werden.



Arbeitsschritte:

1. Das Schreibkommando wird am *Beckhoff Switch By Name* durch eine positive Flanke ausgelöst.
2. Es wird an das **Beckhoff Communication** Modul zur weiteren Ergänzung der Kommandoparameter weitergereicht.
3. Das erweiterte Schreibkommando wird an den TCP/IP Client zum Versenden überreicht.
4. Übertragung der Daten per TCP/IP.
5. Der *TwinCAT Crestron Server* nimmt das Schreibkommando entgegen, wandelt es in ein ADS Request-Telegram um und schickt die Daten zu der TwinCAT SPS.
6. Die SPS antwortet mit einem ADS Response, der vom **TwinCAT Crestron Server** entgegen genommen wird, um eventuell aufgetretene Fehler zu loggen.

Im Ablaufdiagramm wird beschrieben, wie die einzelnen Arbeitsschritte eines Schreibbefehls von der TwinCAT SPS zu der Crestron Steuerung abgearbeitet werden.



Arbeitsschritte:

1. Das Schreibkommando wird am *FB_CrestronWriteInt* durch eine positive Flanke ausgelöst.
2. Es wird als *ADS Request* an den *TwinCAT Crestron Server* geschickt. Der wandelt die empfangenen Daten in ein für Crestron verständliches TCP/IP Telegramm um und schickt dieses per TCP/IP zur Crestron Steuerung.
3. Als Antwort auf den *ADS Request* wird ein *ADS Response* erzeugt, der Infos über den Verlauf des Sendens trägt. So kann bei eventuell aufgetretenen Fehlern entsprechend reagiert werden.
4. Übertragung der Daten per TCP/IP.
5. Der Crestron *TCP/IP Client* nimmt die Daten entgegen und reicht sie zur Verarbeitung an das *Beckhoff Communication* Modul weiter.
6. Das **Beckhoff Communication** Modul reagiert auf die Daten vom TCP/IP Client und filtert die Informationen, um anschließend das angesprochene Signal mit dem übertragenen Wert zu beschreiben.

6 Performance Test

Die Ergebnisse des Tests sollten Aufschluss darüber geben, in welchem Zusammenhang die SPS-Zykluszeit zu der verwendeten Hardware steht. Dieses kann bei der Auswahl der notwendigen Hardware hilfreich sein.

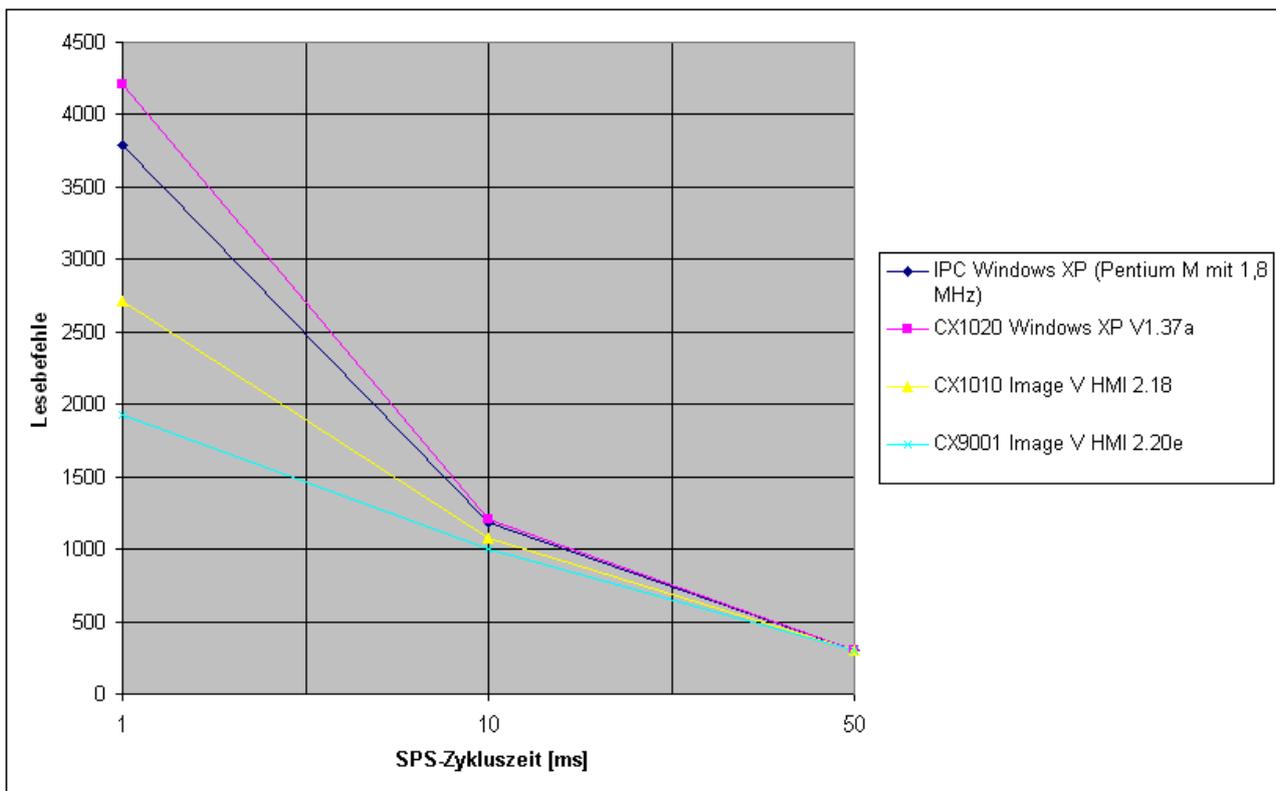


Der Test fand in einer dafür errichteten Testumgebung unter optimalen Bedingungen statt. Die somit gesammelten Ergebnisse sind daher nicht uneingeschränkt auf andere Umgebungen zu übertragen.

Aufbau

Eine Minute lang wurde mit einer SPS-Zykluszeit von **50, 10, 1** Millisekunden ein Lesebefehl auf einen 16 Bit Wert abgesetzt. Die Beckhoff SPS war dabei der lesende Teilnehmer. Die Anzahl der erfolgreich übertragenen Lesebefehle wurde in einem Zähler festgehalten. Daraus ergaben sich folgende Ergebnisse:

Ergebnisse



Gerätebeschreibung	SPS-Zykluszeit			Wertebeschreibung
	50ms	10ms	1ms	
IPC Windows XP (Pentium M mit 1,8 MHz) MP2E Firmware V4.001.1012 (Feb 17 2009) Build 7	0	0	0	Fehler
	299	1180	3785	Lesebefehle
	299	1180	3785	Erfolgreiche Lesebefehle
CX1020 Windows XP V1.37a MP2E Firmware V4.001.1012 (Feb 17 2009) Build 7	0	0	0	Fehler
	298	1208	4208	Lesebefehle
	298	1208	4208	Erfolgreiche Lesebefehle

CX1010 Image V HMI 2.18 MP2E Firmware V4.001.1012 (Feb 17 2009) Build 7	0	0	0	Fehler
	298	1078	2709	Lesebefehle
	298	1078	2709	Erfolgreiche Lesebefehle
CX9001 Image V HMI 2.20e MP2E Firmware V4.001.1012 (Feb 17 2009) Build 7	0	0	0	Fehler
	299	997	1928	Lesebefehle
	299	997	1928	Erfolgreiche Lesebefehle

Der Test zeigt, dass ein Reduzieren der SPS-Zykluszeit die Anzahl der Lesebefehle signifikant erhöht. Je höher die SPS-Zykluszeit ist, desto weniger hat die Leistungsfähigkeit der Hardware-Einfluss auf die Übertragungsgeschwindigkeit.

Wählen Sie die Hardware so aus, dass die SPS-Zykluszeit möglichst gering eingestellt werden kann. Auch kann durch das Verwenden einer weiteren, schnelleren Hintergrundtask, in der die Kommunikation mit der Crestron Steuerung ausgeführt wird, die Performance verbessern.



Der TwinCAT System Manager bietet Hilfsmittel zur Optimierung der SPS-Zykluszeit. Weitere Informationen hierzu finden Sie in der Dokumentation des TwinCAT System Managers.

Ihnen steht die SPS wie auch das SIMPL Windows Programm als Download zur Verfügung.

Beschreibung	Download
Beckhoff SPS zum zyklischen Lesen eines Wertes / SIMPL Windows Programm zum Bereitstellen eines Wertes.	https://infosys.beckhoff.com/content/1031/tcsample_crestron/Resources/11364625547.zip 

7 Installation

7.1 Installation unter Windows CE (WEC)

Übertragen der Installation auf das CE Gerät

Wurde der TwinCAT Crestron Server erfolgreich installiert, befinden sich im Ordner "**C:\TwinCAT\CE\Crestron Server**" zwei Unterordner.

- **\ARMV4I**
 - Dieser Ordner beinhaltet die CE-Installation:
TcCrestronServerCe.ARMV4I.CAB für ARMV4 CPUs (z.B. CX900x)
- **\I586**
 - Dieser Ordner beinhaltet die CE-Installation:
TcCrestronServerCe.I586.CAB für X86 CPUs (z.B. CX10xx)

Übertragen Sie die Datei "**TcCrestronServerCe.ARMV4I.CAB**" oder "**TcCrestronServerCe.I586.CAB**" auf das Windows CE Gerät. Das kann via

- einem frei gegebenem Ordner (Public folder)
- FTP
- ActiveSync
- Compact Flash (CF) Adapters

erfolgen.

Installation auf dem Windows CE Gerät

Die Installation kann mit einem Doppelklick auf die CAB-Datei ausgeführt werden. Bestätigen Sie das vorgewählte (default) Verzeichnis "**\\hard disk\System\TC-Crestron-Server**" mit "OK".

Nach der Installation wird die CAB-Datei automatisch gelöscht. Starten Sie das Gerät nach Beendigung der Installation neu.

7.2 Installation unter Windows Embedded Standard (WES)

Starten der Installation

Starten Sie die Installation und folgen Sie den Anweisungen in den Dialogen.

Nach der Installation

Ist die Installation beendet, befinden sich in Ihrem TwinCAT Ordner zwei neue Verzeichnisse "**..\TwinCAT\CE\Crestron Server**" und "**..\TwinCAT\Crestron Server**".

Im Verzeichnis "**..\TwinCAT\CE\Crestron Server**" finden Sie die Installationsdateien (.CAB) für die

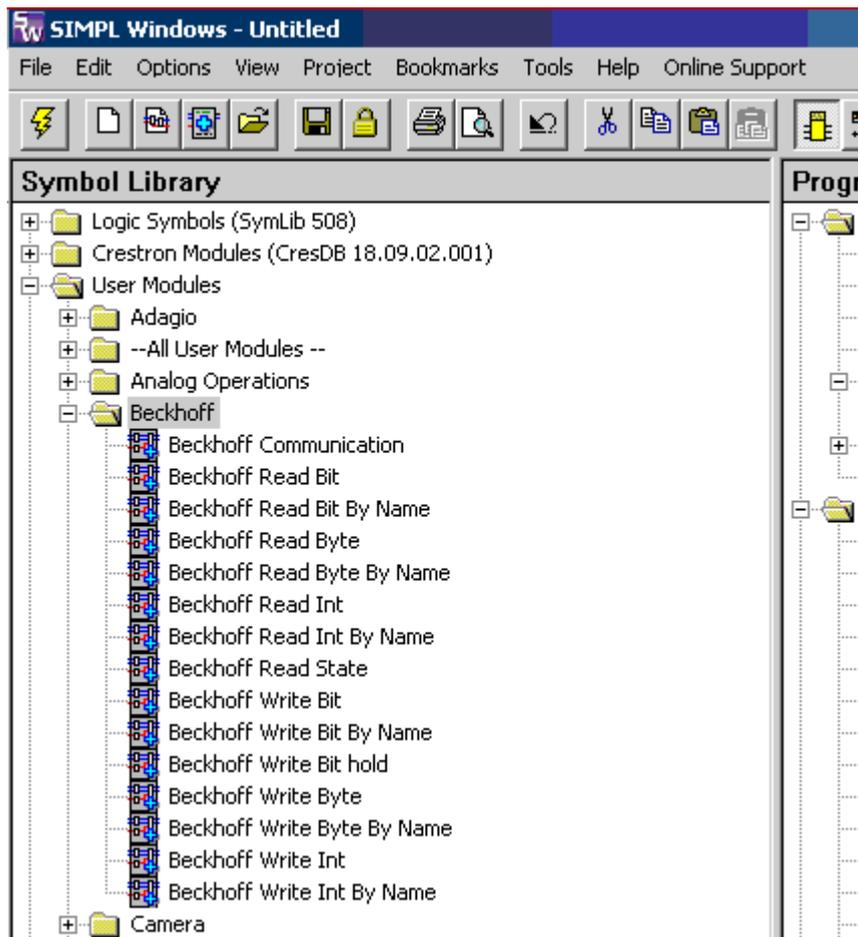
verschiedenen Windows CE Plattformen: ARM und X86.

7.3 Installation Crestron User Modules

Die Installation besteht aus mehreren User Modules. Diese sind aufgeteilt in SIMPL+ Module und User Macros. Die SIMPL+ Module haben die Endung USH und USP. Die User Macros die Endung UMC. Die User Modules befinden sich nach der Installation des TwinCAT Crestron Servers in dem folgenden Ordner: `..\TwinCAT\Crestron Server\CrestronModule\BeckhoffCommunication.zip`. Entpacken Sie die Datei und kopieren Sie diese in die folgenden Verzeichnisse:

Dateien	Verzeichnis
*.umc	<Crestron-Program-Folder>\Simpl\Usrmacro
*.ush und *.usp	<Crestron-Program-Folder>\Simpl\Usrplus

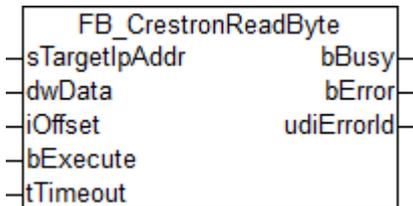
In SIMPL Window sind die einzelnen User Modules unter **aufgelistet.Symbol Library -> User Modules -> Beckhoff**



8 Beckhoff SPS-Funktionsbausteine

Name	Funktion
FB_CrestronWriteByte [► 18]	Schreiben eines Wertes vom Typ Byte (8 Bit).
FB_CrestronWriteByteArray [► 19]	Schreiben von mehreren Werten (Array) vom Typ Byte (8 Bit).
FB_CrestronReadByte [► 17]	Lesen eines Wertes vom Typ Byte (8 Bit).
FB_CrestronWriteInt [► 20]	Schreiben eines Wertes vom Typ Integer (16 Bit).
FB_CrestronWriteIntArray [► 21]	Schreiben von mehreren Werten (Array) vom Typ Integer (16 Bit).
FB_CrestronReadInt [► 19]	Lesen eines Wertes vom Typ Integer (16 Bit).
FB_CrestronWriteReal [► 23]	Schreiben eines Wertes vom Typ Real (32 Bit).
FB_CrestronReadReal [► 22]	Lesen eines Wertes vom Typ Real (32 Bit).
FB_CrestronGetClientConnectionState [► 23]	Lesen eines Wertes vom Typ Real (32 Bit).

8.1 FB_CrestronReadByte



Dieser Funktionsblock erlaubt es, ein Byte (8 Bit) aus der Crestron Steuerung zu lesen.

VAR_INPUT

```
sTargetIpAddr : STRING(15);
dwData        : DWORD;
iOffset       : INT;
bExecute      : BOOL;
tTimeout      : TIME := T#5s;
```

sTargetIpAddr: IP-Adresse der Crestron Steuerung.

dwData: Adresspointer auf das Byte (8 Bit) in welches der gelesene Wert abgelegt wird.

iOffset: Gibt an, ab welcher Position im *Buffer_Tx* aus der Crestron Steuerung ausgelesen werden soll.

bExecute: Eine positive Flanke startet den Baustein.

tTimeout: Gibt die Zeitdauer an, in der auf eine Rückantwort gewartet wird.

VAR_OUTPUT

```
bBusy        : BOOL;
bError       : BOOL;
udiErrorId   : UDINT;
```

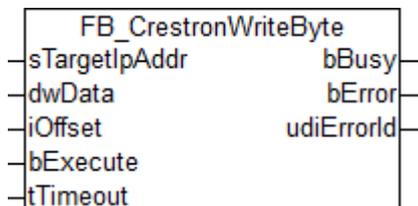
bBusy: Während des Lesevorgangs wird der Ausgang *bBusy* auf TRUE gesetzt und bleibt so lange TRUE, bis das Lesen abgeschlossen wurde. Während dieser Zeit können über diesen Baustein keine weiteren Befehle aktiviert werden.

bError: Der Ausgang ist TRUE, sobald beim Ausführen des Befehls ein Fehler auftritt.

udiErrorId: Enthält den Fehlercode [\[► 39\]](#), der bei der Ausführung des Bausteins auftreten kann. Ist kein Fehler aufgetreten, so ist der Wert 0.

Voraussetzungen

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

8.2 FB_CrestronWriteByte

Dieser Funktionsblock erlaubt es, ein Byte (8 Bit) in die Crestron Steuerung zu schreiben.

VAR_INPUT

```
sTargetIpAddr : STRING(15);
dwData       : DWORD;
iOffset      : INT;
bExecute     : BOOL;
tTimeout     : TIME := T#5s;
```

sTargetIpAddr: IP-Adresse der Crestron Steuerung.

dwData: Adresspointer auf das Byte (8 Bit) aus welchem der zu schreibende Wert gelesen wird.

iOffset: Gibt an, ab welcher Position im *Buffer_Rx* in die Crestron Steuerung geschrieben werden soll.

bExecute: Eine positive Flanke startet den Baustein.

tTimeout: Gibt die Zeitdauer an, in der auf eine Rückantwort gewartet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
udiErrorId  : UDINT;
```

bBusy: Während des Schreibvorgangs wird der Ausgang *bBusy* auf TRUE gesetzt und bleibt so lange TRUE, bis das Schreiben abgeschlossen wurde. Während dieser Zeit können über diesen Baustein keine weiteren Befehle aktiviert werden.

bError: Der Ausgang ist TRUE, sobald beim Ausführen des Befehls ein Fehler auftritt.

udiErrorId: Enthält den Fehlercode [► 39], der bei der Ausführung des Bausteins auftreten kann. Ist kein Fehler aufgetreten, so ist der Wert 0.

Voraussetzungen

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

8.3 FB_CrestronWriteByteArray



Dieser Funktionsblock erlaubt es, ein Byte Array mit einer maximalen Länge von 40 Feldern in die Crestron Steuerung zu schreiben.

VAR_INPUT

```

sTargetIpAddr : STRING(15);
dwData        : DWORD;
iOffset       : INT;
iLen          : INT;
bExecute      : BOOL;
tTimeout      : TIME := T#5s;
  
```

sTargetIpAddr: IP-Adresse der Crestron Steuerung.

dwData: Adresspointer auf das Byte Array.

iOffset: Gibt an, ab welcher Position im *Buffer_Rx* in die Crestron Steuerung geschrieben werden soll.

iLen: Gibt die Länge des Arrays an, dass geschrieben werden soll. Die Länge ist auf 40 beschränkt.

bExecute: Eine positive Flanke startet den Baustein.

tTimeout: Gibt die Zeitdauer an, in der auf eine Rückantwort gewartet wird.

VAR_OUTPUT

```

bBusy        : BOOL;
bError       : BOOL;
udiErrorId   : UDINT;
  
```

bBusy: Während des Schreibvorgangs wird der Ausgang *bBusy* auf TRUE gesetzt und bleibt so lange TRUE, bis das Schreiben abgeschlossen wurde. Während dieser Zeit können über diesen Baustein keine weiteren Befehle aktiviert werden.

bError: Der Ausgang ist TRUE, sobald beim Ausführen des Befehls ein Fehler auftritt.

udiErrorId: Enthält den Fehlercode [► 39], der bei der Ausführung des Bausteins auftreten kann. Ist kein Fehler aufgetreten, so ist der Wert 0.

Voraussetzungen

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

8.4 FB_CrestronReadInt



Dieser Funktionsblock erlaubt es, einen Integer (16 Bit) aus der Crestron Steuerung zu lesen.

VAR_INPUT

```
sTargetIpAddr : STRING(15);
dwData        : DWORD;
iOffset       : INT;
bExecute      : BOOL;
tTimeout      : TIME := T#5s;
```

sTargetIpAddr: IP-Adresse der Crestron Steuerung.

dwData: Adresspointer auf den Integer worein der gelesene Wert abgelegt wird.

iOffset: Gibt an, ab welcher Position im *Buffer_Tx* aus der Crestron Steuerung ausgelesen werden soll.

bExecute: Eine positive Flanke startet den Baustein.

tTimeout: Gibt die Zeitdauer an, in der auf eine Rückantwort gewartet wird.

VAR_OUTPUT

```
bBusy        : BOOL;
bError       : BOOL;
udiErrorId   : UDINT;
```

bBusy: Während des Lesevorgangs wird der Ausgang *bBusy* auf TRUE gesetzt und bleibt so lange TRUE, bis das Lesen abgeschlossen wurde. Während dieser Zeit können über diesen Baustein keine weiteren Befehle aktiviert werden.

bError: Der Ausgang ist TRUE, sobald beim Ausführen des Befehls ein Fehler auftritt.

udiErrorId: Enthält den Fehlercode [► 39], der bei der Ausführung des Bausteins auftreten kann. Ist kein Fehler aufgetreten, so ist der Wert 0.

Voraussetzungen

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

8.5 FB_CrestronWriteInt



Dieser Funktionsblock erlaubt es, einen Integer (16 Bit) in die Crestron Steuerung zu schreiben.

VAR_INPUT

```
sTargetIpAddr : STRING(15);
dwData        : DWORD;
iOffset       : INT;
bExecute      : BOOL;
tTimeout      : TIME := T#5s;
```

sTargetIpAddr: IP-Adresse der Crestron Steuerung.

dwData: Adresspointer auf den Integer (16 Bit) woraus der Wert zum Schreiben gelesen wird.

iOffset: Gibt an, ab welcher Position im *Buffer_Rx* in die Crestron Steuerung geschrieben werden soll.

bExecute: Eine positive Flanke startet den Baustein.

tTimeout: Gibt die Zeitdauer an, in der auf eine Rückantwort gewartet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
udiErrorId : UDINT;
```

bBusy: Während des Schreibvorgangs wird der Ausgang *bBusy* auf TRUE gesetzt und bleibt so lange TRUE, bis das Schreiben abgeschlossen wurde. Während dieser Zeit können über diesen Baustein keine weiteren Befehle aktiviert werden.

bError: Der Ausgang ist TRUE, sobald beim Ausführen des Befehls ein Fehler auftritt.

udiErrorId: Enthält den Fehlercode [► 39], der bei der Ausführung des Bausteins auftreten kann. Ist kein Fehler aufgetreten, so ist der Wert 0.

Voraussetzungen

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

8.6 FB_CrestronWriteIntArray



Dieser Funktionsblock erlaubt es, ein Integer Array mit einer maximalen Länge von 40 Feldern in die Crestron Steuerung zu schreiben.

VAR_INPUT

```
sTargetIpAddr : STRING(15);
dwData        : DWORD;
iOffset       : INT;
iLen          : INT;
bExecute      : BOOL;
tTimeout      : TIME := T#5s;
```

sTargetIpAddr: IP-Adresse der Crestron Steuerung.

dwData: Adresspointer auf das Integer Array.

iOffset: Gibt an, ab welcher Position im *Buffer_Rx* in die Crestron Steuerung geschrieben werden soll.

iLen: Gibt die Länge des Arrays an, das geschrieben werden soll. Die Länge ist auf 40 beschränkt.

bExecute: Eine positive Flanke startet den Baustein.

tTimeout: Gibt die Zeitdauer an, in der auf eine Rückantwort gewartet wird.

VAR_OUTPUT

```
bBusy      : BOOL;
bError     : BOOL;
udiErrorId : UDINT;
```

bBusy: Während des Schreibvorgangs wird der Ausgang *bBusy* auf TRUE gesetzt und bleibt so lange TRUE, bis das Schreiben abgeschlossen wurde. Während dieser Zeit können über diesen Baustein keine weiteren Befehle aktiviert werden.

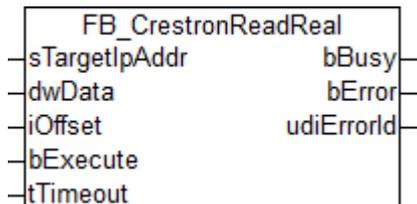
bError: Der Ausgang ist TRUE, sobald beim Ausführen des Befehls ein Fehler auftritt.

udiErrorId: Enthält den Fehlercode [▶ 39], der bei der Ausführung des Bausteins auftreten kann. Ist kein Fehler aufgetreten, so ist der Wert 0.

Voraussetzungen

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

8.7 FB_CrestronReadReal



Dieser Funktionsblock erlaubt es, einen Real (32 Bit) aus der Crestron Steuerung zu lesen.



Der Wert wird in der Crestron Steuerung auf 4 Bytes verteilt. Das bedeutet, man muss vier Platzhalter im Buffer_Tx Array des Beckhoff Communication [▶ 25] oder Beckhoff Communication 400 [▶ 26] Modul benutzen, um einen Variable vom Typ Real passend abzubilden.

VAR_INPUT

```

sTargetIpAddr : STRING(15);
dwData        : DWORD;
iOffset       : INT;
bExecute      : BOOL;
tTimeout      : TIME := T#5s;
  
```

sTargetIpAddr: IP-Adresse der Crestron Steuerung.

dwData: Adresspointer auf den Real (32 Bit) worin der gelesene Wert abgelegt wird.

iOffset: Gibt an, ab welcher Position im *Buffer_Tx* aus der Crestron Steuerung gelesen werden soll.

bExecute: Eine positive Flanke startet den Baustein.

tTimeout: Gibt die Zeitdauer an, in der auf eine Rückantwort gewartet wird.

VAR_OUTPUT

```

bBusy         : BOOL;
bError        : BOOL;
udiErrorId    : UDINT;
  
```

bBusy: Während des Lesevorgangs wird der Ausgang *bBusy* auf TRUE gesetzt und bleibt so lange TRUE, bis das Lesen abgeschlossen wurde. Während dieser Zeit können über diesen Baustein keine weiteren Befehle aktiviert werden.

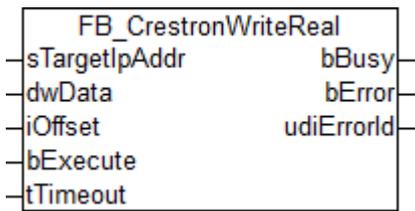
bError: Der Ausgang ist TRUE, sobald beim Ausführen des Befehls ein Fehler auftritt.

udiErrorId: Enthält den Fehlercode [▶ 39], der bei der Ausführung des Bausteins auftreten kann. Ist kein Fehler aufgetreten, so ist der Wert 0.

Voraussetzungen

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

8.8 FB_CrestronWriteReal



Dieser Funktionsblock erlaubt es, einen Real (32 Bit) in die Crestron Steuerung zu schreiben.



Der Wert wird in der Crestron Steuerung auf 4 Bytes verteilt. Es werden 4 Platzhalter im Buffer_Rx vom [Beckhoff Communication](#) [► 25] oder [Beckhoff Communication 400](#) [► 26] Modul benötigt. Der Eingang iOffset beschreibt, an welcher Stelle des Buffer_Rx Array das Schreiben der 4 Bytes beginnt.

VAR_INPUT

```
sTargetIpAddr : STRING(15);
dwData       : DWORD;
iOffset      : INT;
bExecute     : BOOL;
tTimeout     : TIME := T#5s;
```

sTargetIpAddr: IP-Adresse der Crestron Steuerung.

dwData: Adresspointer auf den Real (32 Bit) aus welchem der zu schreibende Wert gelesen wird.

iOffset: Gibt an, ab welcher Position im *Buffer_Rx* in die Crestron Steuerung geschrieben werden soll.

bExecute: Eine positive Flanke startet den Baustein.

tTimeout: Gibt die Zeitdauer an, in der auf eine Rückantwort gewartet wird.

VAR_OUTPUT

```
bBusy       : BOOL;
bError      : BOOL;
udiErrorId  : UDINT;
```

bBusy: Während des Schreibvorgangs wird der Ausgang *bBusy* auf TRUE gesetzt und bleibt so lange TRUE, bis das Schreiben abgeschlossen wurde. Während dieser Zeit können über diesen Baustein keine weiteren Befehle aktiviert werden.

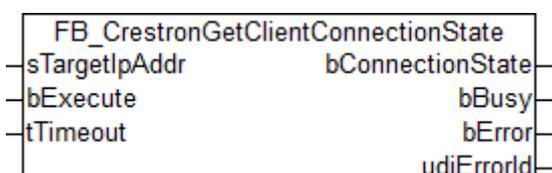
bError: Der Ausgang ist TRUE, sobald beim Ausführen des Befehls ein Fehler auftritt.

udiErrorId: Enthält den Fehlercode [► 39], der bei der Ausführung des Bausteins auftreten kann. Ist kein Fehler aufgetreten, so ist der Wert 0.

Voraussetzungen

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

8.9 FB_CrestronGetClientConnectionState



Dieser Funktionsblock fragt den Verbindungsstatus zu einer Crestron Steuerung ab.

VAR_INPUT

```
sTargetIpAddr : STRING(15);
bExecute      : BOOL;
tTimeout      : TIME := T#5s;
```

sTargetIpAddr: IP-Adresse der Crestron Steuerung.

bExecute: Eine positive Flanke startet den Baustein.

tTimeout: Gibt die Zeitdauer an, in der auf eine Rückantwort gewartet wird.

VAR_OUTPUT

```
bConnectionState : BOOL;
bBusy            : BOOL;
bError           : BOOL;
udiErrorId       : UDINT;
```

bConnectionState: Der Ausgang ist TRUE, wenn eine Crestron Steuerung mit der am Eingang angebenen IP-Adresse mit dem Crestron Server verbunden ist.

bBusy: Während des Abfragevorgangs wird der Ausgang *bBusy* auf TRUE gesetzt und bleibt so lange TRUE, bis die Abfrage abgeschlossen wurde. Während dieser Zeit können über diesen Baustein keine weiteren Befehle aktiviert werden.

bError: Der Ausgang ist TRUE, sobald beim Ausführen des Befehls ein Fehler auftritt.

udiErrorId: Enthält den Fehlercode [► 39], der bei der Ausführung des Bausteins auftreten kann. Ist kein Fehler aufgetreten, so ist der Wert 0.

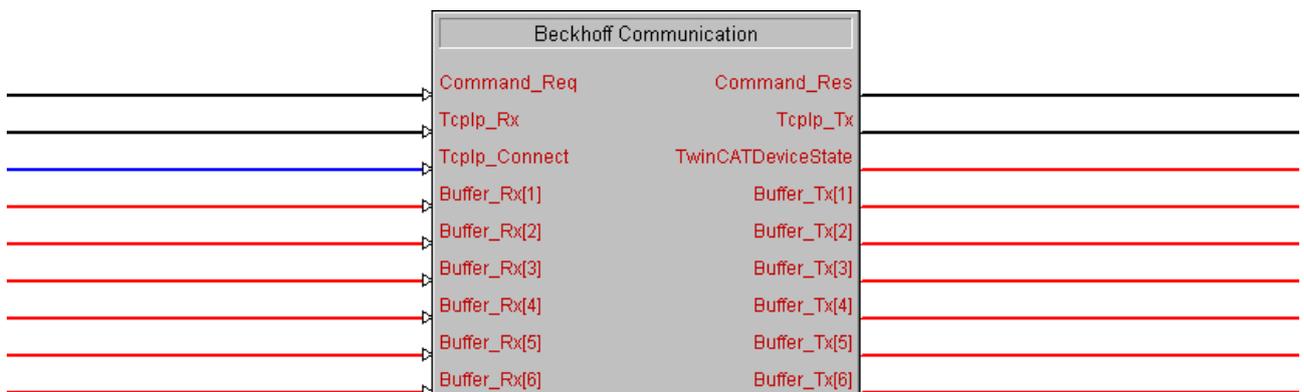
Voraussetzungen

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

9 Crestron SIMPL Windows User Modules

Name	Funktion
Beckhoff Communication [▶ 25]	User Module, welches die Daten (bis zu 100 Signale) vom TCP/IP-Client entgegen nimmt oder an diesen weiterleitet.
Beckhoff Communication 400 [▶ 26]	User Module, welches die Daten (bis zu 400 Signale) vom TCP/IP-Client entgegen nimmt oder an diesen weiterleitet.
Beckhoff Read Bit [▶ 27]	Lesen eines Wertes vom Typ Bit durch den Index Group und den Index Offset.
Beckhoff Read Bit By Name [▶ 28]	Lesen eines Wertes vom Typ Bit durch den Variablenname.
Beckhoff Read Byte [▶ 30]	Lesen eines Wertes vom Typ Byte durch den Index Group und den Index Offset.
Beckhoff Read Byte By Name [▶ 31]	Lesen eines Wertes vom Typ Byte durch den Variablenname.
Beckhoff Read Int [▶ 34]	Lesen eines Wertes vom Typ Integer durch den Index Group und den Index Offset.
Beckhoff Read Int By Name [▶ 35]	Lesen eines Wertes vom Typ Integer durch den Variablenname.
Beckhoff Switch [▶ 37]	Schreiben des Wertes '1' bei positiver Flanke und Schreiben des Wertes '0' bei negativer Flanke vom Typ Bit durch den Index Group und den Index Offset.
Beckhoff Switch By Name [▶ 38]	Schreiben des Wertes '1' bei positiver Flanke und Schreiben des Wertes '0' bei negativer Flanke vom Typ Bit durch den Variablenname.
Beckhoff Write Bit [▶ 29]	Schreiben eines Wertes vom Typ Bit durch den Index Group und den Index Offset.
Beckhoff Write Bit By Name [▶ 30]	Schreiben eines Wertes vom Typ Bit durch den Variablenname.
Beckhoff Write Byte [▶ 32]	Schreiben eines Wertes vom Typ Byte durch den Index Group und den Index Offset.
Beckhoff Write Byte By Name [▶ 33]	Schreiben eines Wertes vom Typ Byte durch den Variablenname.
Beckhoff Write Int [▶ 36]	Schreiben eines Wertes vom Typ Integer durch den Index Group und den Index Offset.
Beckhoff Write Int By Name [▶ 36]	Schreiben eines Wertes vom Typ Integer durch den Variablenname.

9.1 Beckhoff Communication



Der Baustein stellt die Basis der Kommunikation zwischen der TwinCAT SPS und der Crestron Steuerung dar. Er nimmt Lese- / Schreib- Befehle über TCP/IP entgegen und formt sie zu Crestron kompatiblen Signalen um. Ebenfalls werden Crestron Signale so umgeformt, dass der TwinCAT Crestron Server sie verarbeiten kann. Des Weiteren liest der Baustein zyklisch den AdsState des TwinCAT Gerätes aus (siehe FAQ).

Der Baustein kann bis zu 100 Signale verarbeiten.

Eingänge

```
Command_Req      : SERIAL;
TcpIp_Rx         : SERIAL;
TcpIp_Connect   : DIGITAL;
Buffer_Rx[1..100] : ANALOG;
```

Command_Req: Signal von den einzelnen Schreib- und Lesebausteinen. In diesem Signal ist das ADS-Kommando enthalten, welches an den TCP/IP-Client weitergeleitet wird

Tcplp_Rx: Dieses Signal wird mit dem Rx-Ausgang des TCP/IP-Clients verbunden.

Tcplp_Connect: Solange dieser Eingang 'High' ist, baut der Baustein die Verbindung auf.

Buffer_Rx[1..100]: Daten, auf die lesend von Außen zugegriffen werden kann.

Ausgänge

```
Command_Res     : SERIAL;
TcpIp_Tx        : SERIAL;
TwinCATDeviceState : ANALOG;
Buffer_Tx[1..100] : ANALOG;
```

Command_Res: Signal mit den Rückmeldungen (Feedback), die an die einzelnen Schreib- und Lesebausteine weitergegeben werden.

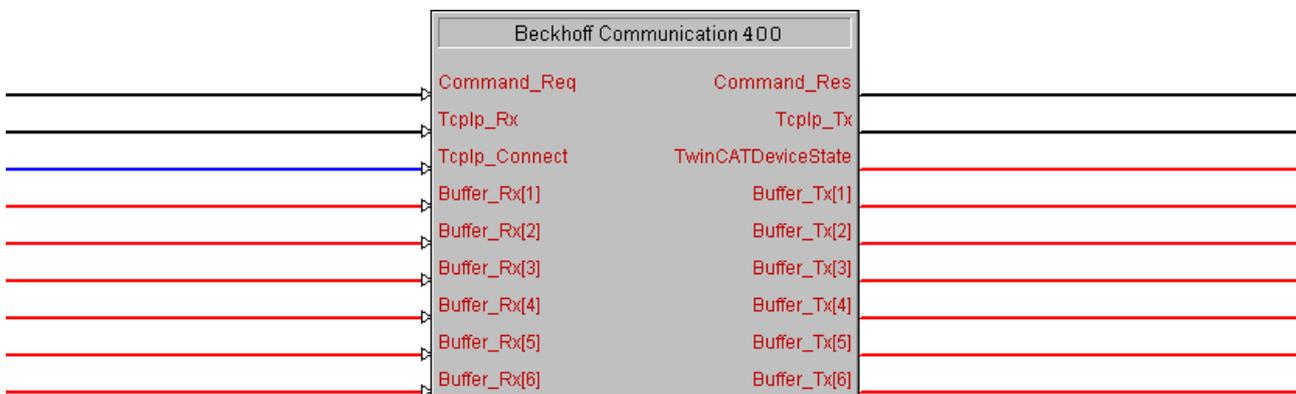
Tcplp_Tx: Dieses Signal wird mit dem Tx-Eingang des TCP/IP-Clients verbunden.

TwinCATDeviceState: Dieses analoge Signal enthält den AdsState der TwinCAT SPS.

Buffer_Tx[1..100]: Daten, auf die schreibend von außen zugegriffen werden kann.

Entwicklungsumgebung	Zielplattform
MP2E V4.001.1012 (Feb 17 2009) Build 7	2-Series mit Netzwerkschnittstelle

9.2 Beckhoff Communication 400



Der Baustein stellt die Basis der Kommunikation zwischen der TwinCAT SPS und der Crestron Steuerung dar. Er nimmt Lese- / Schreib- Befehle über TCP/IP entgegen und formt sie zu Crestron kompatiblen Signalen um. Ebenfalls werden Crestron Signale so umgeformt, dass der TwinCAT Crestron Server sie verarbeiten kann. Des Weiteren liest der Baustein zyklisch den AdsState des TwinCAT Gerätes aus (siehe FAQ).

Der Baustein kann bis zu 400 Signale verarbeiten.

Eingänge

```
Command_Req      : SERIAL;
TcpIp_Rx         : SERIAL;
TcpIp_Connect   : DIGITAL;
Buffer_Rx[1..400] : ANALOG;
```

Command_Req: Signal von den einzelnen Schreib- und Lesebausteinen. In diesem Signal ist das ADS-Kommando enthalten, welches an den TCP/IP-Client weitergeleitet wird

Tcplp_Rx: Dieses Signal wird mit dem Rx-Ausgang des TCP/IP-Clients verbunden.

Tcplp_Connect: Solange dieser Eingang 'High' ist, baut der Baustein die Verbindung auf.

Buffer_Rx[1..400]: Daten, auf die lesend von Außen zugegriffen werden kann.

Ausgänge

```
Command_Res      : SERIAL;
TcpIp_Tx         : SERIAL;
TwinCATDeviceState : ANALOG;
Buffer_Tx[1..400] : ANALOG;
```

Command_Res: Signal mit den Rückmeldungen (Feedback), die an die einzelnen Schreib- und Lesebausteine weitergegeben werden.

Tcplp_Tx: Dieses Signal wird mit dem Tx-Eingang des TCP/IP-Clients verbunden.

TwinCATDeviceState: Dieses analoge Signal enthält den AdsState der TwinCAT SPS.

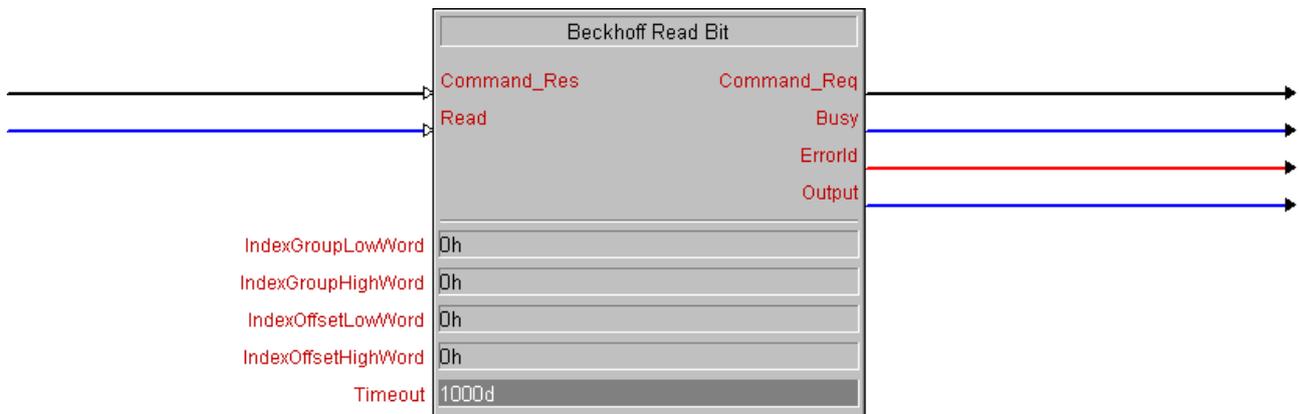
Buffer_Tx[1..400]: Daten, auf die schreibend von außen zugegriffen werden kann.



Der Baustein steht ab der Version 3.1.0 vom TwinCAT Crestron Server zur Verfügung.

Entwicklungsumgebung	Zielplattform
MP2E V4.001.1012 (Feb 17 2009) Build 7	2-Series mit Netzwerkschnittstelle

9.3 Beckhoff Read Bit



Dieser Baustein liest eine 1-Bit-Variable aus einer TwinCAT-SPS aus und stellt diese als digitales Signal an dem Ausgang *Output* zur Verfügung.

Eingänge

```
Command_Res : SERIAL;
Read        : DIGITAL;
```

Command_Res: Dieses Signal wird mit dem *Command_Res*-Ausgang des Bausteins Beckhoff Communication [▶ 25] oder Beckhoff Communication 400 [▶ 26] verbunden. Hierüber werden die Rückmeldungen übertragen.

Read: Über eine positive Flanke an diesem Eingang wird der Lesebefehl aktiviert.

Ausgänge

```
Command_Res : SERIAL;
Busy        : DIGITAL;
ErrorId     : ANALOG;
Output      : DIGITAL;
```

Command_Res: Dieses Signal wird mit dem *Command_Res*-Ausgang des Bausteins Beckhoff Communication [▶ 25] oder Beckhoff Communication 400 [▶ 26] verbunden. Hierüber werden die Rückmeldungen übertragen.

Busy: Während der Übertragung wird der Ausgang gesetzt und bleibt so lange aktiv, bis das Signal gelesen wurde. Während dieser Zeit können über diesen Baustein keine weiteren Lesebefehle aktiviert werden.

ErrorId: Dieses analoge Signal enthält den Fehlercode, der bei der Ausführung des Befehls auftreten kann. Ist kein Fehler aufgetreten, so hat dieses Signal den Wert 0.

Output: Der ausgelesene Wert wird über dieses digitale Signal ausgegeben.

Parameter

```
IndexGroupLowWord : ANALOG;
IndexGroupHighWord : ANALOG;
IndexOffsetLowWord : ANALOG;
IndexOffsetHighWord : ANALOG;
Timeout : ANALOG;
```

IndexGroupLowWord: Das niederwertige Wort vom Index Group.

IndexGroupHighWord: Das höherwertige Wort vom Index Group.

IndexOffsetLowWord: Das niederwertige Wort vom Index Offset.

IndexOffsetHighWord: Das höherwertige Wort vom Index Offset.

Timeout: Gibt die Zeitdauer an, in der auf eine Rückantwort gewartet wird. Die Einheit ist ms.

Entwicklungsumgebung	Zielplattform
MP2E V4.001.1012 (Feb 17 2009) Build 7	2-Series mit Netzwerkschnittstelle

9.4 Beckhoff Read Bit By Name



Dieser Baustein liest eine 1-Bit-Variable aus einer TwinCAT-SPS aus und stellt diese als digitales Signal an dem Ausgang *Output* zur Verfügung.

Eingänge

```
Command_Res : SERIAL;
Read : DIGITAL;
```

Command_Res: Dieses Signal wird mit dem *Command_Res*-Ausgang des Bausteins Beckhoff Communication [▶ 25] oder Beckhoff Communication 400 [▶ 26] verbunden. Hierüber werden die Rückmeldungen übertragen.

Read: Über eine positive Flanke an diesem Eingang wird der Lesebefehl aktiviert.

Ausgänge

```
Command_Res : SERIAL;
Busy : DIGITAL;
ErrorId : ANALOG;
Output : DIGITAL;
```

Command_Req: Dieses Signal wird mit dem *Command_Req*-Eingang des Bausteins Communication [► 25] oder Beckhoff Communication 400 [► 26] verbunden. Hierüber werden die Befehle übertragen.

Busy: Während der Übertragung wird der Ausgang gesetzt und bleibt so lange aktiv, bis das Signal gelesen wurde. Während dieser Zeit können über diesen Baustein keine weiteren Lesebefehle aktiviert werden.

ErrorId: Dieses analoge Signal enthält den Fehlercode, der bei der Ausführung des Befehls auftreten kann. Ist kein Fehler aufgetreten, so hat dieses Signal den Wert 0.

Output: Der ausgelesene Wert wird über dieses digitale Signal ausgegeben.

Parameter

```
VariableName : SERIAL;
Timeout      : ANALOG;
```

VariableName: Gibt den Namen der zu lesenden Variable in der TwinCAT-SPS an.

Timeout: Gibt die Zeitdauer an, in der auf eine Rückantwort gewartet wird. Die Einheit ist ms.

Entwicklungsumgebung	Zielplattform
MP2E V4.001.1012 (Feb 17 2009) Build 7	2-Series mit Netzwerkschnittstelle

9.5 Beckhoff Write Bit



Dieser Baustein schreibt den Wert eines digitalen Signals, der am Eingang *Input* anliegt, in eine 1-Bit-Variable der TwinCAT-SPS.

Eingänge

```
Write : DIGITAL;
Input  : DIGITAL;
```

Write: Über eine positive Flanke an diesem Eingang wird der Schreibbefehl aktiviert.

Input: Dieser Wert wird übermittelt.

Ausgänge

```
Command_Req : SERIAL;
```

Command_Req: Dieses Signal wird mit dem *Command_Req*-Eingang des Bausteins Beckhoff Communication [► 25] oder Beckhoff Communication 400 [► 26] verbunden. Hierüber werden die Befehle übertragen.

Parameter

```
IndexGroupLowWord : ANALOG;
IndexGroupHighWord : ANALOG;
IndexOffsetLowWord : ANALOG;
IndexOffsetHighWord : ANALOG;
```

IndexGroupLowWord: Das niederwertige Wort vom Index Group.

IndexGroupHighWord: Das höherwertige Wort vom Index Group.

IndexOffsetLowWord: Das niederwertige Wort vom Index Offset.

IndexOffsetHighWord: Das höherwertige Wort vom Index Offset.

Entwicklungsumgebung	Zielplattform
MP2E V4.001.1012 (Feb 17 2009) Build 7	2-Series mit Netzwerkschnittstelle

9.6 Beckhoff Write Bit By Name



Dieser Baustein schreibt den Wert eines digitalen Signals, der am Eingang *Input* anliegt, in eine 1-Bit-Variable der TwinCAT-SPS.

Eingänge

Write : DIGITAL;
Input : DIGITAL;

Write: Über eine positive Flanke an diesem Eingang wird der Schreibbefehl aktiviert.

Input: Dieser Wert wird übermittelt.

Ausgänge

Command_Req : SERIAL;

Command_Req: Dieses Signal wird mit dem *Command_Req*-Eingang des Bausteins Beckhoff Communication [► 25] oder Beckhoff Communication 400 [► 26] verbunden. Hierüber werden die Befehle übertragen.

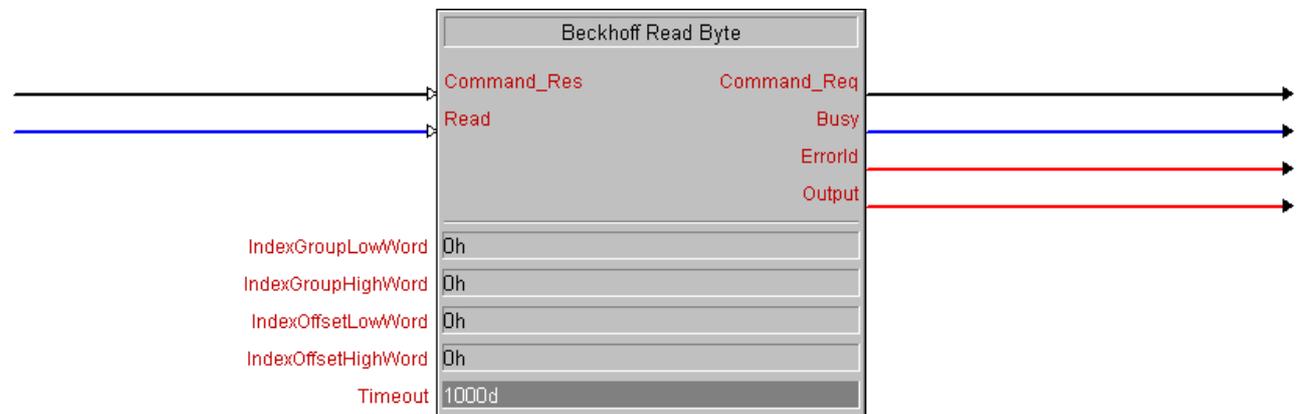
Parameter

VariableName : SERIAL;

VariableName: Gibt den Namen der zu beschreibenden Variablen in der TwinCAT-SPS an.

Entwicklungsumgebung	Zielplattform
MP2E V4.001.1012 (Feb 17 2009) Build 7	2-Series mit Netzwerkschnittstelle

9.7 Beckhoff Read Byte



Dieser Baustein liest eine 8-Bit-Variable aus einer TwinCAT-SPS aus und stellt diese als analoges Signal an dem Ausgang *Output* zur Verfügung.

Eingänge

```
Command_Res : SERIAL;
Read        : DIGITAL;
```

Command_Res: Dieses Signal wird mit dem *Command_Res*-Ausgang des Bausteins Beckhoff Communication [▶ 25] oder Beckhoff Communication 400 [▶ 26] verbunden. Hierüber werden die Rückmeldungen übertragen.

Read: Über eine positive Flanke an diesem Eingang wird der Lesebefehl aktiviert.

Ausgänge

```
Command_Req : SERIAL;
Busy        : DIGITAL;
ErrorId     : ANALOG;
Output      : ANALOG;
```

Command_Req: Dieses Signal wird mit den *Command_Req*-Eingang des Bausteins Beckhoff Communication [▶ 25] oder Beckhoff Communication 400 [▶ 26] verbunden. Hierüber werden die Befehle übertragen.

Busy: Während der Übertragung wird der Ausgang gesetzt und bleibt so lange aktiv, bis das Signal gelesen wurde. Während dieser Zeit können über diesen Baustein keine weiteren Lesebefehle aktiviert werden.

ErrorId: Dieses analoge Signal enthält den Fehlercode der bei der Ausführung des Befehls auftreten kann. Ist kein Fehler aufgetreten, so hat dieses Signal den Wert 0.

Output: Der ausgelesene Wert wird über dieses analoge Signal ausgegeben.

Parameter

```
IndexGroupLowWord  : ANALOG;
IndexGroupHighWord : ANALOG;
IndexOffsetLowWord : ANALOG;
IndexOffsetHighWord : ANALOG;
Timeout            : ANALOG;
```

IndexGroupLowWord: Das niederwertige Wort vom Index Group.

IndexGroupHighWord: Das höherwertige Wort vom Index Group.

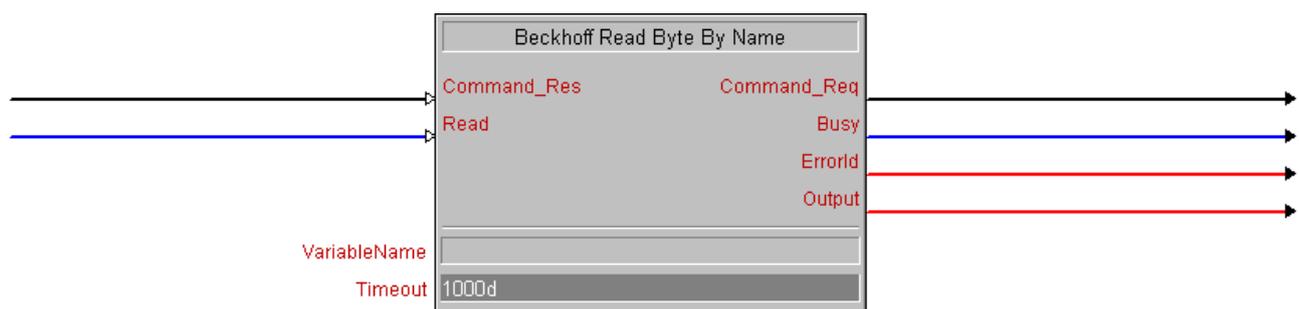
IndexOffsetLowWord: Das niederwertige Wort vom Index Offset.

IndexOffsetHighWord: Das höherwertige Wort vom Index Offset.

Timeout: Gibt die Zeitdauer an, in der auf eine Rückantwort gewartet wird. Die Einheit ist ms.

Entwicklungsumgebung	Zielplattform
MP2E V4.001.1012 (Feb 17 2009) Build 7	2-Series mit Netzwerkschnittstelle

9.8 Beckhoff Read Byte By Name



Dieser Baustein liest eine 8-Bit-Variable aus einer TwinCAT-SPS aus und stellt diese als analoges Signal an dem Ausgang *Output* zur Verfügung.

Eingänge

```
Command_Res : SERIAL;
Read        : DIGITAL;
```

Command_Res: Dieses Signal wird mit dem *Command_Res*-Ausgang des Bausteins Beckhoff Communication [▶ 25] oder Beckhoff Communication 400 [▶ 26] verbunden. Hierüber werden die Rückmeldungen übertragen.

Read: Über eine positive Flanke an diesem Eingang wird der Lesebefehl aktiviert.

Ausgänge

```
Command_Req : SERIAL;
Busy        : DIGITAL;
ErrorId     : ANALOG;
Output      : ANALOG;
```

Command_Req: Dieses Signal wird mit dem *Command_Req*-Eingang des Bausteins Beckhoff Communication [▶ 25] oder Beckhoff Communication 400 [▶ 26] verbunden. Hierüber werden die Befehle übertragen.

Busy: Während der Übertragung wird der Ausgang gesetzt und bleibt so lange aktiv, bis das Signal gelesen wurde. Während dieser Zeit können über diesen Baustein keine weiteren Lesebefehle aktiviert werden.

ErrorId: Dieses analoge Signal enthält den Fehlercode, der bei der Ausführung des Befehls auftreten kann. Ist kein Fehler aufgetreten, so hat dieses Signal den Wert 0.

Output: Der ausgelesene Wert wird über dieses analoge Signal ausgegeben.

Parameter

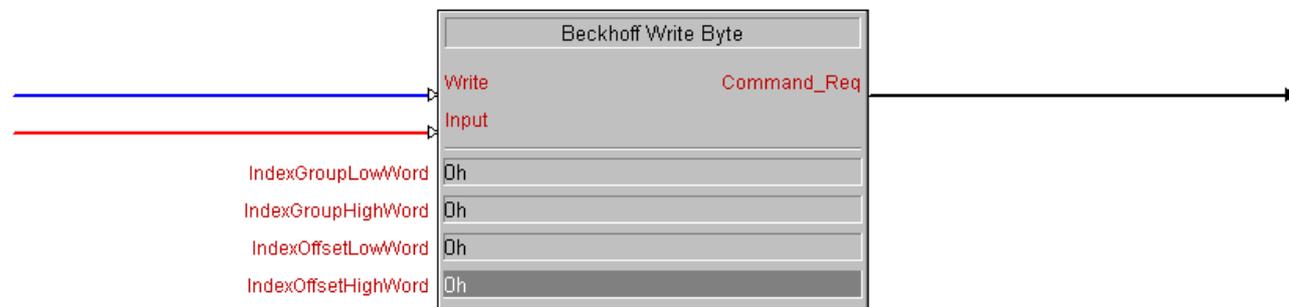
```
VariableName : SERIAL;
Timeout      : ANALOG;
```

VariableName: Gibt den Namen der zu lesenden Variable in der TwinCAT-SPS an.

Timeout: Gibt die Zeitdauer an, in der auf eine Rückantwort gewartet wird. Die Einheit ist ms.

Entwicklungsumgebung	Zielplattform
MP2E V4.001.1012 (Feb 17 2009) Build 7	2-Series mit Netzwerkschnittstelle

9.9 Beckhoff Write Byte



Dieser Baustein schreibt den Wert eines analogen Signals, der am Eingang *Input* anliegt, in eine 8-Bit-Variable der TwinCAT-SPS.

Eingänge

```
Write : DIGITAL;
Input : ANALOG;
```

Write: Über eine positive Flanke an diesem Eingang wird der Schreibbefehl aktiviert.

Input: Dieser Wert wird übermittelt.

Ausgänge

Command_Req : SERIAL;

Command_Req: Dieses Signal wird mit dem *Command_Req*-Eingang des Bausteins Beckhoff Communication [▶ 25] oder Beckhoff Communication 400 [▶ 26] verbunden. Hierüber werden die Befehle übertragen.

Parameter

IndexGroupLowWord : ANALOG;
 IndexGroupHighWord : ANALOG;
 IndexOffsetLowWord : ANALOG;
 IndexOffsetHighWord : ANALOG;

IndexGroupLowWord: Das niederwertige Wort vom Index Group.

IndexGroupHighWord: Das höherwertige Wort vom Index Group.

IndexOffsetLowWord: Das niederwertige Wort vom Index Offset.

IndexOffsetHighWord: Das höherwertige Wort vom Index Offset.

Entwicklungsumgebung	Zielplattform
MP2E V4.001.1012 (Feb 17 2009) Build 7	2-Series mit Netzwerkschnittstelle

9.10 Beckhoff Write Byte By Name



Dieser Baustein schreibt den Wert eines analogen Signals, der am Eingang *Input* anliegt, in eine 8-Bit-Variable der TwinCAT-SPS.

Eingänge

Write : DIGITAL;
 Input : ANALOG;

Write: Über eine positive Flanke an diesem Eingang wird der Schreibbefehl aktiviert.

Input: Dieser Wert wird übermittelt.

Ausgänge

Command_Req : SERIAL;

Command_Req: Dieses Signal wird mit dem *Command_Req*-Eingang des Bausteins Beckhoff Communication [▶ 25] oder Beckhoff Communication 400 [▶ 26] verbunden. Hierüber werden die Befehle übertragen.

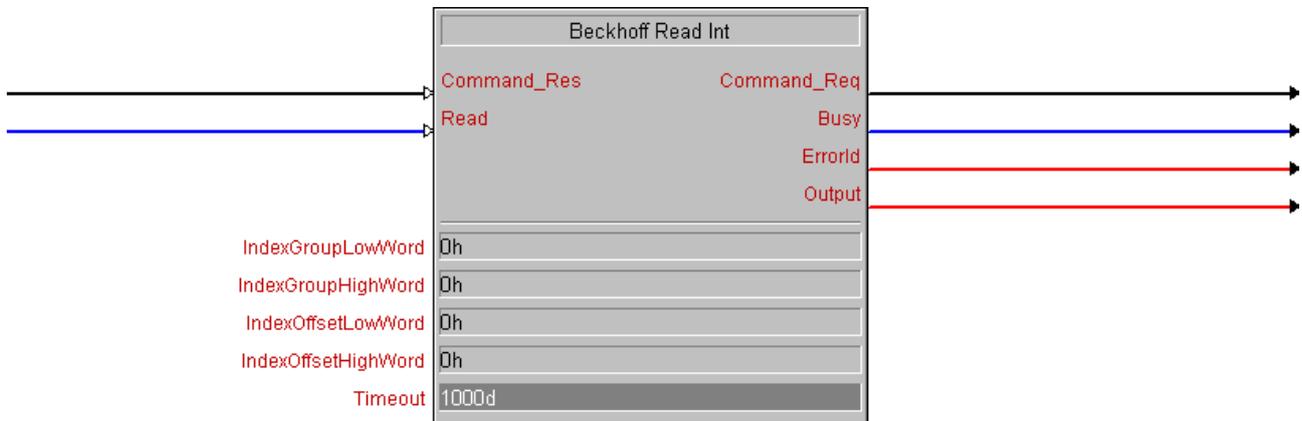
Parameter

VariableName : SERIAL;

VariableName: Gibt den Namen der zu beschreibenden Variablen in der TwinCAT-SPS an.

Entwicklungsumgebung	Zielformat
MP2E V4.001.1012 (Feb 17 2009) Build 7	2-Series mit Netzwerkschnittstelle

9.11 Beckhoff Read Int



Dieser Baustein liest eine 16-Bit-Variable aus einer TwinCAT-SPS aus und stellt diese als analoges Signal an dem Ausgang *Output* zur Verfügung.

Eingänge

```
Command_Res : SERIAL;
Read        : DIGITAL;
```

Command_Res: Dieses Signal wird mit dem *Command_Res*-Ausgang des Bausteins Beckhoff Communication [► 25] oder Beckhoff Communication 400 [► 26] verbunden. Hierüber werden die Rückmeldungen übertragen.

Read: Über eine positive Flanke an diesem Eingang wird der Lesebefehl aktiviert.

Ausgänge

```
Command_Req : SERIAL;
Busy        : DIGITAL;
ErrorId     : ANALOG;
Output      : ANALOG;
```

Command_Req: Dieses Signal wird mit dem *Command_Req*-Eingang des Bausteins Beckhoff Communication [► 25] oder Beckhoff Communication 400 [► 26] verbunden. Hierüber werden die Befehle übertragen.

Busy: Während der Übertragung wird der Ausgang gesetzt und bleibt so lange aktiv, bis das Signal gelesen wurde. Während dieser Zeit können über diesen Baustein keine weiteren Lesebefehle aktiviert werden.

ErrorId: Dieses analoge Signal enthält den Fehlercode, der bei der Ausführung des Befehls auftreten kann. Ist kein Fehler aufgetreten, so hat dieses Signal den Wert 0.

Output: Der ausgelesene Wert wird über dieses analoge Signal ausgegeben.

Parameter

```
IndexGroupLowWord : ANALOG;
IndexGroupHighWord : ANALOG;
IndexOffsetLowWord : ANALOG;
IndexOffsetHighWord : ANALOG;
Timeout           : ANALOG;
```

IndexGroupLowWord: Das niederwertige Wort vom Index Group.

IndexGroupHighWord: Das höherwertige Wort vom Index Group.

IndexOffsetLowWord: Das niederwertige Wort vom Index Offset.

IndexOffsetHighWord: Das höherwertige Wort vom Index Offset.

Timeout: Gibt die Zeitdauer an, in der auf eine Rückantwort gewartet wird. Die Einheit ist ms.

Entwicklungsumgebung	Zielplattform
MP2E V4.001.1012 (Feb 17 2009) Build 7	2-Series mit Netzwerkschnittstelle

9.12 Beckhoff Read Int By Name



Dieser Baustein liest eine 16-Bit-Variable aus einer TwinCAT-SPS aus und stellt diese als analoges Signal an dem Ausgang *Output* zur Verfügung.

Eingänge

```
Command_Res : SERIAL;
Read        : DIGITAL;
```

Command_Res: Dieses Signal wird mit dem *Command_Res*-Ausgang des Bausteins Beckhoff Communication [▶ 25] oder Beckhoff Communication 400 [▶ 26] verbunden. Hierüber werden die Rückmeldungen übertragen.

Read: Über eine positive Flanke an diesem Eingang wird der Lesebefehl aktiviert.

Ausgänge

```
Command_Req : SERIAL;
Busy        : DIGITAL;
ErrorId     : ANALOG;
Output      : ANALOG;
```

Command_Req: Dieses Signal wird mit dem *Command_Req*-Eingang des Bausteins Beckhoff Communication [▶ 25] oder Beckhoff Communication 400 [▶ 26] verbunden. Hierüber werden die Befehle übertragen.

Busy: Während der Übertragung wird der Ausgang gesetzt und bleibt so lange aktiv, bis das Signal gelesen wurde. Während dieser Zeit können über diesen Baustein keine weiteren Lesebefehle aktiviert werden.

ErrorId: Dieses analoge Signal enthält den Fehlercode, der bei der Ausführung des Befehls auftreten kann. Ist kein Fehler aufgetreten, so hat dieses Signal den Wert 0.

Output: Der ausgelesene Wert wird über dieses analoge Signal ausgegeben.

Parameter

```
VariableName : SERIAL;
Timeout      : ANALOG;
```

VariableName: Gibt den Namen der zu lesenden Variable in der TwinCAT-SPS an.

Timeout: Gibt die Zeitdauer an, in der auf eine Rückantwort gewartet wird. Die Einheit ist ms.

Entwicklungsumgebung	Zielplattform
MP2E V4.001.1012 (Feb 17 2009) Build 7	2-Series mit Netzwerkschnittstelle

9.13 Beckhoff Write Int



Dieser Baustein schreibt den Wert eines analogen Signals, der am Eingang *Input* anliegt, in eine 16-Bit-Variable der TwinCAT-SPS.

Eingänge

Write : DIGITAL;
Input : ANALOG;

Write: Über eine positive Flanke an diesem Eingang wird der Schreibbefehl aktiviert.

Input: Dieser Wert wird übermittelt.

Ausgänge

Command_Req : SERIAL;

Command_Req: Dieses Signal wird mit dem *Command_Req*-Eingang des Bausteins Beckhoff Communication [► 25] oder Beckhoff Communication 400 [► 26] verbunden. Hierüber werden die Befehle übertragen.

Parameter

IndexGroupLowWord : ANALOG;
IndexGroupHighWord : ANALOG;
IndexOffsetLowWord : ANALOG;
IndexOffsetHighWord : ANALOG;

IndexGroupLowWord: Das niederwertige Wort vom Index Group.

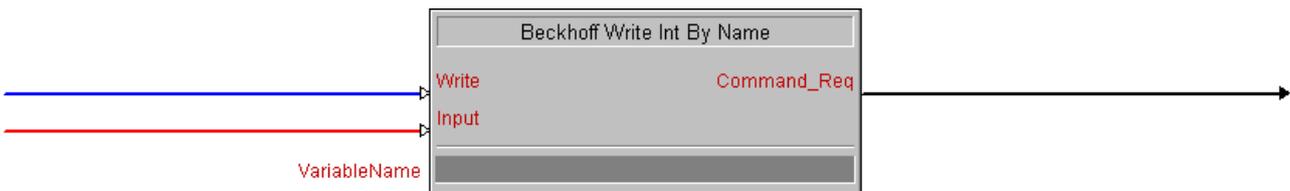
IndexGroupHighWord: Das höherwertige Wort vom Index Group.

IndexOffsetLowWord: Das niederwertige Wort vom Index Offset.

IndexOffsetHighWord: Das höherwertige Wort vom Index Offset.

Entwicklungsumgebung	Zielplattform
MP2E V4.001.1012 (Feb 17 2009) Build 7	2-Series mit Netzwerkschnittstelle

9.14 Beckhoff Write Int By Name



Dieser Baustein schreibt den Wert eines analogen Signals, der am Eingang *Input* anliegt, in eine 16-Bit-Variable der TwinCAT-SPS.

Eingänge

```
Write : DIGITAL;
Input : ANALOG;
```

Write: Über eine positive Flanke an diesem Eingang wird der Schreibbefehl aktiviert.

Input: Dieser Wert wird übermittelt.

Ausgänge

```
Command_Req : SERIAL;
```

Command_Req: Dieses Signal wird mit dem *Command_Req*-Eingang des Bausteins Beckhoff Communication [▶ 25] oder Beckhoff Communication 400 [▶ 26] verbunden. Hierüber werden die Befehle übertragen.

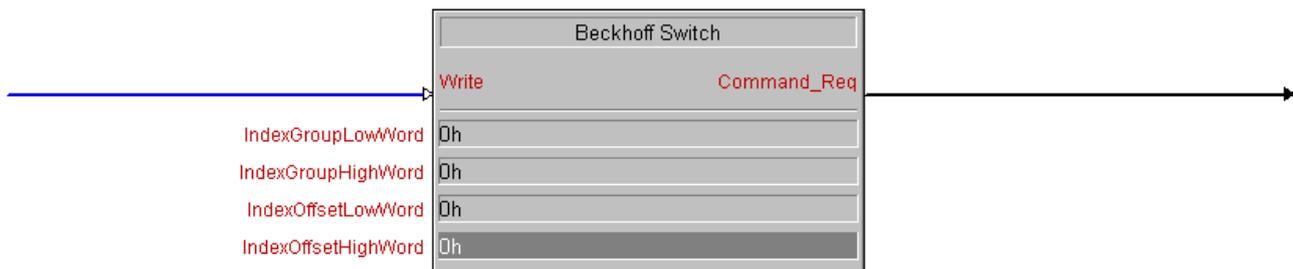
Parameter

```
VariableName : SERIAL;
```

VariableName: Gibt den Namen der zu beschreibenden Variablen in der TwinCAT-SPS an.

Entwicklungsumgebung	Zielplattform
MP2E V4.001.1012 (Feb 17 2009) Build 7	2-Series mit Netzwerkschnittstelle

9.15 Beckhoff Switch



Dieser Baustein schreibt den Wert '1' bei einer positiven Flanke am Eingang *Write* und den Wert '0' bei einer negativen Flanke am Eingang *Write*, in eine 1-Bit-Variablen der TwinCAT-SPS.

Eingänge

```
Write : DIGITAL;
```

Write: Über eine positive Flanke an diesem Eingang wird der Wert '1' und über eine negative Flanke der Wert '0' geschrieben.

Ausgänge

```
Command_Req : SERIAL;
```

Command_Req: Dieses Signal wird mit dem *Command_Req*-Eingang des Bausteins Beckhoff Communication [▶ 25] oder Beckhoff Communication 400 [▶ 26] verbunden. Hierüber werden die Befehle übertragen.

Parameter

```
IndexGroupLowWord : ANALOG;
IndexGroupHighWord : ANALOG;
IndexOffsetLowWord : ANALOG;
IndexOffsetHighWord : ANALOG;
```

IndexGroupLowWord: Das niederwertige Wort vom Index Group.

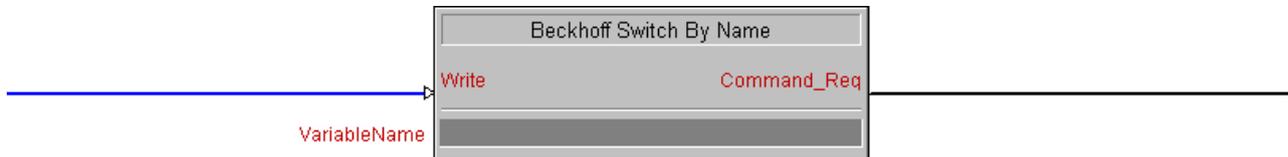
IndexGroupHighWord: Das höherwertige Wort vom Index Group.

IndexOffsetLowWord: Das niederwertige Wort vom Index Offset.

IndexOffsetHighWord: Das höherwertige Wort vom Index Offset.

Entwicklungsumgebung	Zielplattform
MP2E V4.001.1012 (Feb 17 2009) Build 7	2-Series mit Netzwerkschnittstelle

9.16 Beckhoff Switch By Name



Dieser Baustein schreibt den Wert '1' bei einer positiven Flanke am Eingang *Write* und den Wert '0' bei einer negativen Flanke am Eingang *Write*, in eine 1-Bit-Variable der TwinCAT-SPS.

Eingänge

`Write : DIGITAL;`

Write: Über eine positive Flanke an diesem Eingang wird der Wert '1', und über eine negative Flanke der Wert '0' geschrieben.

Ausgänge

`Command_Req : SERIAL;`

Command_Req: Dieses Signal wird mit dem *Command_Req*-Eingang des Bausteins Beckhoff Communication [▶ 25] oder Beckhoff Communication 400 [▶ 26] verbunden. Hierüber werden die Befehle übertragen.

Parameter

`VariableName : SERIAL;`

VariableName: Gibt den Namen der zu beschreibenden Variable in der TwinCAT-SPS an.

Entwicklungsumgebung	Zielplattform
MP2E V4.001.1012 (Feb 17 2009) Build 7	2-Series mit Netzwerkschnittstelle

10 Fehlercodes

Hex Code	Dec Code	Beschreibung
0x8501	34049	Eingabeparameter Fehler an einem SPS Funktionsbaustein.
0x8502	34050	Die Konfigurationsdatei TcCrestronServer.cfg wurde nicht gefunden
0x8503	34051	Interner Server Fehler beim Verarbeiten der Konfigurationsdatei TcCrestronServer.cfg .
0x8504	34052	Fehler in der Konfigurationsdatei TcCrestronServer.cfg . Der XML-Knoten "TcpPort" wurde nicht gefunden.
0x8505	34053	Fehler in der Konfigurationsdatei TcCrestronServer.cfg . Der XML-Knoten "AmsPort" wurde nicht gefunden.
0x8506	34054	Der Client wurde nicht gefunden.
0x8507	34055	Fehler beim Verarbeiten einer Leseanforderung.
0x8508	34056	Fehler beim Verarbeiten einer Schreibanforderung.
0x8509	34057	Fehler beim Verarbeiten einer Array Leseanforderung.
0x850A	34058	Fehler beim Verarbeiten einer Array Schreibanforderung.
0x850B	34059	Fehler beim Auslesen von einem Client Status. Der angegebene Client wurde nicht gefunden.
0x850C	34060	Zeitüberschreitung beim Lesen von einem SPS-Wert.
0x850D	34061	SPS Funktionsbaustein Eingabeparameter sTargetIpAddr ist fehlerhaft.
0x850E	34062	SPS Funktionsbaustein Eingabeparameter dwData ist fehlerhaft.
0x850F	34063	SPS Funktionsbaustein Eingabeparameter iOffset ist fehlerhaft.
0x8510	34064	SPS Funktionsbaustein Eingabeparameter iLen ist fehlerhaft.
0x8511	34065	SPS Funktionsbaustein Eingabeparameter iLen ist fehlerhaft. iLen hat den Maximalwert von 40 überschritten.
0x8512	34066	SPS Funktionsbaustein Eingabeparameter Fehler. iOffset plus iLen ist größer als die maximal zulässigen Feldergröße von 100. Der Funktionsbaustein schreibt über die zulässigen Felder hinaus, Werte gehen verloren.
0x8F07	36615	Fehler in der Konfigurationsdatei TcCrestronServer.cfg . Der XML-Knoten "AdsMessageCycleTime" wurde nicht gefunden.

11 Beispiele

11.1 Beckhoff SPS-Programmbeispiele

Voraussetzungen

Folgende Komponenten werden benötigt:

- PC mit TwinCAT ab V2.10 Build 1330 oder neuer.
- CX Gerät mit WinCE + **.NET Compact Framework** oder CX Gerät / IPC mit CE oder Windows XP Embedded + **.NET Framework**.
- Beckhoff Supplement: TwinCAT Crestron Server.
- SIMPL Window V2.08.44 mit Crestron Data Base V18.09.02.001 oder neuer.

Beschreibung	Download
<p>Write Int: In diesem Beispiel wird demonstriert, wie ein Integer (16Bit) zu einer Crestron Steuerung geschickt wird und ein Eingangssignal beschreibt.</p>	<p>https://info-sys.beckhoff.com/content/1031/tc-sample_crestron/Resources/11364626955.zip</p> <p>https://info-sys.beckhoff.com/content/1031/tc-sample_crestron/Resources/11364626955.zip</p>
<p>Write Int Array: In diesem Beispiel wird demonstriert, wie ein Integer (16Bit) Array von zehn Feldern zu einer Crestron Steuerung geschickt wird und dort zehn Eingangssignale beschreibt.</p>	<p>https://info-sys.beckhoff.com/content/1031/tc-sample_crestron/Resources/11364626955.zip</p> <p>https://info-sys.beckhoff.com/content/1031/tc-sample_crestron/Resources/11364628363.zip</p>
<p>Read Int: In diesem Beispiel wird demonstriert, wie ein Integer (16 Bit) Wert aus einem Ausgangssignal einer Crestron Steuerung ausgelesen wird.</p>	<p>https://info-sys.beckhoff.com/content/1031/tc-sample_crestron/Resources/11364626955.zip</p> <p>https://info-sys.beckhoff.com/content/1031/tc-sample_crestron/Resources/11364629771.zip</p>
<p>Get Client Connection State: In diesem Beispiel wird demonstriert, wie die Information, ob sich eine Crestron Steuerung mit einer bestimmten IP-Adresse mit dem TwinCAT Crestron Server verbunden hat, ausgelesen wird.</p>	<p>https://info-sys.beckhoff.com/content/1031/tc-sample_crestron/Resources/11364626955.zip</p>

Beschreibung	Download
	https://infosys.beckhoff.com/content/1031/tc-sample_crestron/Resources/11364631179.zip

11.2 Crestron Programmierbeispiele

Voraussetzungen

Folgende Komponenten werden benötigt:

- PC mit TwinCAT ab V2.10 Build 1330 oder höher.
- CX Gerät mit WinCE + **.NET Compact Framework** oder CX Gerät / IPC mit CE oder Windows XP Embedded + **.NET Framework**.
- Beckhoff Supplement: TwinCAT Crestron Server.
- SIMPL Window V2.08.44 mit Crestron Data Base V18.09.02.001 oder höher.



Die folgenden Crestron Beispiele wurden auf einer Crestron MP2E V4.001.1012 (Feb 17 2009) Build 7 erstellt.

Beschreibung	Download
<p>Write Int: In diesem Beispiel wird demonstriert, wie mit dem Beckhoff Modul "<i>Write Int By Name</i> und <i>Write Int By Index Group / Index Offset</i>" für Crestron ein Integer (16 Bit) Wert in der Beckhoff SPS beschrieben wird.</p>	https://infosys.beckhoff.com/content/1031/tcsample_crestron/Resources/11364632587.zip https://infosys.beckhoff.com/content/1031/tcsample_crestron/Resources/11364632587.zip
<p>Read Int: In diesem Beispiel wird demonstriert, wie mit dem Beckhoff Modul "<i>Read Int By Name</i> und <i>Read Int By Index Group / Index Offset</i>" für Crestron ein Integer (16 Bit) Wert aus der Beckhoff SPS ausgelesen wird.</p>	https://infosys.beckhoff.com/content/1031/tcsample_crestron/Resources/11364632587.zip https://infosys.beckhoff.com/content/1031/tcsample_crestron/Resources/11364633995.zip
<p>Switch Module: In diesem Beispiel wird demonstriert, wie das Beckhoff Modul "<i>Switch By Name</i> und <i>Switch By Index Group / Index Offset</i>" schreibend auf eine Toggle Variable in der Beckhoff SPS zugreift.</p>	https://infosys.beckhoff.com/content/1031/tcsample_crestron/Resources/11364632587.zip https://infosys.beckhoff.com/content/1031/tcsample_crestron/Resources/11364635403.zip

Mehr Informationen:
www.beckhoff.de/ts8036

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

