

Dokumentation | DE

EL6752

Master/Slave-Klemme für DeviceNet[®]



Inhaltsverzeichnis

1 Vorwort	5
1.1 Hinweise zur Dokumentation	5
1.2 Wegweiser durch die Dokumentation	6
1.3 Sicherheitshinweise	7
1.4 Ausgabestände der Dokumentation	8
1.5 Versionsidentifikation von EtherCAT-Geräten	9
1.5.1 Allgemeine Hinweise zur Kennzeichnung	9
1.5.2 Versionsidentifikation von EL-Klemmen	10
1.5.3 Beckhoff Identification Code (BIC)	11
1.5.4 Elektronischer Zugriff auf den BIC (eBIC)	13
2 Produktübersicht	15
2.1 Einführung	15
2.2 Technische Daten	16
3 DeviceNet® Grundlagen	17
4 Montage und Verkabelung	18
4.1 Hinweise zum ESD-Schutz	18
4.2 Explosionsschutz	19
4.2.1 ATEX - Besondere Bedingungen (erweiterter Temperaturbereich)	19
4.2.2 IECEx - Besondere Bedingungen	20
4.2.3 Weiterführende Dokumentation zu ATEX und IECEx	21
4.3 UL-Hinweise	22
4.4 Montage und Demontage - Zughebelentriegelung	23
4.5 Montage und Demontage - Frontentriegelung oben	25
4.6 Entsorgung	26
4.7 DeviceNet® Verkabelung	27
4.7.1 CAN / DeviceNet® Topologie	27
4.7.2 Buslänge	27
4.7.3 Stichleitungen	27
4.7.4 Sternverteiler (Multiport Tap)	28
4.7.5 CAN-Kabel	29
4.7.6 Schirmung	29
4.7.7 Kabelfarben und Pin-Belegung	30
4.8 Einbaulagen	31
4.9 Positionierung von passiven Klemmen	34
5 DeviceNet® Kommunikation	35
5.1 DeviceNet® Einführung	35
5.2 Explizite Nachrichten	37
6 Parametrierung und Inbetriebnahme	39
6.1 CoE-Interface	39
6.2 Allgemeine Hinweise zur Watchdog-Einstellung	45
6.3 EtherCAT State Machine	47
6.4 TwinCAT System Manager	49
6.5 Beckhoff DeviceNet® Buskoppler	59

6.6	Allgemeines DeviceNet® Gerät.....	64
6.6.1	Einbinden eines DeviceNet® Gerätes mit EDS-File	64
6.6.2	Einbinden eines DeviceNet® Gerätes ohne EDS-File	65
6.6.3	Parametrierung eines DeviceNet® Gerätes	69
6.7	EtherCAT Beschreibung	73
6.7.1	Einführung	73
6.7.2	Objektbeschreibung und Parametrierung	79
7	Fehlerbehandlung und Diagnose	104
7.1	EL6752 - LED Beschreibung.....	104
7.2	EL6752/-0010 Diagnose	106
7.2.1	EL6752/-0010 - WC-State	106
7.2.2	EL6752/-0010 - State	107
7.2.3	EL6752/-0010 - Error / DiagFlag	108
7.3	DeviceNet® Geräte Diagnose	109
7.3.1	DeviceNet® Slave Gerät / EL6752-0010 - MacState	109
7.3.2	DeviceNet® Slave Gerät / EL6752-0010 - DiagFlag	111
7.3.3	Beckhoff DeviceNet® Slave Gerät - CouplerState	112
7.4	EL6752/-0010 - ADS Error Codes.....	113
7.5	DeviceNet® / CAN Trouble Shooting	116
8	Anhang	119
8.1	EtherCAT AL Status Codes	119
8.2	Firmware Kompatibilität.....	120
8.3	Firmware Update EL/ES/ELM/EM/EP/EPP/ERPxxxx	121
8.3.1	Gerätebeschreibung ESI-File/XML	122
8.3.2	Erläuterungen zur Firmware.....	125
8.3.3	Update Controller-Firmware *.efw.....	126
8.3.4	FPGA-Firmware *.rbf.....	128
8.3.5	Gleichzeitiges Update mehrerer EtherCAT-Geräte.....	132
8.4	Abkürzungen	133
8.5	Support und Service	134

1 Vorwort

1.1 Hinweise zur Dokumentation

Zielgruppe

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

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

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

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

Disclaimer

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

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

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

Marken

Beckhoff®, ATRO®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, MX-System®, Safety over EtherCAT®, TC/BSD®, TwinCAT®, TwinCAT/BSD®, TwinSAFE®, XFC®, XPlanar® und XTS® 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.



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.

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

Fremdmarken

In dieser Dokumentation können Marken Dritter verwendet werden. Die zugehörigen Markenvermerke finden Sie unter: <https://www.beckhoff.com/trademarks>

1.2 Wegweiser durch die Dokumentation

HINWEIS



Weitere Bestandteile der Dokumentation

Diese Dokumentation beschreibt gerätespezifische Inhalte. Sie ist Bestandteil des modular aufgebauten Dokumentationskonzepts für Beckhoff I/O-Komponenten. Für den Einsatz und sicheren Betrieb des in dieser Dokumentation beschriebenen Gerätes / der in dieser Dokumentation beschriebenen Geräte werden zusätzliche, produktübergreifende Beschreibungen benötigt, die der folgenden Tabelle zu entnehmen sind.

Titel	Beschreibung
EtherCAT System-Dokumentation (PDF)	<ul style="list-style-type: none"> • Systemübersicht • EtherCAT-Grundlagen • Kabel-Redundanz • Hot Connect • Konfiguration von EtherCAT-Geräten
Explosionsschutz für Klemmensysteme (PDF)	Hinweise zum Einsatz der Beckhoff Klemmensysteme in explosionsgefährdeten Bereichen gemäß ATEX und IECEx
Infrastruktur für EtherCAT/Ethernet (PDF)	Technische Empfehlungen und Hinweise zur Auslegung, Ausfertigung und Prüfung
Software-Deklarationen I/O (PDF)	Open-Source-Software-Deklarationen für Beckhoff-I/O-Komponenten

Die Dokumentationen können auf der Beckhoff-Homepage (www.beckhoff.com) eingesehen und heruntergeladen werden über:

- den Bereich „Dokumentation und Downloads“ der jeweiligen Produktseite,
- den [Downloadfinder](#),
- das [Beckhoff Information System](#).

Sollten Sie Vorschläge oder Anregungen zu unserer Dokumentation haben, schicken Sie uns bitte unter Angabe von Dokumentationstitel und Versionsnummer eine E-Mail an: dokumentation@beckhoff.com

1.3 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.

Signalwörter

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

Warnungen vor Personenschäden

GEFAHR

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

WARNUNG

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

VORSICHT

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

Warnung vor Umwelt- oder Sachschäden

HINWEIS

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

Information zum Umgang mit dem Produkt



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

1.4 Ausgabestände der Dokumentation

Version	Kommentar
2.5.0	<ul style="list-style-type: none"> • Update Kapitel „Technische Daten“ • Kapitel „Objektbeschreibung und Parametrierung“ • Update Revisionsstand • Update Struktur
2.4	<ul style="list-style-type: none"> • Kapitel „Empfohlene Tragschiene“ entfernt • Update Kapitel „Technische Daten“ • Kapitel „IECEX - Besondere Bedingungen hinzugefügt“ • Update Revisionsstand
2.3	<ul style="list-style-type: none"> • Update Kapitel „Technische Daten“ • Update Kapitel „Versionsidentifikation von EtherCAT-Geräten“ • Update Struktur • Update Hinweise • Update Revisionsstand • Kapitel Entsorgung hinzugefügt
2.2	<ul style="list-style-type: none"> • Update Struktur
2.1	<ul style="list-style-type: none"> • Kapitel „Explizite Nachrichten ergänzt“ • Aktualisierung Kapitel „Technische Daten“ • Update Struktur • Update Revisionstatus
2.0	<ul style="list-style-type: none"> • Migration • Strukturupdate
1.4	<ul style="list-style-type: none"> • Hinzugefügt: Kapitel "Konfiguration": DeviceNet® Adresse und Baudrate über ADS ändern • Strukturupdate
1.3	<ul style="list-style-type: none"> • Korrekturen Kapitel "Technische Daten" • Hinzugefügt: Kapitel "Firmware Status" • Strukturupdate
1.2	<ul style="list-style-type: none"> • Korrekturen Kapitel "Montage und Verdrahtung"
1.1	<ul style="list-style-type: none"> • Korrekturen Kapitel "Montage und Verdrahtung"
1.0	<ul style="list-style-type: none"> • Korrekturen und Ergänzungen, erste Veröffentlichung
0.2	<ul style="list-style-type: none"> • Korrekturen und Ergänzungen
0.1	<ul style="list-style-type: none"> • Interne Vorabversion

1.5 Versionsidentifikation von EtherCAT-Geräten

1.5.1 Allgemeine Hinweise zur Kennzeichnung

Bezeichnung

Ein Beckhoff EtherCAT-Gerät hat eine 14-stellige technische Bezeichnung, die sich zusammen setzt aus

- Familienschlüssel
- Typ
- Version
- Revision

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

Hinweise

- Die oben genannten Elemente ergeben die **technische Bezeichnung**, im Folgenden wird das Beispiel EL3314-0000-0016 verwendet.
- Davon ist EL3314-0000 die Bestellbezeichnung, umgangssprachlich bei „-0000“ dann oft nur EL3314 genannt. „-0016“ ist die EtherCAT-Revision.
- Die **Bestellbezeichnung** setzt sich zusammen aus
 - Familienschlüssel (EL, EP, CU, ES, KL, CX, ...)
 - Typ (3314)
 - Version (-0000)
- Die **Revision** -0016 gibt den technischen Fortschritt wie z. B. Feature-Erweiterung in Bezug auf die EtherCAT Kommunikation wieder und wird von Beckhoff verwaltet.
Prinzipiell kann ein Gerät mit höherer Revision ein Gerät mit niedrigerer Revision ersetzen, wenn nicht anders - z. B. in der Dokumentation - angegeben.
Jeder Revision zugehörig und gleichbedeutend ist üblicherweise eine Beschreibung (ESI, EtherCAT Slave Information) in Form einer XML-Datei, die zum Download auf der Beckhoff Webseite bereitsteht. Die Revision wird seit Januar 2014 außen auf den IP20-Klemmen aufgebracht, siehe Abb. „EL2872 mit Revision 0022 und Seriennummer 01200815“.
- Typ, Version und Revision werden als dezimale Zahlen gelesen, auch wenn sie technisch hexadezimal gespeichert werden.

1.5.2 Versionsidentifikation von EL-Klemmen

Als Seriennummer/Date Code bezeichnet Beckhoff im IO-Bereich im Allgemeinen die 8-stellige Nummer, die auf dem Gerät aufgedruckt oder mit einem Aufkleber angebracht ist. Diese Seriennummer gibt den Bauzustand im Auslieferungszustand an und kennzeichnet somit eine ganze Produktions-Charge, unterscheidet aber nicht die Module innerhalb einer Charge.

Aufbau der Seriennummer: **KK YY FF HH**

KK - Produktionswoche (Kalenderwoche)

YY - Produktionsjahr

FF - Firmware-Stand

HH - Hardware-Stand

Beispiel mit Seriennummer 12 06 3A 02:

12 - Produktionswoche 12

06 - Produktionsjahr 2006

3A - Firmware-Stand 3A

02 - Hardware-Stand 02



Abb. 1: EL2872 mit Revision 0022 und Seriennummer 01200815

1.5.3 Beckhoff Identification Code (BIC)

Der Beckhoff Identification Code (BIC) wird vermehrt auf Beckhoff-Produkten zur eindeutigen Identitätsbestimmung des Produkts aufgebracht. Der BIC ist als Data Matrix Code (DMC, Code-Schema ECC200) dargestellt, der Inhalt orientiert sich am ANSI-Standard MH10.8.2-2016.

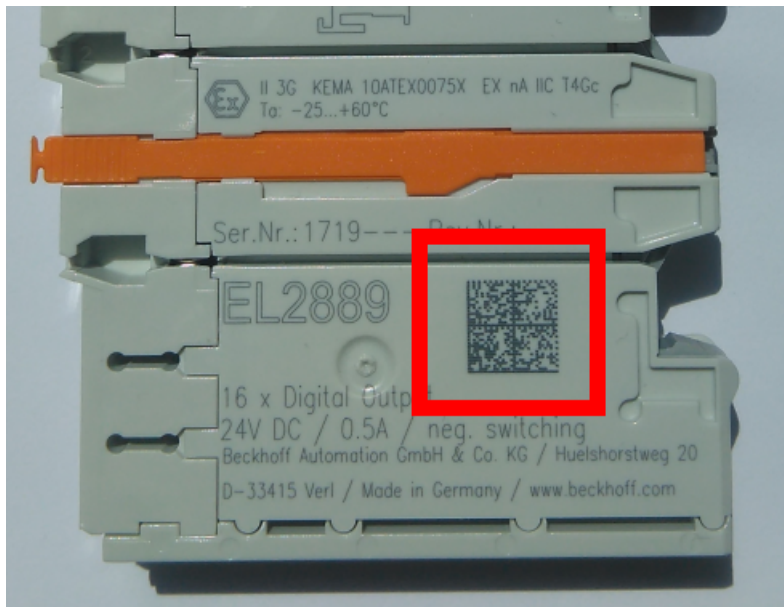


Abb. 2: BIC als Data Matrix Code (DMC, Code-Schema ECC200)

Die Einführung des BIC erfolgt schrittweise über alle Produktgruppen hinweg. Er ist je nach Produkt an folgenden Stellen zu finden:

- auf der Verpackungseinheit
- direkt auf dem Produkt (bei ausreichendem Platz)
- auf Verpackungseinheit und Produkt

Der BIC ist maschinenlesbar und enthält Informationen, die auch kundenseitig für Handling und Produktverwaltung genutzt werden können.

Jede Information ist anhand des so genannten Datenidentifikators (ANSI MH10.8.2-2016) eindeutig identifizierbar. Dem Datenidentifikator folgt eine Zeichenkette. Beide zusammen haben eine maximale Länge gemäß nachstehender Tabelle. Sind die Informationen kürzer, werden sie um Leerzeichen ergänzt.

Folgende Informationen sind möglich, die Positionen 1 bis 4 sind immer vorhanden, die weiteren je nach Produktfamilienbedarf:

Pos-Nr.	Art der Information	Erklärung	Datenidentifikator	Anzahl Stellen inkl. Datenidentifikator	Beispiel
1	Beckhoff-Artikelnummer	Beckhoff - Artikelnummer	1P	8	1P072222
2	Beckhoff Traceability Number (BTN)	Eindeutige Seriennummer, Hinweis s. u.	SBTN	12	SBTNk4p562d7
3	Artikelbezeichnung	Beckhoff Artikelbezeichnung, z. B. EL1008	1K	32	1KEL1809
4	Menge	Menge in Verpackungseinheit, z. B. 1, 10...	Q	6	Q1
5	Chargennummer	Optional: Produktionsjahr und -woche	2P	14	2P401503180016
6	ID-/Seriennummer	Optional: vorheriges Seriennummer-System, z. B. bei Safety-Produkten oder kalibrierten Klemmen	51S	12	51S678294
7	Variante	Optional: Produktvarianten-Nummer auf Basis von Standardprodukten	30P	12	30PF971, 2*K183
...					

Weitere Informationsarten und Datenidentifikatoren werden von Beckhoff verwendet und dienen internen Prozessen.

Aufbau des BIC

Beispiel einer zusammengesetzten Information aus den Positionen 1 bis 4 und dem o.a. Beispielwert in Position 6. Die Datenidentifikatoren sind in Fettschrift hervorgehoben:

1P072222**SBTN**k4p562d**71KEL**1809 **Q1 51S**678294

Entsprechend als DMC:



Abb. 3: Beispiel-DMC **1P072222SBTN**k4p562d**71KEL**1809 **Q1 51S**678294

BTN

Ein wichtiger Bestandteil des BICs ist die Beckhoff Traceability Number (BTN, Pos.-Nr. 2). Die BTN ist eine eindeutige, aus acht Zeichen bestehende Seriennummer, die langfristig alle anderen Seriennummern-Systeme bei Beckhoff ersetzen wird (z. B. Chargenbezeichnungen auf IO-Komponenten, bisheriger Seriennummernkreis für Safety-Produkte, etc.). Die BTN wird ebenfalls schrittweise eingeführt, somit kann es vorkommen, dass die BTN noch nicht im BIC codiert ist.

HINWEIS

Diese Information wurde sorgfältig erstellt. Das beschriebene Verfahren wird jedoch ständig weiterentwickelt. Wir behalten uns das Recht vor, Verfahren und Dokumentation jederzeit und ohne Ankündigung zu überarbeiten und zu ändern. Aus den Angaben, Abbildungen und Beschreibungen in dieser Dokumentation können keine Ansprüche auf Änderung geltend gemacht werden.

1.5.4 Elektronischer Zugriff auf den BIC (eBIC)

Elektronischer BIC (eBIC)

Der Beckhoff Identification Code (BIC) wird auf Beckhoff-Produkten außen sichtbar aufgebracht. Er soll, wo möglich, auch elektronisch auslesbar sein.

Für die elektronische Auslesung ist die Schnittstelle entscheidend, über die das Produkt angesprochen werden kann.

K-Bus Geräte (IP20, IP67)

Für diese Geräte ist derzeit keine elektronische Speicherung und Auslesung geplant.

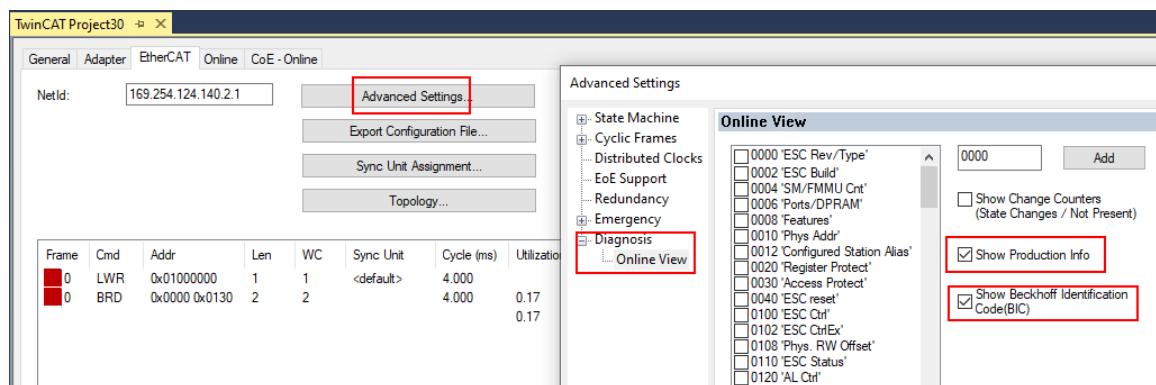
EtherCAT-Geräte (IP20, IP67)

Alle Beckhoff EtherCAT-Geräte haben ein sogenanntes ESI-EEPROM, das die EtherCAT-Identität mit der Revision beinhaltet. Darin wird die EtherCAT-Slave-Information gespeichert, umgangssprachlich auch als ESI/XML-Konfigurationsdatei für den EtherCAT-Master bekannt. Zu den Zusammenhängen siehe die entsprechenden Kapitel im EtherCAT-Systemhandbuch ([Link](#)).

In das ESI-EEPROM wird durch Beckhoff auch die eBIC geschrieben. Die Einführung des eBIC in die Beckhoff-IO-Produktion (Klemmen, Box-Module) erfolgt ab 2020; Stand 2023 ist die Umsetzung weitgehend abgeschlossen.

Anwenderseitig ist die eBIC (wenn vorhanden) wie folgt elektronisch zugänglich:

- Bei allen EtherCAT-Geräten kann der EtherCAT-Master (TwinCAT) den eBIC aus dem ESI-EEPROM auslesen:
 - Ab TwinCAT 3.1 Build 4024.11 kann der eBIC im Online-View angezeigt werden.
 - Dazu unter EtherCAT → Erweiterte Einstellungen → Diagnose das Kontrollkästchen „Show Beckhoff Identification Code (BIC)“ aktivieren:



- Die BTN und Inhalte daraus werden dann angezeigt:

No	Addr	Name	State	CRC	Fw	Hw	Production Data	ItemNo	BTN	Description	Quantity	BatchNo	SerialNo
1	1001	Term 1 (EK1100)	OP	0.0	0	0	---						
2	1002	Term 2 (EL1018)	OP	0.0	0	0	2020 KW36 Fr	072222	k4p562d7	EL1809	1		678294
3	1003	Term 3 (EL3204)	OP	0.0	7	6	2012 KW24 Sa						
4	1004	Term 4 (EL2004)	OP	0.0	0	0	---	072223	k4p562d7	EL2004	1		678295
5	1005	Term 5 (EL1008)	OP	0.0	0	0	---						
6	1006	Term 6 (EL2008)	OP	0.0	0	12	2014 KW14 Mo						
7	1007	Term 7 (EK1110)	OP	0	1	8	2012 KW25 Mo						

- Hinweis: ebenso können wie in der Abbildung zu sehen die seit 2012 programmierten Produktionsdaten HW-Stand, FW-Stand und Produktionsdatum per „Show Production Info“ angezeigt werden.
- Zugriff aus der PLC: Ab TwinCAT 3.1. Build 4024.24 stehen in der Tc2_EtherCAT Library ab v3.3.19.0 die Funktionen `FB_EcReadBIC` und `FB_EcReadBTN` zum Einlesen in die PLC bereit.

- Bei EtherCAT-Geräten mit CoE-Verzeichnis kann zusätzlich das Objekt 0x10E2:01 zur Anzeige der eigenen eBIC vorhanden sein, auch hierauf kann die PLC einfach zugreifen:
 - Das Gerät muss zum Zugriff in PREOP/SAFEOP/OP sein

Index	Name	Flags	Value
1000	Device type	RO	0x015E1389 (22942601)
1008	Device name	RO	ELM3704-0000
1009	Hardware version	RO	00
100A	Software version	RO	01
100B	Bootloader version	RO	J0.1.27.0
1011:0	Restore default parameters	RO	> 1 <
1018:0	Identity	RO	> 4 <
10E2:0	Manufacturer-specific Identification C...	RO	> 1 <
10E2:01	Subindex 001	RO	1P158442SBTN0008jekp1KELM3704 Q1 2P482001000016
10F0:0	Backup parameter handling	RO	> 1 <
10F3:0	Diagnosis History	RO	> 21 <
10F8	Actual Time Stamp	RO	0x170bfb277e

- Das Objekt 0x10E2 wird in Bestandsprodukten vorrangig im Zuge einer notwendigen Firmware-Überarbeitung eingeführt.
- Ab TwinCAT 3.1. Build 4024.24 stehen in der Tc2_EtherCAT Library ab v3.3.19.0 die Funktionen *FB_EcCoEReadBIC* und *FB_EcCoEReadBTN* zum Einlesen in die PLC zur Verfügung
- Zur Verarbeitung der BIC/BTN Daten in der PLC stehen noch als Hilfsfunktionen ab TwinCAT 3.1 Build 4024.24 in der *Tc2_Uutilities* zur Verfügung
 - *F_SplitBIC*: Die Funktion zerlegt den BIC sBICValue anhand von bekannten Kennungen in seine Bestandteile und liefert die erkannten Teil-Strings in einer Struktur *ST_SplittedBIC* als Rückgabewert
 - *BIC_TO_BTN*: Die Funktion extrahiert vom BIC die BTN und liefert diese als Rückgabewert
- Hinweis: bei elektronischer Weiterverarbeitung ist die BTN als String(8) zu behandeln, der Identifier „SBTN“ ist nicht Teil der BTN.
- Zum technischen Hintergrund:
Die neue BIC Information wird als Category zusätzlich bei der Geräteproduktion ins ESI-EEPROM geschrieben. Die Struktur des ESI-Inhalts ist durch ETG Spezifikationen weitgehend vorgegeben, demzufolge wird der zusätzliche herstellerspezifische Inhalt mithilfe einer Category nach ETG.2010 abgelegt. Durch die ID 03 ist für alle EtherCAT-Master vorgegeben, dass sie im Updatefall diese Daten nicht überschreiben bzw. nach einem ESI-Update die Daten wiederherstellen sollen.
Die Struktur folgt dem Inhalt des BIC, siehe dort. Damit ergibt sich ein Speicherbedarf von ca. 50..200 Byte im EEPROM.
- Sonderfälle
 - Bei einer hierarchischen Anordnung mehrerer ESC (EtherCAT Slave Controller) in einem Gerät trägt lediglich der oberste ESC die eBIC-Information.
 - Sind mehrere ESC in einem Gerät verbaut die nicht hierarchisch angeordnet sind, tragen alle ESC die eBIC-Information gleich.
 - Besteht das Gerät aus mehreren Sub-Geräten mit eigener Identität, aber nur das TopLevel-Gerät ist über EtherCAT zugänglich, steht im CoE-Objekt-Verzeichnis 0x10E2:01 die eBIC dieses ESC, in 0x10E2:nn folgen die eBIC der Sub-Geräte.

PROFIBUS-, PROFINET-, DeviceNet®-Geräte usw.

Für diese Geräte ist derzeit keine elektronische Speicherung und Auslesung geplant.

2 Produktübersicht

2.1 Einführung

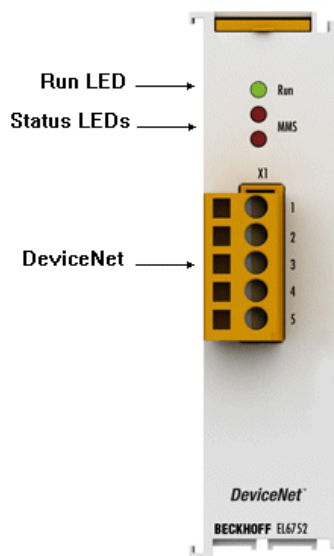


Abb. 4: EL6752

Master- und Slave-Klemmen für DeviceNet®

Die Master- und Slave-Klemmen für DeviceNet® entsprechen der Beckhoff PCI-Karte FC5201. Durch den Anschluss via EtherCAT kann im PC auf PCI-Slots verzichtet werden. Im EtherCAT-Klemmenverbund erlaubt die Klemme die Integration beliebiger DeviceNet®-Geräte.

Die EL6752 ist wahlweise als Master- oder Slave-Ausführung erhältlich und verfügt über eine leistungsfähige Protokollimplementierung mit vielen Features:

- Alle I/O-Modi des DeviceNet® werden unterstützt: Polling, Change-of-State, Cyclic, Strobed
- Unconnected-Message-Manager (UCMM)
- Leistungsfähige Parameter- und Diagnoseschnittstellen
- Error-Management für jeden Busteilnehmer frei konfigurierbar

Eine Beschreibung aller Funktionalitäten und Betriebsmodi finden Sie im Kapitel "[Konfiguration](#) [► 73]" und den entsprechenden Unterkapiteln.

2.2 Technische Daten

Technische Daten	EL6752-0000	EL6752-0010
Bus-System	DeviceNet®	
Variante	Master	Slave
Anzahl Feldbuskanäle	1	
Übertragungsrate	125, 250 oder 500 kBaud	
Bus-Interface	Open-Style-Connector 5-polig gemäß DeviceNet®-Spezifikation, galvanisch entkoppelt, Stecker wird mitgeliefert	
Busteilnehmer	maximal 63 Slaves	
Kommunikation	DeviceNet®-Netzwerkmaster (Scanner)	DeviceNet®-Slave
Diagnose	Status-LEDs	
Spannungsversorgung	über den E-Bus	
Stromaufnahme aus dem E-Bus	typ. 260 mA	
Potenzialtrennung	500 V (E-Bus/CANopen)	
Konfiguration	mit TwinCAT System-Manager	
Gewicht	ca. 70 g	
zulässiger Umgebungstemperaturbereich im Betrieb	-25°C ... +60°C	
zulässiger Umgebungstemperaturbereich bei Lagerung	-40°C ... +85°C	
zulässige relative Luftfeuchtigkeit	95%, keine Betauung	
Abmessungen (B x H x T)	ca. 26 mm x 100 mm x 52 mm	
Montage [► 23]	auf 35 mm Tragschiene nach EN 60715	
Vibrations- / Schockfestigkeit	gemäß EN 60068-2-6 / EN 60068-2-27	
EMV-Festigkeit / Aussendung	gemäß EN 61000-6-2 / EN 61000-6-4	
Schutzart	IP20	
Einbaulage	beliebig	
Zulassungen / Kennzeichnungen*	CE, UKCA, EAC, CCC cULus [► 22] , ATEX [► 19] , IECEx [► 20]	

*) Real zutreffende Zulassungen/Kennzeichnungen siehe seitliches Typenschild (Produktbeschriftung).

Ex-Kennzeichnungen

Standard	Kennzeichnung
ATEX	II 3 G Ex ec IIC T4 Gc
IECEx	Ex ec IIC T4 Gc

3 DeviceNet® Grundlagen

Systemvorstellung

DeviceNet® ist ein offenes System das auf der Basis von CAN aufsetzt. CAN wurde vor einigen Jahren von der Firma R. Bosch für die Datenübertragung in Kraftfahrzeugen entwickelt. Seitdem sind Millionen von CAN-Chips im Einsatz. Nachteilig für einen Einsatz in der Automatisierungstechnik ist, dass CAN keine Definitionen für die Applikationsschicht enthält. CAN definiert nur die physikalische und Datensicherungsschicht.

Mit DeviceNet® ist eine einheitliche Applikationsschicht festgelegt, mit der das CAN-Protokoll für Industrieanwendungen nutzbar wird. Die ODVA (Open DeviceNet® Vendor Association) unterstützt Hersteller und Anwender des Systems DeviceNet® als unabhängiger Verein. Die ODVA stellt sicher, dass alle Geräte, die der Spezifikation entsprechen, herstellernerutral zusammen in einem System arbeiten. CAN bietet durch das Verfahren der Bitarbitration grundsätzlich die Möglichkeit Kommunikationsnetze mit Master/ Slave- und Multimaster- Zugriffsverfahren zu betreiben.

Weitere Details finden Sie auf der offiziellen Webseite der ODVA (<http://www.odva.org>).

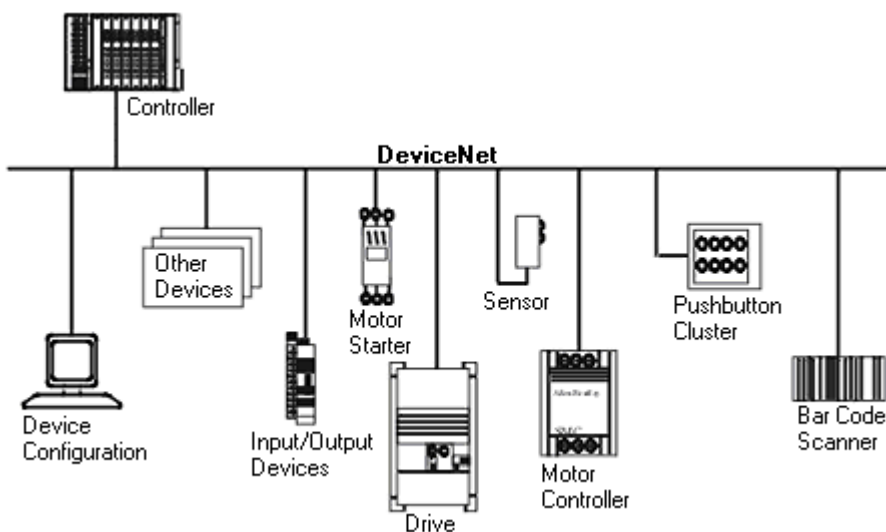


Abb. 5: Beispiel für eine Vernetzung mit DeviceNet®

Buskabel

Als Buskabel dient eine 2x2 adrig verdrehte und geschirmte Leitung. Ein Adernpaar übernimmt die Datenübertragung und das zweite Adernpaar ist für die Stromversorgung zuständig. Es kann Ströme bis zu 8 Ampere führen. Die maximal zulässige Leitungslänge ist vorrangig von der Baudrate abhängig. Wird die höchste Baudrate (500 kBaud) gewählt, sind max. 100 m Leitung zulässig. Bei der geringsten Baudrate (125 kBaud) ist eine gesamte Kabellänge von 500 m realisierbar. Details siehe im Kapitel "Montage und Verdrahtung" [► 27]

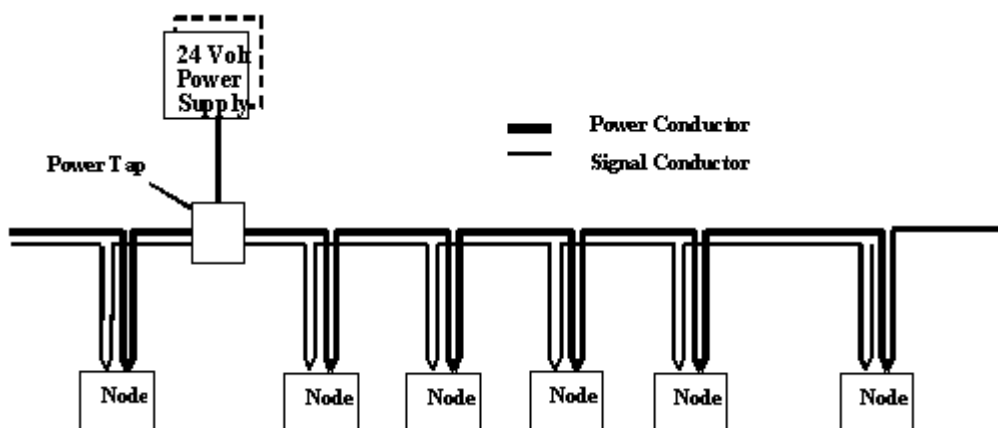


Abb. 6: Beispiel DeviceNet® Verkabelung

4 Montage und Verkabelung

4.1 Hinweise zum ESD-Schutz

HINWEIS

Zerstörung der Geräte durch elektrostatische Aufladung möglich!

Die Geräte enthalten elektrostatisch gefährdete Bauelemente, die durch unsachgemäße Behandlung beschädigt werden können.

- Beim Umgang mit den Bauteilen ist auf elektrostatische Entladung zu achten; außerdem ist das direkte Berühren der Federkontakte (siehe Abbildung) zu vermeiden.
- Der Kontakt mit hoch isolierenden Stoffen (Kunstfasern, Kunststofffolien etc.) sollte beim gleichzeitigen Umgang mit Komponenten vermieden werden.
- Beim Umgang mit den Komponenten ist auf eine sachgemäße Erdung der Umgebung (Arbeitsplatz, Verpackung und Personen) zu achten.
- Jede Busstation muss auf der rechten Seite mit der Endkappe [EL9011](#) oder [EL9012](#) abgeschlossen werden, um die Schutzart und den ESD-Schutz zu gewährleisten.

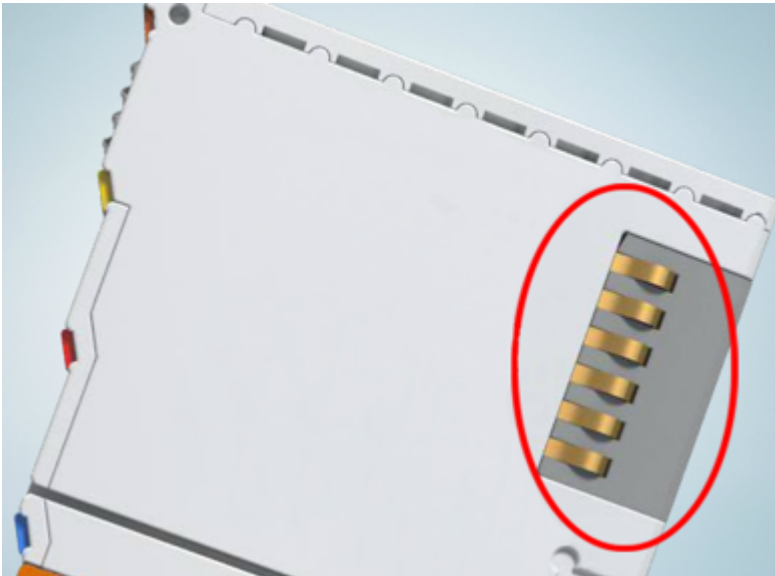


Abb. 7: Federkontakte der Beckhoff I/O-Komponenten

4.2 Explosionsschutz

4.2.1 ATEX - Besondere Bedingungen (erweiterter Temperaturbereich)

⚠️ WARNUNG

Beachten Sie die besonderen Bedingungen für die bestimmungsgemäße Verwendung von Beckhoff-Feldbuskomponenten mit erweitertem Temperaturbereich (ET) in explosionsgefährdeten Bereichen!

- Die zertifizierten Komponenten sind in ein geeignetes Gehäuse zu errichten, das eine Schutzart von mindestens IP54 gemäß EN 60079-7 gewährleistet! Dabei sind die Umgebungsbedingungen bei der Verwendung zu berücksichtigen!
- Wenn die Temperaturen bei Nennbetrieb an den Einführungsstellen der Kabel, Leitungen oder Rohrleitungen höher als 70°C oder an den Aderverzweigungsstellen höher als 80°C ist, so müssen Kabel ausgewählt werden, deren Temperaturdaten den tatsächlich gemessenen Temperaturwerten entsprechen!
- Beachten Sie für Beckhoff-Feldbuskomponenten mit erweitertem Temperaturbereich (ET) beim Einsatz in explosionsgefährdeten Bereichen den zulässigen Umgebungstemperaturbereich von -25 bis 60°C!
- Es müssen Maßnahmen zum Schutz gegen Überschreitung der Nennbetriebsspannung durch kurzzeitige Störspannungen um mehr als 40% getroffen werden!
- Die einzelnen Klemmen dürfen nur aus dem Busklemmensystem gezogen oder entfernt werden, wenn die Versorgungsspannung abgeschaltet wurde bzw. bei Sicherstellung einer nicht-explosionsfähigen Atmosphäre!
- Die Anschlüsse der zertifizierten Komponenten dürfen nur verbunden oder unterbrochen werden, wenn die Versorgungsspannung abgeschaltet wurde bzw. bei Sicherstellung einer nicht-explosionsfähigen Atmosphäre!
- Die Sicherung der Einspeiseklemmen KL92xx/EL92xx dürfen nur gewechselt werden, wenn die Versorgungsspannung abgeschaltet wurde bzw. bei Sicherstellung einer nicht-explosionsfähigen Atmosphäre!
- Adresswahlschalter und ID-Switche dürfen nur eingestellt werden, wenn die Versorgungsspannung abgeschaltet wurde bzw. bei Sicherstellung einer nicht-explosionsfähigen Atmosphäre!

Normen

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

- EN IEC 60079-0:2018
- EN 60079-7:2015+A1:2018

Kennzeichnung

Die gemäß den Regulierungen für den explosionsgefährdeten Bereich zertifizierten Beckhoff-Feldbuskomponenten mit erweitertem Temperaturbereich (ET) tragen die folgende Kennzeichnung:



II 3 G Ex ec IIC T4 Gc
KEMA 10 ATEX0075 X
DEKRA 22UKEX6024X
Ta: -25 ... + 60°C

4.2.2 IECEx - Besondere Bedingungen

WARNUNG

Beachten Sie die besonderen Bedingungen für die bestimmungsgemäße Verwendung von Beckhoff-Feldbuskomponenten in explosionsgefährdeten Bereichen!

- Für Gas: Die Komponenten sind in ein geeignetes Gehäuse zu errichten, das gemäß IEC 60079-7 eine Schutzart von IP54 gewährleistet! Dabei sind die Umgebungsbedingungen bei der Verwendung zu berücksichtigen!
- Die Komponenten dürfen nur in einem Bereich mit mindestens Verschmutzungsgrad 2 gemäß IEC 60664-1 verwendet werden!
- Es sind Vorkehrungen zu treffen, um zu verhindern, dass die Nennspannung durch transiente Störungen von mehr als 119 V überschritten wird!
- Wenn die Temperaturen bei Nennbetrieb an den Einführungsstellen der Kabel, Leitungen oder Rohrleitungen höher als 70°C oder an den Aderverzweigungsstellen höher als 80°C ist, so müssen Kabel ausgewählt werden, deren Temperaturdaten den tatsächlich gemessenen Temperaturwerten entsprechen!
- Beachten Sie für Beckhoff-Feldbuskomponenten beim Einsatz in explosionsgefährdeten Bereichen den zulässigen Umgebungstemperaturbereich!
- Die einzelnen Klemmen dürfen nur aus dem Busklemmensystem gezogen oder entfernt werden, wenn die Versorgungsspannung abgeschaltet wurde bzw. bei Sicherstellung einer nicht-explosionsfähigen Atmosphäre!
- Die Anschlüsse der zertifizierten Komponenten dürfen nur verbunden oder unterbrochen werden, wenn die Versorgungsspannung abgeschaltet wurde bzw. bei Sicherstellung einer nicht-explosionsfähigen Atmosphäre!
- Adresswahlschalter und ID-Switche dürfen nur eingestellt werden, wenn die Versorgungsspannung abgeschaltet wurde bzw. bei Sicherstellung einer nicht-explosionsfähigen Atmosphäre!
- Die Frontklappe von zertifizierten Geräten darf nur geöffnet werden, wenn die Versorgungsspannung abgeschaltet wurde bzw. bei Sicherstellung einer nicht-explosionsfähigen Atmosphäre!

Normen

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

- IEC 60079-0:2017 (Edition 7.0)
- IEC 60079-7:2017 (Edition 5.1)

Kennzeichnung

Die gemäß IECEx für den explosionsgefährdeten Bereich zertifizierten Beckhoff-Feldbuskomponenten tragen die folgende Kennzeichnung:

IECEx DEK 16.0078 X

Ex ec IIC T4 Gc

4.2.3 Weiterführende Dokumentation zu ATEX und IECEx

HINWEIS



Weiterführende Dokumentation zum Explosionsschutz gemäß ATEX und IECEx




Beachten Sie auch die weiterführende Dokumentation

Explosionsschutz für Klemmensysteme

Hinweise zum Einsatz der Beckhoff Klemmensysteme in explosionsgefährdeten Bereichen gemäß ATEX und IECEx,

die Ihnen auf der Beckhoff-Homepage www.beckhoff.de im Download-Bereich Ihres Produktes zum Download zur Verfügung steht!

4.3 UL-Hinweise

⚠ VORSICHT	
	Application The modules are intended for use with Beckhoff's UL Listed EtherCAT System only.
⚠ VORSICHT	
	Examination For cULus examination, the Beckhoff I/O System has only been investigated for risk of fire and electrical shock (in accordance with UL508 and CSA C22.2 No. 142).
⚠ VORSICHT	
	For devices with Ethernet connectors Not for connection to telecommunication circuits.

Grundlagen

UL-Zertifikation nach UL508. Solcherart zertifizierte Geräte sind gekennzeichnet durch das Zeichen:



4.4 Montage und Demontage - Zughebelentriegelung

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



Tragschienenbefestigung

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

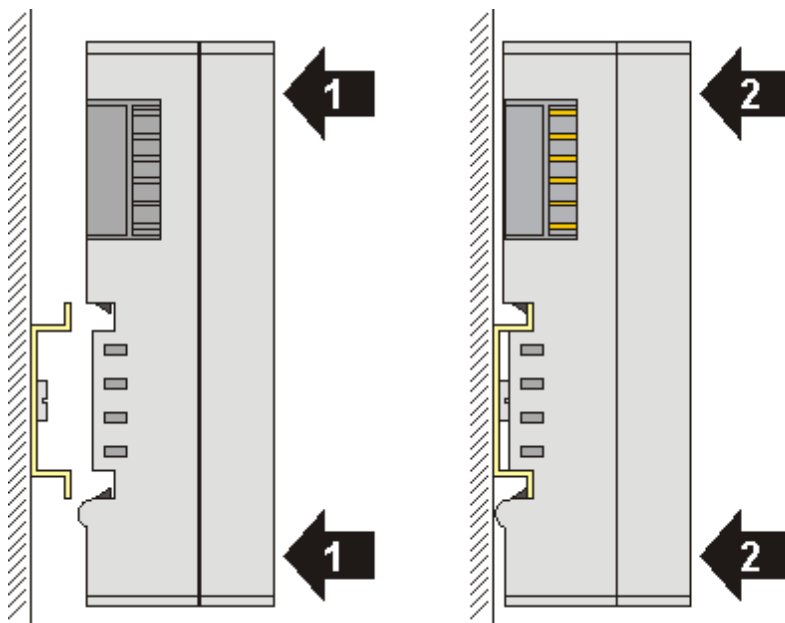
WARNUNG

Verletzungsgefahr durch Stromschlag und Beschädigung des Gerätes möglich!

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

Montage

- Montieren Sie die Tragschiene an der vorgesehenen Montagestelle

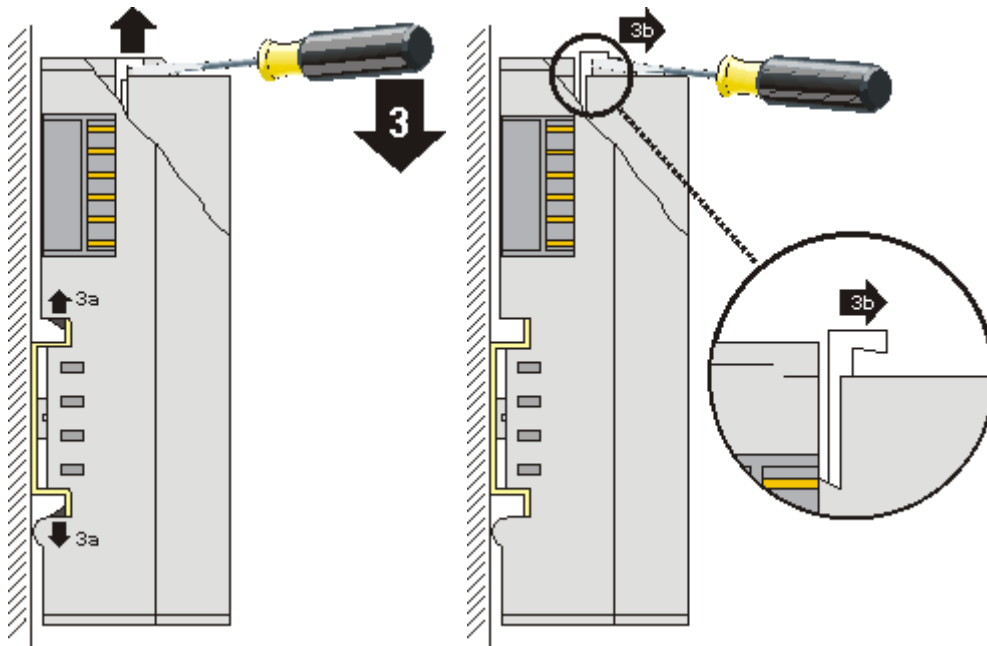


und drücken Sie (1) das Klemmenmodul gegen die Tragschiene, bis es auf der Tragschiene Einrastet (2).

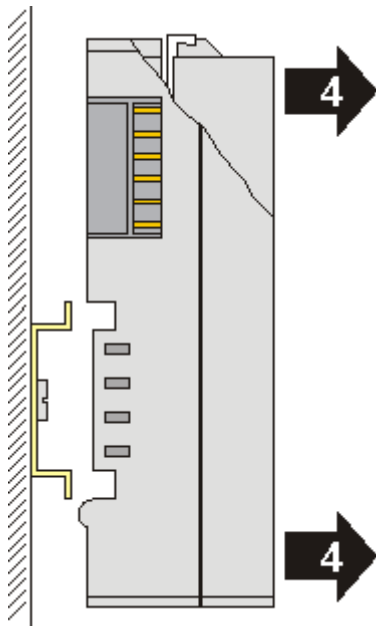
- Schließen Sie die Leitungen an.

Demontage

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



- Bei 32- und 64-kanaligen Klemmenmodulen (KMxxx4 und KMxxx8 bzw. EMxxx4 und EMxxx8) hebeln Sie nun den zweiten Entriegelungshaken auf der rechten Seite des Klemmenmoduls auf die gleiche Weise nach oben.
- Ziehen Sie (4) das Klemmenmodul von der Montagefläche weg.



4.5 Montage und Demontage - Frontentriegelung oben

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

i Tragschienenbefestigung

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

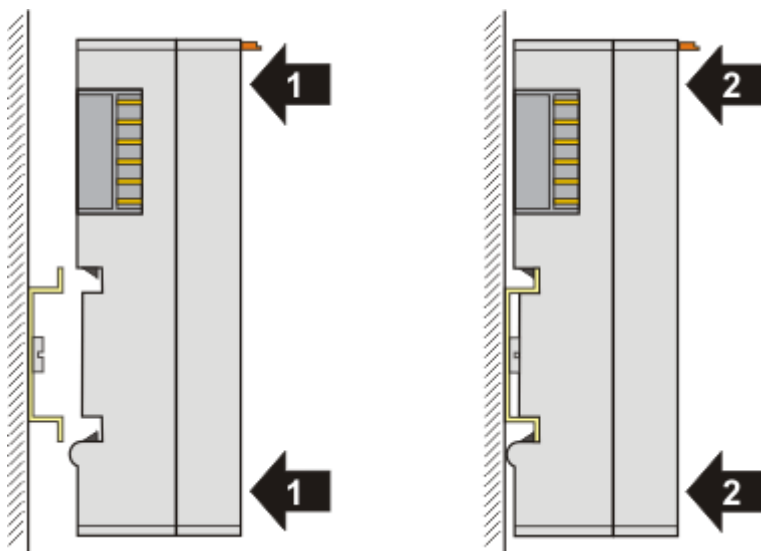
⚠️ WARNUNG

Verletzungsgefahr durch Stromschlag und Beschädigung des Gerätes möglich!

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

Montage

- Montieren Sie die Tragschiene an der vorgesehenen Montagestelle

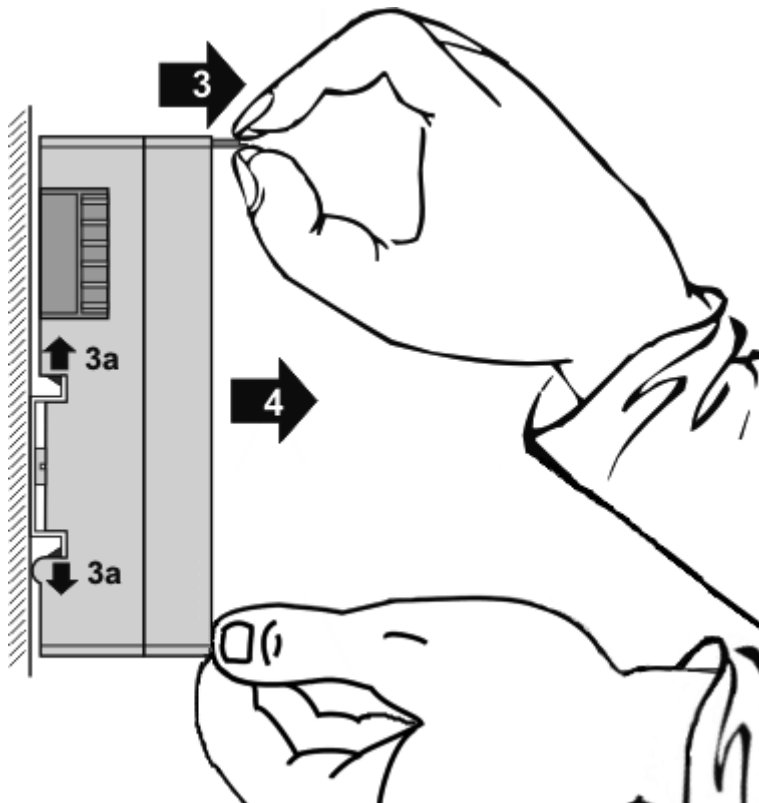


und drücken Sie (1) das Klemmenmodul gegen die Tragschiene, bis es auf der Tragschiene einrastet (2).

- Schließen Sie die Leitungen an.

Demontage

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



- Ziehen Sie (4) das Klemmenmodul von der Montagefläche weg. Vermeiden Sie ein Verkanten; stabilisieren Sie das Modul ggf. mit der freien Hand

4.6 Entsorgung



Die mit einer durchgestrichenen Abfalltonne gekennzeichneten Produkte dürfen nicht in den Hausmüll. Das Gerät gilt bei der Entsorgung als Elektro- und Elektronik-Altgerät. Die nationalen Vorgaben zur Entsorgung von Elektro- und Elektronik-Altgeräten sind zu beachten.

4.7 DeviceNet® Verkabelung

4.7.1 CAN / DeviceNet® Topologie

CAN / DeviceNet ist ein 2-Draht-Bussystem, an dem alle Teilnehmer parallel (d.h. mit kurzen Stichleitungen) angeschlossen werden (Abb. *DeviceNet Topologie*). Der Bus muss an jedem Ende mit einem Abschlusswiderstand von 120 (bzw. 121) Ohm abgeschlossen werden, um Reflexionen zu vermeiden. Dies ist auch bei sehr kurzen Leitungslängen erforderlich!

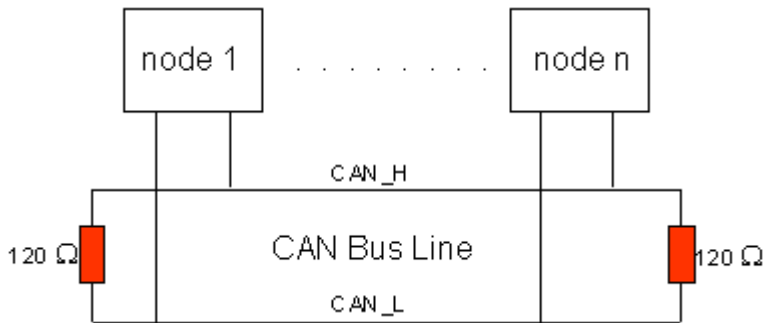


Abb. 8: DeviceNet Topologie

Da die CAN-Signale als Differenzpegel auf dem Bus dargestellt werden, ist die CAN-Leitung vergleichsweise unempfindlich gegen eingetragte Störungen (EMI). Es sind jeweils beide Leitungen betroffen, somit verändert die Störung den Differenzpegel kaum.

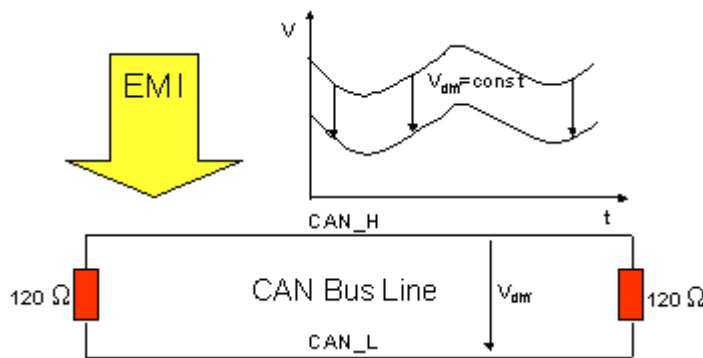


Abb. 9: Geringe Störeinwirkung durch Differenzpegel

4.7.2 Buslänge

Die maximale Buslänge wird bei CAN vorwiegend durch die Signallaufzeit beschränkt. Das Multi-Master-Buszugriffsverfahren (Arbitrierung) erfordert, dass die Signale quasi gleichzeitig (vor der Abtastung innerhalb einer Bitzeit) an allen Knoten anliegen. Da die Signallaufzeit in den CAN-Anschaltungen (Transceiver, Optokoppler, CAN-Controller) nahezu konstant sind, muss die Leitungslänge an die Baud-Rate angepasst werden.

Baud-Rate	Buslänge
500 kBit/s	< 100 m
250 kBit/s	< 250 m
125 kBit/s	< 500 m

4.7.3 Stichleitungen

Stichleitungen ("drop lines") sind nach Möglichkeit zu vermeiden, da sie grundsätzlich zu Signalreflexionen führen. Die durch Stichleitungen hervorgerufenen Reflexionen sind jedoch in der Regel unkritisch, wenn sie vor dem Abtastzeitpunkt vollständig abgeklungen sind. Bei den in den Buskopplern gewählten Bit-Timing-Einstellungen kann dies angenommen werden, wenn folgende Stichleitungslängen nicht überschritten werden:

Baud-Rate	Länge Stichleitung	gesamte Länge aller Stichleitungen
500 kBit/s	< 6 m	< 39 m
250 kBit/s	< 6 m	< 78 m
125 kBit/s	< 6 m	< 156 m

Stichleitungen dürfen nicht mit Abschlusswiderständen versehen werden (Abb. *Topologie Stichleitungen*).

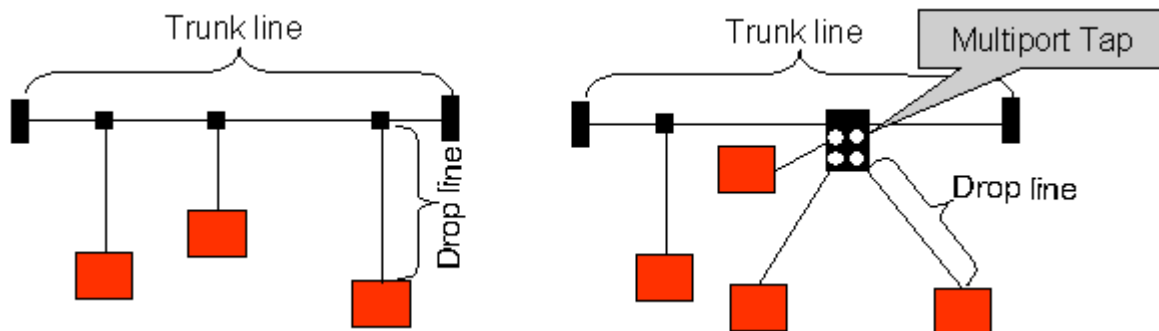


Abb. 10: Topologie Stichleitungen

4.7.4 Sternverteiler (Multiport Tap)

Beim Einsatz von passiven Verteilern ("Multiport Taps"), z. B. der Beckhoff Verteilerbox ZS5052-4500 sind kürzere Stichleitungslängen einzuhalten. Die folgende Tabelle gibt die maximalen Stichleitungslängen und die maximale Länge der Trunk Line (ohne Stichleitungen) an:



Richtwerte

Die folgenden Werte sind Empfehlungen von BECKHOFF!

Baud-Rate	Länge Stichleitung bei Multiport Topologie	Länge Trunk Line (ohne Stichleitungen)
500 kBit/s	< 1,2 m	< 66 m
250 kBit/s	< 2,4 m	< 120 m
125 kBit/s	< 4,8 m	< 310 m

4.7.5 CAN-Kabel

Für die CAN-Verdrahtung wird die Verwendung von paarig verdrehten, geschirmten Kabeln (2x2) mit einem Wellenwiderstand von 108...132 Ohm empfohlen. Wenn das Bezugspotential der CAN-Transceiver (CAN-Ground) nicht verbunden werden soll, so kann auf das zweite Adernpaar verzichtet werden (nur bei kleinen Netzausdehnungen mit gemeinsamer Speisung aller Teilnehmer empfehlenswert).

ZB5200 CAN / DeviceNet®-Kabel

Das Kabelmaterial ZB5200 entspricht der DeviceNet® Spezifikation und eignet sich ebenso für CANopen Systeme. Aus diesem Kabelmaterial sind auch die vorkonfektionierten Busleitungen ZK1052-xxxx-xxxx für die Feldbus Box Baugruppen gefertigt. Es hat folgende Spezifikation:

- 2 x 2 x 0,34 mm² (AWG 22) paarig verseilt
- doppelt geschirmt - Schirmgeflecht mit Beilaufitze
- Wellenwiderstand (1 MHz): 126 Ohm
- Leiterwiderstand 54 Ohm/km
- Mantel: PVC grau, Außendurchmesser 7,3 mm
- Bedruckt mit "InterlinkBT DeviceNet Type 572" sowie UL und CSA Ratings
- Litzenfarben entsprechen DeviceNet® Spezifikation
- UL anerkanntes AWM Type 2476 Rating
- CSA AWM I/II A/B 80°C 300V FT1
- Entspricht DeviceNet® "Thin Cable" Spezifikation

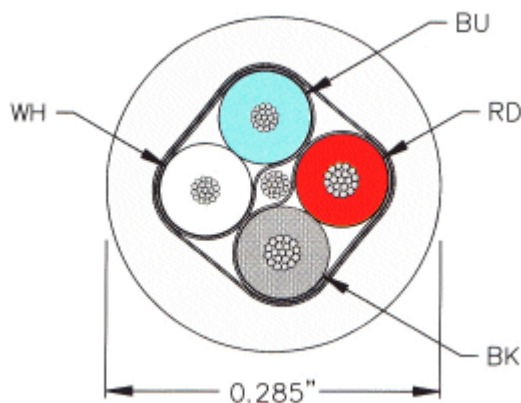


Abb. 11: Aufbau DeviceNet® Kabel

4.7.6 Schirmung

Der Schirm ist über das gesamte Buskabel zu verbinden und nur an einer Stelle galvanisch zu erden um Masseschleifen zu vermeiden.

Das Schirmungskonzept mit HF-Ableitung von Störungen über R/C-Glieder auf die Tragschiene geht davon aus, dass die Tragschiene entsprechend geerdet und störungsfrei ist. Sollte dies nicht der Fall sein, so kann es vorkommen, dass HF-Störpegel über die Tragschiene auf den Schirm des Buskabels übertragen werden. In diesem Fall sollte der Schirm an den Kopplern nicht aufgelegt werden - aber dennoch komplett durchverbunden sein.

4.7.7 Kabelfarben und Pin-Belegung



Abb. 12: Steckerbelegung (Draufsicht EL6752)

Vorschlag für die Verwendung der Beckhoff CAN-Kabel an Busklemme und Feldbus Box:

Pin	Belegung EL6752	Kabelfarbe ZB5200
1	V+ (24 V)	rot
2	CAN-High	weiß
3	Schirm	Beilaufleiste
4	CAN-Low	blau
5	V-	schwarz

4.8 Einbaulagen

HINWEIS

Einschränkung von Einbaulage und Betriebstemperaturbereich

- Entnehmen Sie den technischen Daten des Geräts, ob es Einschränkungen bei Einbaulage und/oder Betriebstemperaturbereich unterliegt.
- Sorgen Sie bei der Montage von Geräten mit erhöhter thermischer Verlustleistung dafür, dass im Betrieb oberhalb und unterhalb der Geräte ausreichend Abstand zu anderen Komponenten eingehalten wird, so dass die Belüftung gewährleistet ist!

Im Folgenden werden die Einbaulagen und deren Benennung für die Montage von Geräten auf Tragschienen definiert. Die Darstellung der Geräte in den folgenden Abbildungen ist exemplarisch. Für alle Einbaulagen gilt: Bezugsrichtung "unten" (siehe Pfeil) ist hier die Erdbeschleunigung.

Einbaulage Waagrecht (Standard-Einbaulage)

Die Tragschiene wird waagrecht an eine senkrechte Montageplatte montiert. Die Anschlussebene der Geräte weist nach vorne.

Die Geräte werden dabei von unten nach oben durchlüftet, was eine optimale Kühlung der Elektronik durch Konvektionslüftung ermöglicht. Deshalb stellt dies auch die empfohlene Einbaulage dar.

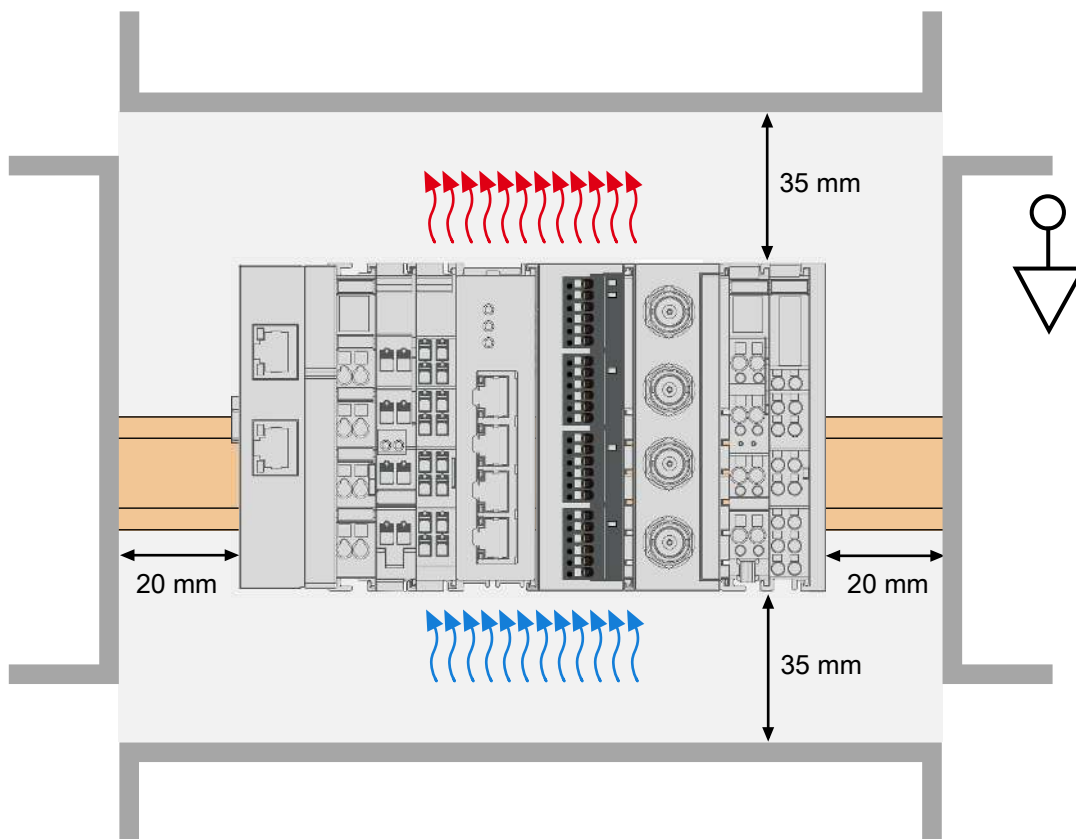


Abb. 13: Empfohlene Mindestabstände bei Standard-Einbaulage

HINWEIS

Einhaltung der Mindestabstände

Die Einhaltung der Mindestabstände gemäß Abbildung „Empfohlene Mindestabstände bei Standard Einbaulage“ wird in allen Einbaulagen dringend empfohlen.

Einbaulage Senkrecht

Die Tragschiene wird senkrecht an eine senkrechte Montageplatte montiert.
Die Anschlussebene der Geräte zeigt nach vorne.
Die Geräte können dabei wie folgt angeordnet werden:

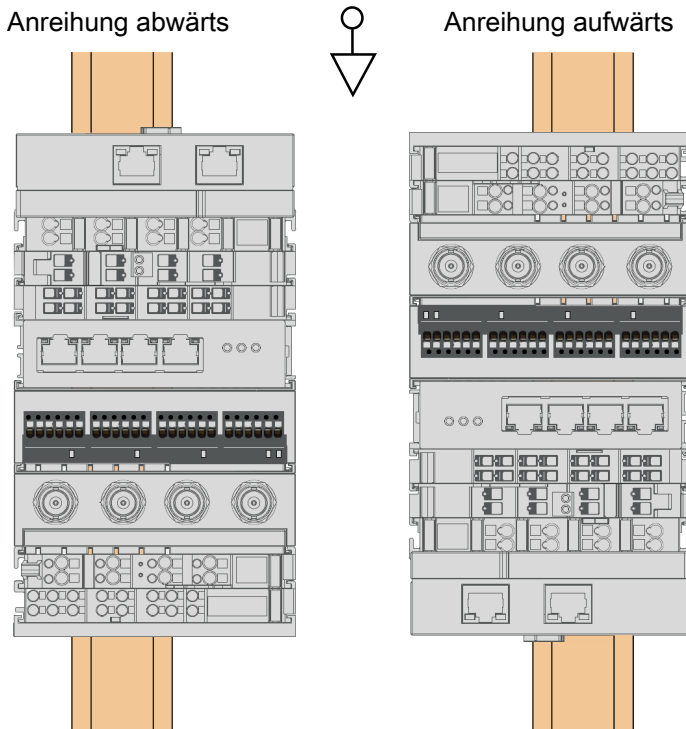


Abb. 14: Einbaulage Senkrecht, Anreihung abwärts (links) / Anreihung aufwärts (rechts)

Einbaulage Liegend

Die Tragschiene wird auf einer waagerechten liegenden Montageplatte montiert.
Die Anschlussebene der Geräte zeigt nach oben.



Abb. 15: Einbaulage Liegend

HINWEIS

Gefahr durch Abrutschen von der Tragschiene

Insbesondere in der Einbaulage „Senkrecht“, aber bei entsprechender mechanischer Belastung auch in anderen Einbaulagen, kann es zu Verschiebebewegungen des Klemmenstrangs auf der Tragschiene kommen. Diese können zu unerwünschten Fehlfunktionen führen.

- Wenn diese Gefahr besteht, sichern sie den Klemmenstrang durch entsprechende Arretierungen z. B. durch Schraubklemmung auf der Tragschiene.

HINWEIS

Einhaltung der Mindestabstände

Die Einhaltung der Mindestabstände gemäß Abbildung „Empfohlene Mindestabstände bei Standard Einbaulage“ wird in allen Einbaulagen dringend empfohlen.

Einbaulagen mit Lüftermodul **ZB8610**

Soll oder muss die Kühlung für den beabsichtigten Einsatzfall verstärkt werden, kann das Lüftermodul ZB8610 an der Geräteunterseite montiert werden. In Waagerechter Einbaulage werden die Geräte dabei vom Lüftermodul unterstützend von unten nach oben durchlüftet. Dabei wird die optimale Kühlung durch Konvektionslüftung zusätzlich verstärkt (s. folgende Abb.).

Der Einsatz des Lüftermoduls ist in jeder Einbaulage möglich.

Weitere Hinweise zum Betrieb mit und ohne Lüfter sind ggf. den Technischen Daten des Geräts zu entnehmen (z. B. Derating, Hinweise zu Einbaulagen etc.).

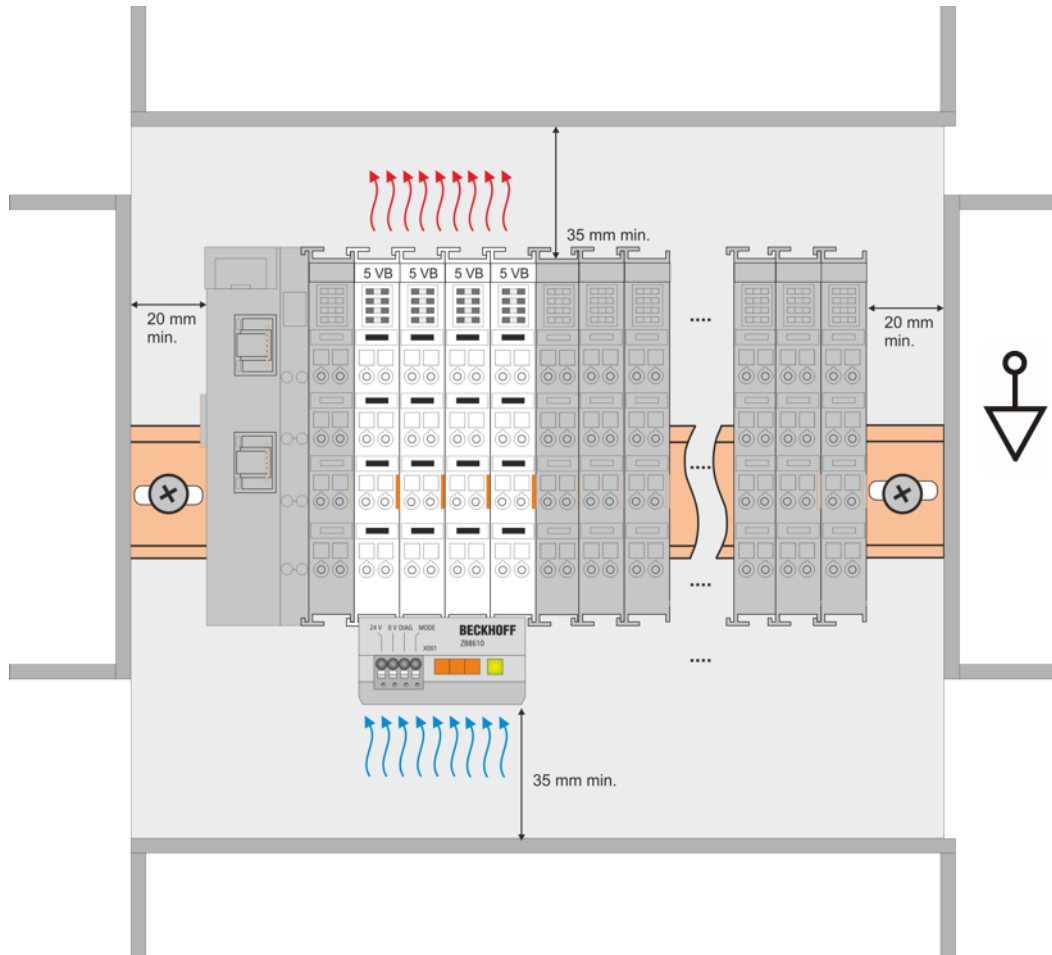


Abb. 16: Empfohlene Mindestabstände bei Betrieb mit Lüfter, am Beispiel Einbaulage Waagerecht

HINWEIS

Einhaltung der Mindestabstände

Die Einhaltung der Mindestabstände gemäß Abbildung „Empfohlene Mindestabstände bei Betrieb mit Lüfter“ wird dringend empfohlen.

4.9 Positionierung von passiven Klemmen

i Hinweis zur Positionierung von passiven Klemmen im Busklemmenblock

EtherCAT-Klemmen (ELxxxx / ESxxxx), die nicht aktiv am Datenaustausch innerhalb des Busklemmenblocks teilnehmen, werden als passive Klemmen bezeichnet. Diese Klemmen sind an der nicht vorhandenen Stromaufnahme aus dem E-Bus zu erkennen. Um einen optimalen Datenaustausch zu gewährleisten, dürfen nicht mehr als zwei passive Klemmen direkt aneinander gereiht werden!

Beispiele für die Positionierung von passiven Klemmen (hell eingefärbt)

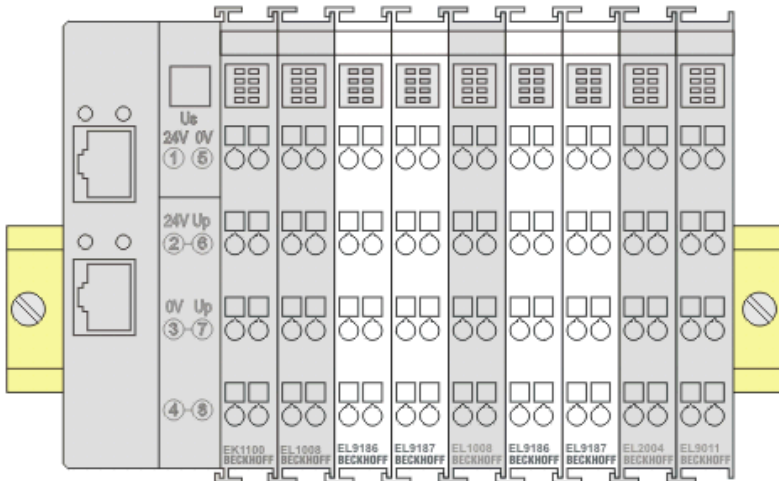


Abb. 17: Korrekte Positionierung

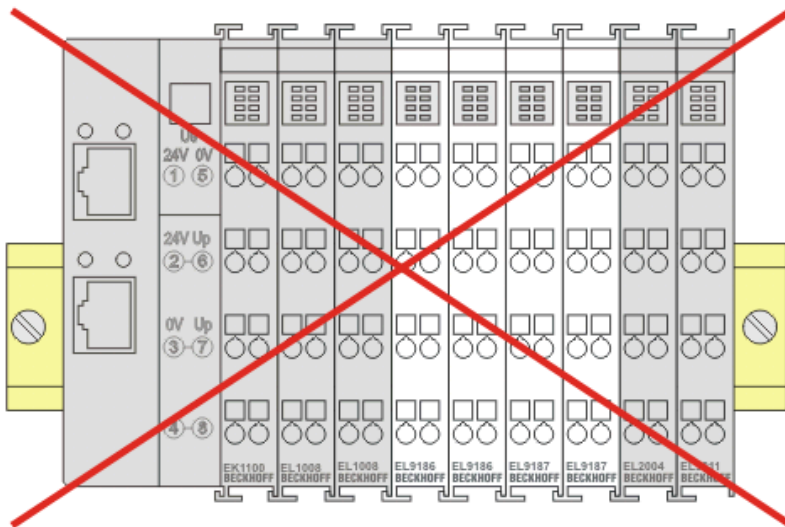


Abb. 18: Inkorrekte Positionierung

5 DeviceNet® Kommunikation

5.1 DeviceNet® Einführung



Abb. 19: DeviceNet®

DeviceNet® ist ein offenes System das auf der Basis von CAN aufsetzt. CAN wurde vor einigen Jahren von der Firma R. Bosch für die Datenübertragung in Kraftfahrzeugen entwickelt. Seitdem sind Millionen von CAN-Chips im Einsatz. Nachteilig für einen Einsatz in der Automatisierungstechnik ist, dass CAN keine Definitionen für die Applikationsschicht enthält. CAN definiert nur die physikalische und Datensicherungsschicht.

Mit DeviceNet® ist eine einheitliche Applikationsschicht festgelegt, mit der das CAN-Protokoll für Industrieanwendungen nutzbar wird. Die ODVA (Open DeviceNet Vendor Association) unterstützt Hersteller und Anwender des Systems DeviceNet® als unabhängiger Verein. Die ODVA stellt sicher, dass alle Geräte, die der Spezifikation entsprechen, herstellernerneutral zusammen in einem System arbeiten.

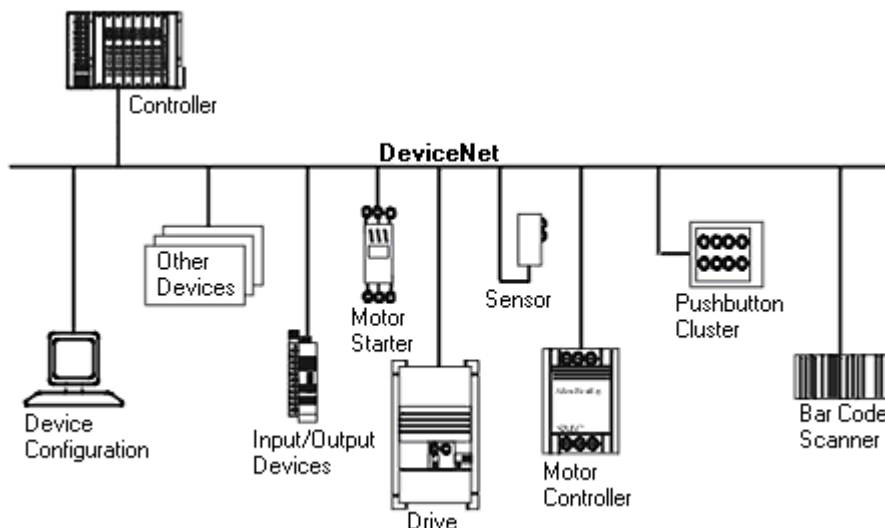


Abb. 20: Beispiel für eine Vernetzung mit DeviceNet®

DeviceNet® ist ein Sensor-/Aktorbussystem. Es ist international genormt (EN50325) und basiert auf CAN (Controller Area Net-work). DeviceNet® unterstützt mehrere Kommunikationsarten für die Ein- und Ausgangsdaten:

- Polling: die Masterbaugruppe („Scanner“) versendet die Ausgangsdaten zyklisch zu den zugeordneten Teilnehmern und erhält die Eingangsdaten im Antworttelegramm.
- Change-of-State: Telegramme werden versendet, sobald sich der Inhalt geändert hat.
- Cyclic: Die Baugruppen versenden die Daten nach Ablauf einer Zykluszeit selbstständig.
- Strobed: Der Scanner fordert die Eingangsdaten mit einem Broadcast-Telegramm an alle Teilnehmer an.

Die DeviceNet®-Geräte unterstützen alle I/O-Kommunikationsarten.

Parametriert werden Device-Net-Geräte über azyklische Dienste (Explicit-Messaging).

Durch die effektive Nutzung der Busbandbreite erreicht DeviceNet® vor allem im Change-of-State-Modus kurze Systemreaktionszeiten trotz der vergleichsweise niedrigen Datenraten. Die BECKHOFF DeviceNet®-Geräte verfügen über eine leistungsfähige Protokollimplementierung. Durch aktive Mitarbeit in den technischen Gremien der ODVA trägt BECKHOFF zur weiteren Entwicklung dieses Bussystems bei und verfügt damit über profundes DeviceNet®-Know-how.

Konfiguration

Die Knotenadresse wird mit zwei dezimal kodierten Drehwahlschaltern im Bereich von 0 bis 63 eingestellt. Die am DeviceNet®-Scanner eingestellte Übertragungsrate wird von der DeviceNet®-Box selbstständig erkannt (Auto-Baudrate). Für DeviceNet®-Konfigurationstools stehen „elektronische Datenblätter“ (EDS-Files) zum Download von der Beckhoff-Internetseite (<http://www.beckhoff.de>) sowie auf den BECKHOFF Produkt-CDs bereit. Spezielle I/O-Parameter, die nicht vom DeviceNet®-Standard abgedeckt werden, können über die KS2000-Software (serielle Verbindung) oder über azyklische Explicit-Messages eingestellt werden.

Diagnose

Die umfangreichen Diagnosefunktionen der BECKHOFF DeviceNet®-Geräte ermöglichen die schnelle Fehlerlokalisierung. Die Diagnosemeldungen werden über den Bus übertragen und beim Master zusammengefasst. Der Status der Netzwerkverbindung, der Gerätestatus, der Status der Ein- und Ausgänge sowie der Spannungsversorgung werden mit LEDs dargestellt.

Übertragungsraten

Drei Übertragungsraten von 125 kBaud bis 500 kBaud stehen für unterschiedliche Buslängen zur Verfügung. Durch die effektive Nutzung der Busbandbreite erreicht DeviceNet® kurze Systemreaktionszeiten bei vergleichsweise niedrigen Datenraten.

Topologie

DeviceNet® basiert auf einer linienförmigen Topologie. Die Anzahl der Teilnehmer pro Netz ist dabei von DeviceNet® logisch auf 64 begrenzt, physikalisch erlaubt die aktuelle Treiber-Generation bis zu 64 Knoten in einem Netzsegment. Die bei einer bestimmten Datenrate maximal mögliche Netzausdehnung ist durch die auf dem Busmedium erforderliche Signallaufzeit begrenzt. Bei 500 kBaud ist z. B. eine Netzausdehnung von 100 m, bei 125 kBaud eine Netzausdehnung von 500 m möglich. Bei niedrigen Datenraten kann die Netzausdehnung durch den Einsatz von Repeatern erhöht werden, diese ermöglichen auch den Aufbau von Baumstrukturen.

Buszugriffsverfahren

CAN arbeitet nach dem Verfahren Carrier Sense Multiple Access (CSMA), d.h. jeder Teilnehmer ist bezüglich des Buszugriffs gleichberechtigt und kann auf den Bus zugreifen, sobald dieser frei ist (Multi-Master-Buszugriff). Der Nachrichtenaustausch ist dabei nicht Teilnehmerbezogen sondern Nachrichtenbezogen. Das bedeutet, dass jede Nachricht mit einem priorisierten Identifier eindeutig gekennzeichnet ist. Damit beim Verschicken der Nachrichten verschiedener Teilnehmer keine Kollisionen auf dem Bus entstehen, wird beim Start der Datenübertragung eine bitweise Busarbitrierung durchgeführt. Die Busarbitrierung vergibt die Busbandbreite an die Nachrichten in der Reihenfolge ihrer Priorität, am Ende der Arbitrierungsphase belegt jeweils nur ein Busteilnehmer den Bus, Kollisionen werden vermieden und die Bandbreite wird optimal genutzt.

Konfiguration und Parametrierung

Mit dem TwinCAT System Manager können alle DeviceNet® Parameter komfortabel eingestellt werden. Für die Parametrierung der BECKHOFF DeviceNet®-Geräte mit Konfigurationstools dritter Hersteller steht Ihnen auf der BECKHOFF Website (<http://www.beckhoff.de>) ein eds-File (electronic data sheet) zur Verfügung.

5.2 Explizite Nachrichten

Programmbeispiel „ExplMessageEditor“:  <https://infosys.beckhoff.com/content/1031/el6752/Resources/5979571979.zip>

Mit den folgenden ADS – Befehlen können über die EL6752 Explizit Messages genutzt werden.

```
GET_ATTRIBUTE_SINGLE via ADSRead Data Transfer
SET_ATTRIBUTE_SINGLE via ADSWrite Data Transfer
COMMON SERVICE via ADSReadWrite Data Transfer
```

Für die ADS NetID und den Port sind die Werte aus dem Systemmanager einzusetzen.

```
(*
GET_ATTRIBUTE_SINGLE via ADSRead Data Transfer

IDXGRP: Index GroupNumber = Object Class
IDXOFFS: Index OffsetNumber = (Object Instance *. 0x100) + Attribute Id
LEN: Read Data Lengths in Bytes
DESTADDR: Address of DataBuffer to read with the Get-Attribute Single Service
*)
```

```
fbADSRead(
NETID:= ADSNetId,
PORT:= ADSPort,
IDXGRP:= IGrp_ADSRead,
IDXOFFS:= IOff_ADSRead,
LEN:= ADSReadLen,
DESTADDR:= ADR(GetAttributeData[0]),
READ:= ADSReadCommand,
TMOUT:= T#5s,
BUSY=> ADSReadBusy,
ERR=> ADSReadErr,
ERRID=> ADSReadErrID);
(*)
```

```
COMMON SERVICE via ADSReadWrite Data Transfer

IDXGRP: Index GroupNumber = Object Class
IDXOFFS: Index OffsetNumber = (Object Instance *. 0x100) + Service Id
WRITELEN: Write Data Lengths in Bytes
READLEN: Read Data Lengths in Bytes
SRCADDR: Address of DataBuffer to write
DESTADDR: Address of DataBuffer to read
*)
```

```
fbADSReadWrite(
NETID:= ADSNetId,
PORT:= ADSPort,
IDXGRP:= Grp_ADSReadWrite,
IDXOFFS:= IOff_ADSReadWrite,
WRITELEN:= ADSReadWriteWriteLen,
READLEN:= ADSReadWriteReadLen,
SRCADDR:= ADR(CommonServiceWriteData[0]),
DESTADDR:= ADR(CommonServiceReadData[0]),
WRTRD:= ADSReadWriteCommand,
TMOUT:= T#5s,
BUSY=> ADSReadWriteBusy,
ERR=> ADSReadWriteErr,
ERRID=> ADSReadWriteErrID);
```

und

```
(*
SET_ATTRIBUTE_SINGLE via ADSWrite Data Transfer
IDXGRP: Index GroupNumber = Object Class
IDXOFFS: Index OffsetNumber = (Object Instance *. 0x100) + Attribute Id
LEN: Write Data Lengths in Bytes
SRCADDR: Address of DataBuffer to write with the Set-Attribute Single Service
*)
```

```
fbADSWrite(
NETID:= ADSNetId,
PORT:= ADSPort,
IDXGRP:= IGrp_ADSWrite,
```

```

IDXOFFS:= IOFF_ADSWrite,
LEN:= ADSWriteLen,
SRCADDR:= ADR(SetAttributeWriteData[0]),
WRITE:= ADSWriteCommand,
TMOUT:= T#5s,
BUSY=> ADSWriteBusy,
ERR=> ADSWriteErr,
ERRID=> ADSWriteErrID;

```

Solution Explorer

Search Solution Explorer (Ctrl+U)

- Solution 'explicitTC3' (1 project)
 - explicitTC3
 - SYSTEM
 - MOTION
 - PLC
 - SAFETY
 - C++
 - I/O
 - Devices
 - Device 1 (EtherCAT)
 - Image
 - Image-Info
 - SyncUnits
 - Inputs
 - Outputs
 - InfoData
 - Term 1 (EK1200)
 - Term 3 (EK1100)
 - Device 2 (EL6752)
 - Image
 - Inputs
 - Box 1 (BK5200)
 - Box 2 (BK5210)
 - Box 8 (BK5220)
 - Inputs
 - Outputs
 - Term 10 (KL1104)
 - Term 11 (KL2114)
 - Term 12 (KL3002)
 - Term 13 (KL3062)
 - Term 14 (KL4002)
 - End Term (KL9010)
 - Box 16 (BK5210)
 - Box 32 (BK5200)
 - Mappings

explicitTC3

General BK52x0/IX-B52x Startup Attributes ADS Parameter Diag

ADS Address (acyclic services): NetId: 10.3.32.22.3.1 Port: 4104 (0x1008)

ADS-Router on Box

☐ Enable Router

Net-Id:

Remote Name:

Online-Access

Object Class: Service Id:

Object Instance: Attribute Id:

Read-Length:

Read-Data:

Write-Data:

Number	Terminal Name	Online	Type	In Size	Out Siz
1	Term 10 (KL1104)		KL1104 (33...	0.4	0.0
	Channel 1				
	Input	0			
	Channel 2				
	Input	0			
	Channel 3				
	Input	0			
	Channel 4				
	Input	0			
2	Term 11 (KL2114)		KL2114 (33...	0.0	0.4
	Channel 1				
	Output	0			

Abb. 21: ADS NetID und Port aus Systemmanager einsetzen

6 Parametrierung und Inbetriebnahme

6.1 CoE-Interface

Allgemeine Beschreibung

Das CoE-Interface (CAN application protocol over EtherCAT Interface) ist die Parameterverwaltung für EtherCAT-Geräte. EtherCAT-Slaves oder auch der EtherCAT-Master verwalten darin feste (ReadOnly) oder veränderliche Parameter, die sie zum Betrieb, Diagnose oder Inbetriebnahme benötigen.

CoE-Parameter sind in einer Tabellen-Hierarchie angeordnet und prinzipiell dem Anwender über den Feldbus zugänglich. Der EtherCAT-Master (TwinCAT System Manager) kann über EtherCAT auf die lokalen CoE-Verzeichnisse der Slaves zugreifen und je nach Eigenschaften lesend oder schreibend einwirken.

Es sind verschiedene Typen für CoE-Datentypen möglich wie String (Text), Integer-Zahlen, Bool'sche Werte oder größere Byte-Felder. Damit lassen sich ganz verschiedene Eigenschaften beschreiben. Beispiele für solche Parameter sind Herstellerkennung, Seriennummer, Prozessdateneinstellungen, Gerätenamen, Abgleichwerte für analoge Messungen oder Passwörter.

Die Ordnung erfolgt in zwei Ebenen über hexadezimale Nummerierung:
Zuerst wird der (Haupt)Index genannt, dann der Subindex.

Die Wertebereiche sind:

- Index: 0x0000...0xFFFF (0...65535_{dez})
- Subindex: 0x00...0xFF (0...255_{dez})

Üblicherweise wird ein so lokalisierter Parameter geschrieben als 0x8010:07 mit voranstehendem „0x“ als Kennzeichen des hexadezimalen Zahlenraumes und Doppelpunkt zwischen Index und Subindex.

Die für den EtherCAT-Feldbusanwender wichtigen Bereiche sind

- 0x1000: Hier sind feste Identitätsinformationen zum Gerät hinterlegt wie Name, Hersteller, Seriennummer etc. Außerdem liegen hier Angaben über die aktuellen und verfügbaren Prozessdatenkonstellationen.
- 0x8000: Hier sind die für den Betrieb erforderlichen funktionsrelevanten Parameter für alle Kanäle zugänglich wie Filtereinstellung oder Ausgabefrequenz.

Weitere wichtige Bereiche sind:

- 0x4000: Hier befinden sich bei manchen EtherCAT-Geräten die Kanalparameter. Historisch war dies der erste Parameterbereich, bevor der 0x8000 Bereich eingeführt wurde. EtherCAT-Geräte, die früher mit Parametern in 0x4000 ausgerüstet wurden und auf 0x8000 umgestellt wurden, unterstützen aus Kompatibilitätsgründen beide Bereiche und spiegeln intern.
- 0x6000: Hier liegen die Eingangs-PDO („Eingänge“ aus Sicht des EtherCAT-Masters)
- 0x7000: Hier liegen die Ausgangs-PDO („Ausgänge“ aus Sicht des EtherCAT-Masters)



Verfügbarkeit

Nicht jedes EtherCAT-Gerät muss über ein CoE-Verzeichnis verfügen. Einfache I/O-Module ohne eigenen Prozessor verfügen in der Regel über keine veränderlichen Parameter und haben deshalb auch kein CoE-Verzeichnis.

Wenn ein Gerät über ein CoE-Verzeichnis verfügt, stellt sich dies im TwinCAT System Manager als ein eigener Karteireiter mit der Auflistung der Elemente dar:

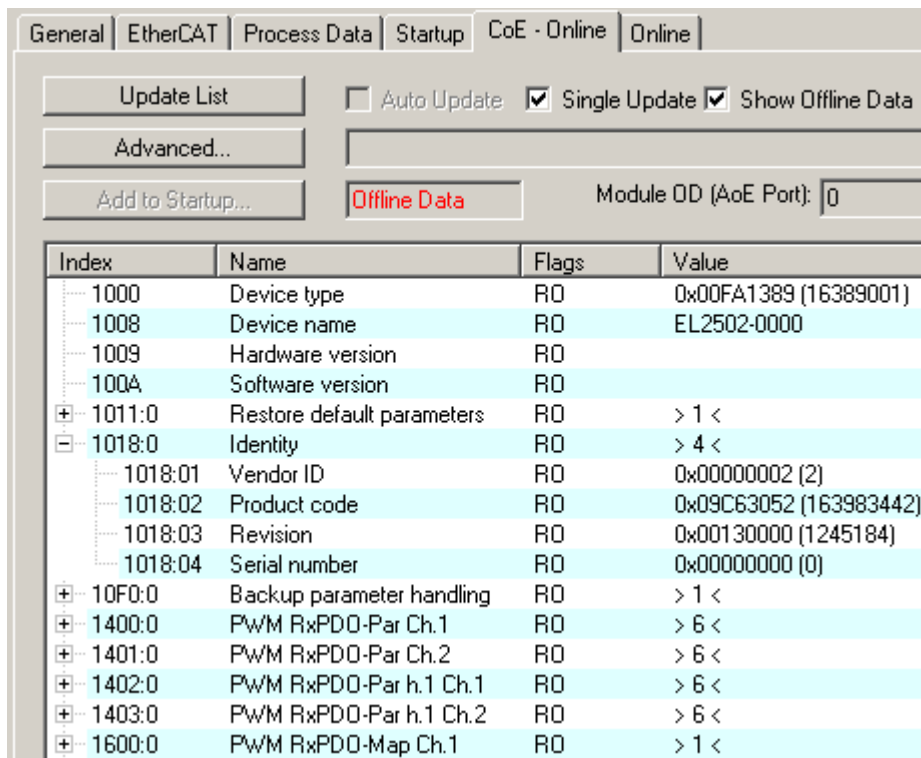


Abb. 22: Karteireiter „CoE-Online“

In der Abbildung „Karteireiter ‚CoE-Online‘“ sind die im Gerät „EL2502“ verfügbaren CoE-Objekte von 0x1000 bis 0x1600 zu sehen, die Subindizes von 0x1018 sind aufgeklappt.

HINWEIS

Veränderungen im CoE-Verzeichnis (CAN over EtherCAT-Verzeichnis), Programmzugriff

Beachten Sie bei Verwendung/Manipulation der CoE-Parameter die allgemeinen CoE-Hinweise im Kapitel „CoE-Interface“ der EtherCAT-System-Dokumentation:

- Startup-Liste führen für den Austauschfall,
- Unterscheidung zwischen Online/Offline Dictionary,
- Vorhandensein aktueller XML-Beschreibung (Download von der [Beckhoff Website](#)),
- "CoE-Reload" zum Zurücksetzen der Veränderungen
- Programmzugriff im Betrieb über die PLC (s. [TwinCAT 3 | PLC-Bibliothek: „Tc2_EtherCAT“](#) und [Beispielprogramm R/W CoE](#))

Datenerhaltung und Funktion „NoCoeStorage“

Einige, insbesondere die vorgesehenen Einstellungsparameter des Slaves, sind veränderlich und beschreibbar,

- über den System Manager (siehe Abb. „Karteireiter ‚CoE-Online‘“) durch Anklicken. Dies bietet sich bei der Inbetriebnahme der Anlage bzw. Slaves an. Klicken Sie auf die entsprechende Zeile des zu parametrierenden Indizes und geben Sie einen entsprechenden Wert im „SetValue“-Dialog ein.
- aus der Steuerung bzw. PLC über ADS z. B. durch die Bausteine aus der TcEtherCAT.lib Bibliothek. Dies wird für Änderungen während der Anlagenlaufzeit empfohlen oder wenn kein System Manager bzw. Bedienpersonal zur Verfügung steht.

i Datenerhaltung

Werden online auf dem Slave CoE-Parameter geändert, wird dies in Beckhoff-Geräten üblicherweise ausfallsicher im Gerät (EEPROM) gespeichert. D. h. nach einem Neustart (Re Power) sind die veränderten CoE-Parameter immer noch erhalten. Andere Hersteller können dies anders handhaben.

Ein EEPROM unterliegt in Bezug auf Schreibvorgänge einer begrenzten Lebensdauer. Ab typischerweise 100.000 Schreibvorgängen kann eventuell nicht mehr sichergestellt werden, dass neue (veränderte) Daten sicher gespeichert werden oder noch auslesbar sind. Dies ist für die normale Inbetriebnahme ohne Belang. Werden allerdings zur Maschinenlaufzeit fortlaufend CoE-Parameter über ADS verändert, kann die Lebensdauergrenze des EEPROMs durchaus erreicht werden.

Es ist von der FW-Version abhängig, ob die Funktion NoCoeStorage unterstützt wird, die das Abspeichern veränderter CoE-Werte unterdrückt.

Ob das auf das jeweilige Gerät zutrifft, ist den technischen Daten der entsprechenden Dokumentation zu entnehmen.

- Wird diese unterstützt: Die Funktion ist per einmaligem Eintrag des Codeworts 0x12345678 im CoE-Index 0xF008 zu aktivieren. Die Funktion ist solange aktiv, wie das Codewort unverändert bleibt. Nach dem Einschalten des Gerätes ist sie nicht aktiv. Veränderte CoE-Werte werden dann nicht im EEPROM abgespeichert, sie können somit beliebig oft verändert werden.
- Wird diese nicht unterstützt: Eine fortlaufende Änderung von CoE-Werten ist angesichts der o.a. Lebensdauergrenze nicht zulässig.

i Startup-Liste

Veränderungen im lokalen CoE-Verzeichnis der Klemme gehen im Austauschfall mit der alten Klemme verloren. Wird im Austauschfall eine neue Klemme mit Beckhoff Werkseinstellungen eingesetzt, bringt diese die Standardeinstellungen mit. Es ist deshalb empfehlenswert, alle Veränderungen im CoE-Verzeichnis eines EtherCAT-Slaves in der Startup-Liste des Slaves zu verankern, die bei jedem Start des EtherCAT-Feldbus abgearbeitet wird. So wird auch im Austauschfall ein neuer EtherCAT-Slave automatisch mit den Vorgaben des Anwenders parametrierung.

Wenn EtherCAT-Slaves verwendet werden, die lokal CoE-Werte nicht dauerhaft speichern können, ist zwingend die Startup-Liste zu verwenden.

Empfohlenes Vorgehen bei manueller Veränderung von CoE-Parametern

- Gewünschte Änderung im System Manager vornehmen (Werte werden lokal im EtherCAT-Slave gespeichert).
- Wenn der Wert dauerhaft Anwendung finden soll, einen entsprechenden Eintrag in der Startup-Liste vornehmen. Die Reihenfolge der Startup-Einträge ist dabei i.d.R. nicht relevant.

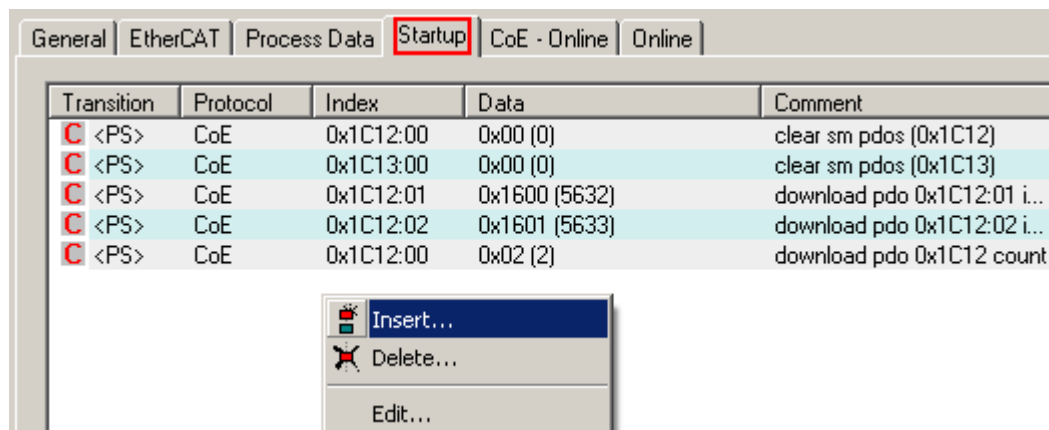


Abb. 23: Startup-Liste im TwinCAT System Manager

In der Startup-Liste können bereits Werte enthalten sein, die vom System Manager nach den Angaben der ESI dort angelegt werden. Zusätzliche anwendungsspezifische Einträge können ebenfalls angelegt werden.

Online- / Offline Verzeichnis

Im Rahmen der Arbeit mit dem TwinCAT System Manager ist zu differenzieren, ob das EtherCAT-Gerät gegenwärtig „verfügbar“ ist, also angeschaltet und über EtherCAT verbunden – somit **online** – oder ob eine Konfiguration **offline** erstellt wird, ohne dass Slaves angeschlossen sind.

In beiden Fällen ist ein CoE-Verzeichnis nach Abb. „Karteireiter ‚CoE-Online‘“ zu sehen, die Konnektivität wird allerdings als offline oder online angezeigt.

- Wenn der Slave offline ist,
 - wird das Offline-Verzeichnis aus der ESI-Datei angezeigt; Änderungen sind hier nicht sinnvoll bzw. möglich.
 - wird in der Identität der konfigurierte Stand angezeigt.
 - wird kein Firmware- oder Hardware-Stand angezeigt, da dies Eigenschaften des realen Gerätes sind.
 - ist ein rotes **Offline Data** zu sehen.

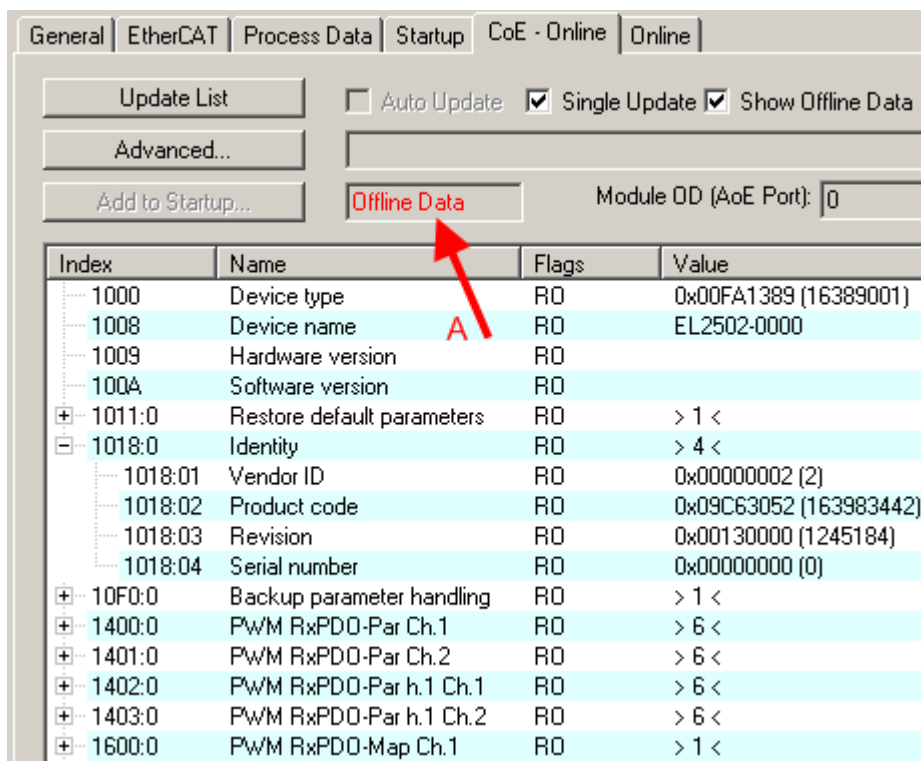


Abb. 24: Offline-Verzeichnis

- Wenn der Slave online ist,
 - wird das reale, aktuelle Verzeichnis des Slaves ausgelesen; dies kann je nach Größe und Zykluszeit einige Sekunden dauern.
 - wird die tatsächliche Identität angezeigt.
 - wird der Firmware- und Hardware-Stand des Gerätes im CoE angezeigt.
 - ist ein grünes **Online Data** zu sehen.

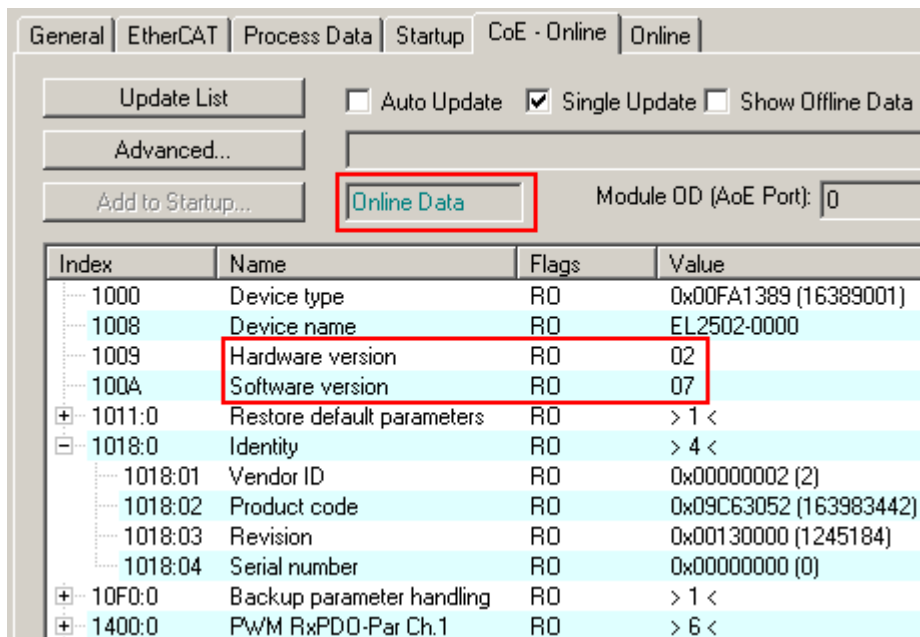


Abb. 25: Online-Verzeichnis

Kanalweise Ordnung

Das CoE-Verzeichnis ist in EtherCAT-Geräten angesiedelt, die meist mehrere funktional gleichwertige Kanäle umfassen; z. B. hat eine vierkanalige Analogeingangsklemme auch vier logische Kanäle und damit vier gleiche Sätze an Parameterdaten für die Kanäle. Um in den Dokumentationen nicht jeden Kanal auflisten zu müssen, wird gerne der Platzhalter „n“ für die einzelnen Kanalnummern verwendet.

Im CoE-System sind für die Menge aller Parameter eines Kanals eigentlich immer 16 Indizes mit jeweils 255 Subindizes ausreichend. Deshalb ist die kanalweise Ordnung in 16_{dez} bzw. 10_{hex} -Schritten eingerichtet. Am Beispiel des Parameterbereichs 0x8000 sieht man dies deutlich:

- Kanal 0: Parameterbereich 0x8000:00 ... 0x800F:255
- Kanal 1: Parameterbereich 0x8010:00 ... 0x801F:255
- Kanal 2: Parameterbereich 0x8020:00 ... 0x802F:255
- ...

Allgemein wird dies geschrieben als 0x80n0.

Ausführliche Hinweise zum CoE-Interface finden Sie in der [EtherCAT-Systemdokumentation](#) auf der Beckhoff Website.

BackUp-Objekte und Checksumme 0x10F0:01

In Bezug auf den Informationserhalt sind für das CoE-Parameter-/Objektverzeichnis u. a. folgende Objekttypen definiert, die aber nicht alle zugleich in jedem Gerät vorkommen müssen:

- Vendor-Objekte
 - Sind persistent (versorgungsausfallsicher) im Gerät gespeichert.
 - Technisch mit der Eigenschaft ReadWrite (RW).
 - Sie sind nur unter Kenntnis des jeweiligen Herstellerpassworts änderbar/löschbar.
 - Sie werden für herstellerbezogene Justage- oder Identitätsdaten verwendet.
- BackUp-Objekte
 - Das sind Objekte die persistent im Gerät gespeichert werden, auch nach Änderung.
 - Technisch mit der Eigenschaft ReadWrite (RW).
 - Sie sind jederzeit durch CoE-Zugriff aus dem EtherCAT-Master änderbar/löschbar

- Insbesondere werden sie durch „Restore Default Parameters“ (siehe Kapitel „Wiederherstellen des Auslieferungszustandes“) auf den Default-/Initialzustand zurückgesetzt, wie er in der Firmware unveränderlich gespeichert ist. Da dieses Zurücksetzen auf einen ehemaligen Wert wie die Wiederherstellung eines Backups erscheint, werden sie „BackUp-Objekte“ genannt.
- Sie werden für reguläre Funktionsparameter des Geräts verwendet, die das Verhalten festlegen.
- BackUp-Objekte mit Schreibschutzmöglichkeit.
 - Es gilt das Gleiche wie für die Backup-Objekte.
 - Zusätzlich kann der Anwender für diese Objekte einen Schreibschutz per Codewort in xF009 aktivieren und so die versehentliche Veränderung blockieren. Details dazu finden sich in den Gerätedokumentationen der Geräte, die diese Objekte enthalten.
- Volatile Objekte
 - Das sind Objekte, die nicht persistent im Gerät gespeichert werden.
 - Sie werden zur Anzeige von internen Informationen (Prozessdaten, Zustände, Temperaturen ...) verwendet und sind als ReadOnly (RO) oder als Funktionsparameter (ReadWrite) verfügbar. Letztere müssen jedoch bei jedem PowerOn vom EtherCAT Master beschrieben werden, wenn sie einen anderen Wert als den Default-Wert haben sollen.

Über den aktuellen Zustand der sogenannten BackUp-Objekte zeigt das Gerät eine 16-Bit-CRC im 32-Bit-Objekt 0x10F0:01 CheckSum, Subindex 01 von Backup Parameter Handling an:

10F0:0	Backup parameter handling	RO	> 1 <
10F0:01	Checksum	RO	0x00003C62 (15458)

Abb. 26: CoE Index 10F0

Wird ein BackUp-Objekt verändert, errechnet die Firmware entsprechend eine neue Checksumme. Diese kann verwendet werden, um Änderungen der BackUp-Objekte festzustellen.

Hinweis: Der Initialwert kann sich je nach Firmware-Stand ändern, wenn durch Funktionserweiterungen Objekte in den Erfassungsbereich der CRC dazukommen.

6.2 Allgemeine Hinweise zur Watchdog-Einstellung

Die EtherCAT-Klemmen sind mit einer Sicherungseinrichtung (Watchdog) ausgestattet, die z. B. bei unterbrochenem Prozessdatenverkehr nach einer voreinstellbaren Zeit die Ausgänge (sofern vorhanden) in einen gegebenenfalls vorgebbaren Zustand schaltet, in Abhängigkeit von Gerät und Einstellung z. B. auf FALSE (aus) oder einen Ausgabewert.

Der EtherCAT Slave Controller verfügt dazu über zwei Watchdogs:

- Sync Manager (SM)-Watchdog (default: 100 ms)
- Process-Data (PDI)-Watchdog (default: 100 ms)

Deren Zeiten werden in TwinCAT wie folgt einzeln parametriert:

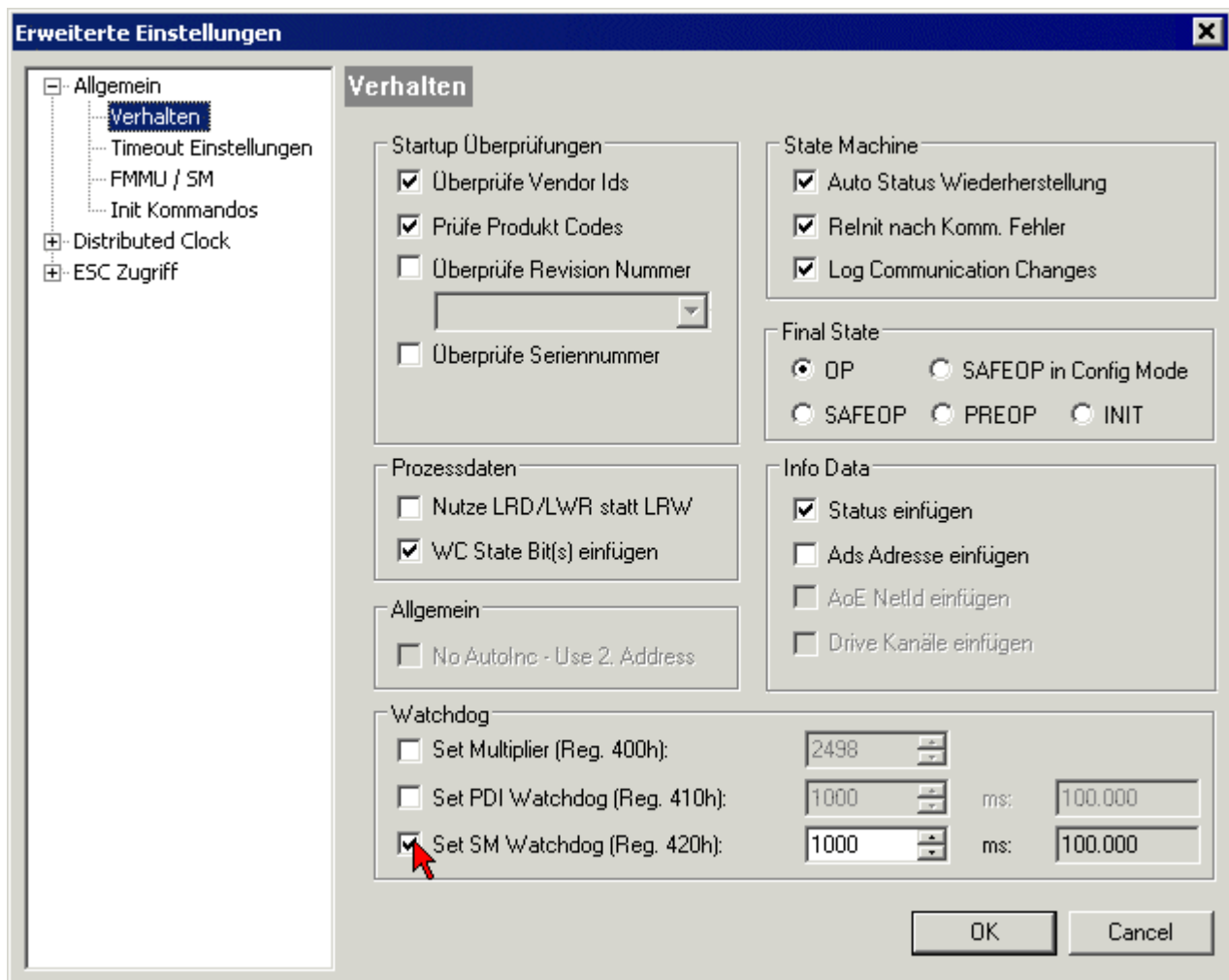


Abb. 27: Karteireiter EtherCAT -> Erweiterte Einstellungen -> Verhalten -> Watchdog

Anmerkungen:

- Das Multiplier-Register 400h (hexadezimal, also 0x0400), ist für beide Watchdogs gültig.
- Jeder Watchdog hat seine eigene Timer-Einstellung 410h bzw. 420h, die zusammen mit dem Multiplier eine resultierende Zeit ergibt.
- Wichtig: die Multiplier-/Timer-Einstellung wird nur dann beim EtherCAT-Start in den Slave geladen, wenn die Checkbox davor aktiviert ist. Ist diese nicht aktiviert, wird nichts herunter geladen und die im ESC befindliche Einstellung bleibt unverändert.
- Die heruntergeladenen Werte können in den ESC-Registern 400h, 410h und 420h eingesehen werden: ESC Zugriff -> Speicher (ESC Access -> Memory).

SM-Watchdog (SyncManager-Watchdog)

Der SyncManager-Watchdog wird bei jeder erfolgreichen EtherCAT-Prozessdatenkommunikation mit der Klemme zurückgesetzt. Findet z. B. durch eine Leitungsunterbrechung länger als die eingestellte und aktivierte SM-Watchdog-Zeit keine EtherCAT-Prozessdatenkommunikation mit der Klemme statt, löst der Watchdog aus. Der Status der Klemme (in der Regel OP) bleibt davon unberührt. Der Watchdog wird erst wieder durch einen erfolgreichen EtherCAT-Prozessdatenzugriff zurückgesetzt.

Der SyncManager-Watchdog ist also eine Überwachung auf korrekte und rechtzeitige Prozessdatenkommunikation zwischen Master und ESC, die allein auf EtherCAT-Ebene abläuft.

Die maximal mögliche Watchdog-Zeit ist geräteabhängig. Beispielsweise beträgt sie bei „einfachen“ EtherCAT-Slaves (ohne Firmware) mit Watchdog-Ausführung im ESC in der Regel bis zu 170 Sekunden. Bei komplexen EtherCAT-Slaves (mit Firmware) wird die SM-Watchdog-Funktion in der Regel zwar über Register 400h/420h parametriert, aber vom Microcontroller (µC) ausgeführt und kann deutlich darunter liegen. Außerdem kann die Ausführung dann einer gewissen Zeitunsicherheit unterliegen. Da der TwinCAT-Dialog ggf. Eingaben bis 65535 zulässt, wird ein Test der gewünschten Watchdog-Zeit empfohlen.

PDI-Watchdog (Process Data Watchdog)

Findet länger als die eingestellte und aktivierte PDI-Watchdog-Zeit keine PDI (Process Data Interface)-Kommunikation mit dem ESC statt, löst dieser Watchdog aus.

PDI ist die interne Schnittstelle des ESC z. B. zu lokalen Prozessoren im EtherCAT-Slave. Mit dem PDI-Watchdog kann diese Kommunikation auf Ausfall überwacht werden.

Der PDI-Watchdog ist also eine Überwachung auf korrekte und rechtzeitige Prozessdatenkommunikation mit dem ESC, nun aber von der Applikationsseite aus betrachtet.

Berechnung

Watchdog-Zeit = $[1/25 \text{ MHz} * (\text{Watchdog-Multiplier} + 2)] * \text{SM/PDI Watchdog}$

Beispiel: Default-Einstellung Multiplier = 2498, SM-Watchdog = 1000 => 100 ms

Der Wert in „Watchdog-Multiplier + 2“ in der oberen Formel entspricht der Anzahl 40ns-Basisticks, die einen Watchdog-Tick darstellen.

⚠ VORSICHT

Ungewolltes Verhalten des Systems möglich!

Die Abschaltung des SM-Watchdog durch SM-Watchdog = 0 funktioniert erst in Klemmen ab Revision -0016. In vorherigen Versionen wird vom Einsatz dieser Betriebsart abgeraten.

⚠ VORSICHT

Beschädigung von Geräten und ungewolltes Verhalten des Systems möglich!

Bei aktiviertem SM-Watchdog und eingetragenen Wert 0 schaltet der Watchdog vollständig ab! Dies ist die Deaktivierung des Watchdogs! Gesetzte Ausgänge werden dann bei einer Kommunikationsunterbrechung NICHT in den sicheren Zustand gesetzt!

6.3 EtherCAT State Machine

Über die EtherCAT State Machine (ESM) wird der Zustand des EtherCAT-Slaves gesteuert. Je nach Zustand sind unterschiedliche Funktionen im EtherCAT-Slave zugänglich bzw. ausführbar. Insbesondere während des Hochlaufs des Slaves müssen in jedem State spezifische Kommandos vom EtherCAT-Master zum Gerät gesendet werden.

Es werden folgende Zustände unterschieden:

- Init
- Pre-Operational
- Safe-Operational
- Operational
- Bootstrap

Regulärer Zustand eines jeden EtherCAT-Slaves nach dem Hochlauf ist der Status Operational (OP).

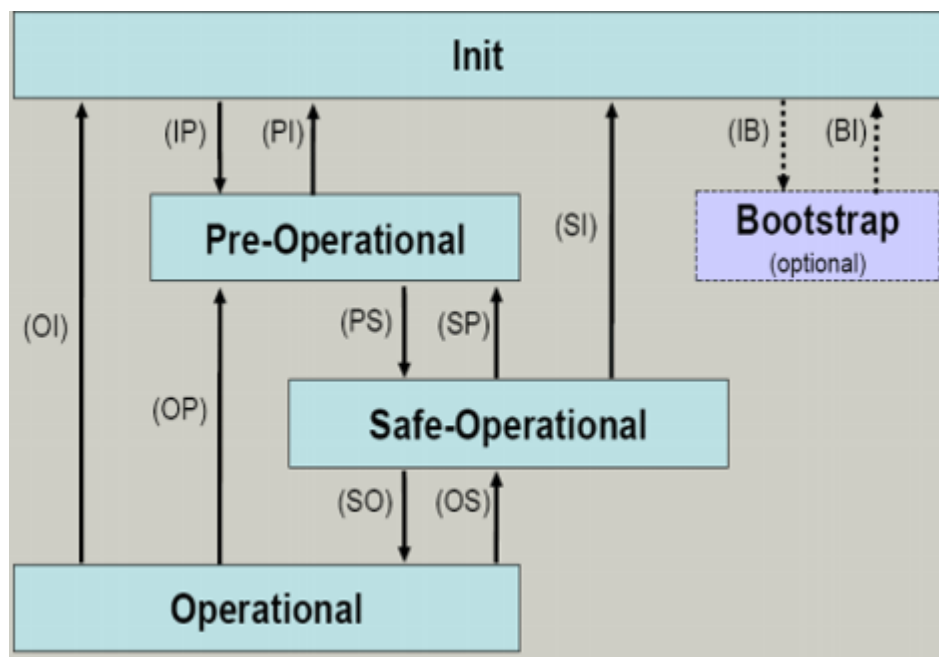


Abb. 28: Zustände der EtherCAT State Machine

Init

Nach dem Einschalten befindet sich der EtherCAT-Slave im Zustand *Init*. Dort ist weder Mailbox- noch Prozessdatenkommunikation möglich. Der EtherCAT-Master initialisiert die Sync-Manager-Kanäle 0 und 1 für die Mailbox-Kommunikation.

Pre-Operational (Pre-Op)

Beim Übergang von *Init* nach *Pre-Op* prüft der EtherCAT-Slave, ob die Mailbox korrekt initialisiert wurde.

Im Zustand *Pre-Op* ist Mailbox-Kommunikation aber keine Prozessdatenkommunikation möglich. Der EtherCAT-Master initialisiert die Sync-Manager-Kanäle für Prozessdaten (ab Sync-Manager-Kanal 2), die Kanäle der Fieldbus Memory Management Unit (FMMU) und, falls der Slave ein konfigurierbares Mapping unterstützt, das Mapping der Prozessdatenobjekte (PDOs) oder das Sync-Manager-PDO-Assigment. Weiterhin werden in diesem Zustand die Einstellungen für die Prozessdatenübertragung sowie ggf. noch klemmenspezifische Parameter übertragen, die von den Default-Einstellungen abweichen.

Safe-Operational (Safe-Op)

Beim Übergang von *Pre-Op* nach *Safe-Op* prüft der EtherCAT-Slave, ob die Sync-Manager-Kanäle für die Prozessdatenkommunikation sowie ggf. die Einstellungen für die Distributed Clocks korrekt sind. Bevor er den Zustandswechsel quittiert, kopiert der EtherCAT-Slave aktuelle Inputdaten in die entsprechenden Dual Port (DP)-RAM-Bereiche des ESC.

Im Zustand *Safe-Op* ist Mailbox- und Prozessdatenkommunikation möglich, allerdings hält der Slave seine Ausgänge im sicheren Zustand und gibt sie noch nicht aus. Die Inputdaten werden aber bereits zyklisch aktualisiert.



Ausgänge im SAFEOP

Die standardmäßig aktivierte Überwachung mittels Watchdog bringt die Ausgänge im ESC-Modul in Abhängigkeit von den Einstellungen im SAFEOP und OP in einen sicheren Zustand - je nach Gerät und Einstellung - z. B. auf AUS. Wird dies durch Deaktivieren der Überwachung unterbunden, können auch im Geräte-Zustand SAFEOP Ausgänge geschaltet werden bzw. gesetzt bleiben.

Operational (Op)

Bevor der EtherCAT-Master den EtherCAT-Slave von *Safe-Op* nach *Op* schaltet, muss er bereits gültige Outputdaten übertragen.

Im Zustand *Op* kopiert der Slave die Ausgangsdaten des Masters auf seine Ausgänge. Es ist Prozessdaten- und Mailboxkommunikation möglich.

Boot

Im Zustand *Boot* kann ein Update der Slave-Firmware vorgenommen werden. Der Zustand *Boot* ist nur über den Zustand *Init* zu erreichen.

Im Zustand *Boot* ist Mailbox-Kommunikation über das Protokoll File-Access over EtherCAT (FoE) möglich, aber keine andere Mailbox- und Prozessdatenkommunikation.

6.4 TwinCAT System Manager

Zur Konfiguration der EL6752 DeviceNet® Master-/Slave Klemme dient das TwinCAT System Manager Tool. Der System Manager stellt die Anzahl und Programme der TwinCAT SPS-Systeme, die Konfiguration der Achsregelung und die angeschlossenen E/A-Kanäle als Struktur dar und organisiert das Mapping des Datenverkehrs.



Abb. 29: TwinCAT System Manager Logo

Für Applikationen ohne TwinCAT SPS oder NC konfiguriert das TwinCAT System Manager Tool die Programmierschnittstellen für vielfältige Applikationsprogramme:

- ActiveX-Control (ADS-OCX) für z. B. Visual Basic, Visual C++, Delphi, etc
- DLL-Interface (ADS-DLL) für z. B. Visual C++ Projekte
- Script-Interface (ADS-Script DLL) für z. B. VBScript, JScript, etc.

System Manager – Eigenschaften

- Verbindung zwischen Server-Prozessabbildern und E/A-Kanälen bitweise
- Standard-Datenformate, z. B. Arrays und Strukturen
- Benutzerdefinierte Datenformate
- Fortlaufende Verbindung von Variablen
- Drag und Drop
- Import und Export auf allen Ebenen

Konfiguration mit dem TwinCAT System Manager

Im Folgenden werden das Vorgehen und die Konfigurationsmöglichkeiten im System Manager beschrieben:

[EL6752 - DeviceNet® Master Klemme \[► 49\]](#)

[EL6752-0010 - DeviceNet® Slave Klemme \[► 52\]](#)

EL6752 - DeviceNet® Master Klemme

Gerät anfügen

Das Anfügen der Klemme in die E/A-Konfiguration kann entweder durch die "Geräte suchen"-Routine im TwinCat System Manager durchgeführt oder durch die manuelle Auswahl des "DeviceNet Master EL6752, EtherCAT" aus den möglichen DeviceNet®-Geräten eingefügt werden (Abb. *Anfügen des Gerätes „DeviceNet Master EL6752, EtherCAT“*). Durch Rechtsklick besteht die Auswahl aus folgendem Kontextmenü:

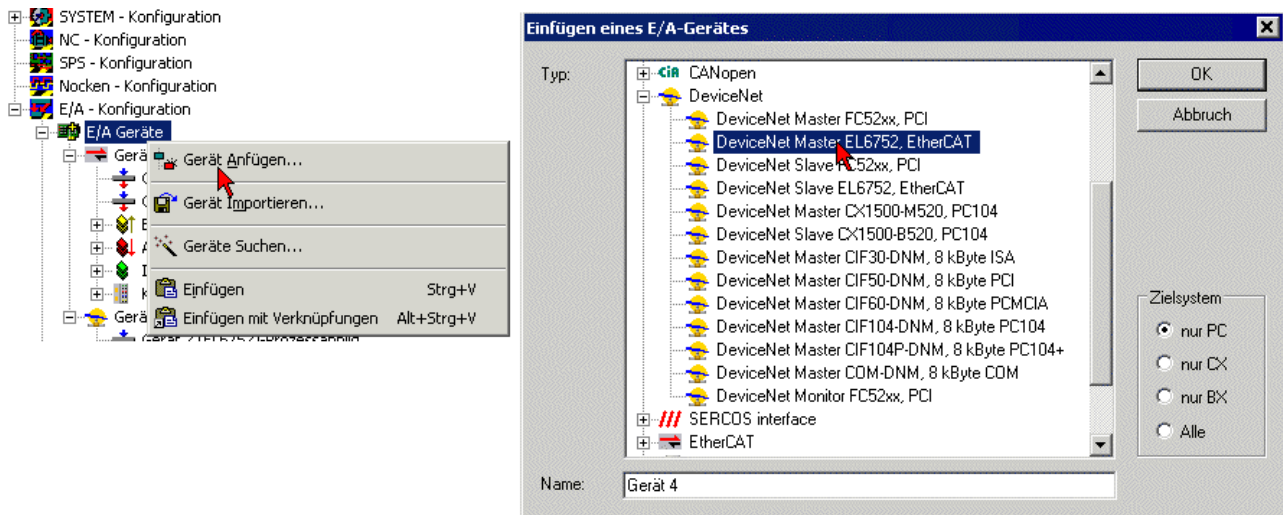


Abb. 30: Anfügen des Gerätes „DeviceNet Master EL6752, EtherCAT“

Karteireiter "EL6752"

Klicken Sie im TwinCAT Baum auf das "Gerät EL6752" und anschließend auf den EL6752-Reiter:

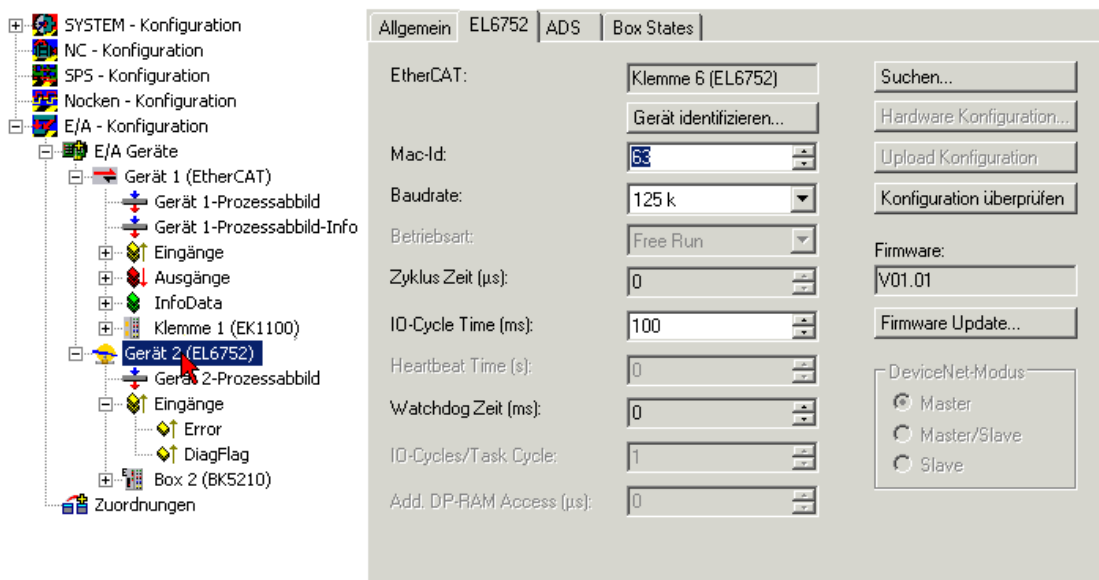


Abb. 31: Karteireiter „EL6752“

EtherCAT

Bezeichnung der Klemme im Klemmenverbund.

MAC-ID

Jedes DeviceNet® Gerät - einschließlich Master - benötigt eine eindeutige Stationsnummer die MAC-ID (Medium Access Identifier) - Wertebereich: 0...63.

Baudrate

Einstellung der Baudrate: 125 kBaud, 250 kBaud oder 500 kBaud.

Zykluszeit

Hier wird die Zykluszeit der zugehörigen höchstpriorigen Task angezeigt. Die Anzeige wird aktualisiert sobald das Mapping erzeugt wird.

IO-Cycle time

Einstellung der Zykluszeit für die E/A Verbindungen. Dieser Wert ist der Standardwert für neu eingefügte Boxen.

Watchdog time

Zeit bis zum Triggern des Watchdogs

Suchen...

Hierüber werden alle vorhandenen Kanäle der EL6752 gesucht, und es kann der gewünschte ausgewählt werden.

Konfiguration überprüfen

In Vorbereitung.

Firmware

Hier wird die aktuelle Firmware-Version der EL6752 angezeigt.

Firmware Update...

Hierüber kann die Firmware der EL6752 aktualisiert werden. Achtung: Das TwinCAT System muss hierzu gestoppt sein.

Karteireiter "ADS"

The screenshot shows the 'ADS' configuration tab. At the top are four tabs: 'Allgemein', 'EL6752', 'ADS' (selected), and 'Box States'. Below the tabs, there is a 'Use Port' checkbox which is checked. To its right is a 'Port No.' field containing '28674 (0x7002)' and an 'Ändern...' button. Below that is a 'NetId' field containing '172.16.6.49.3.1'. Further down is a 'Remote Name' field containing 'Gerät 2 (EL6752)'. At the bottom, there is an 'Add. NetIds' section with two empty input boxes and two buttons: 'Hinzufügen' and 'Löschen'.

Abb. 32: Karteireiter „ADS“

Die EL6752 ist ein ADS-Device mit einer eigenen Net-ID, die hier verändert werden kann. Alle ADS-Dienste (Diagnose, azyklische Kommunikation), die an die EL6752 gehen, müssen die Karte mittels dieser Net-ID adressieren.

Karteireiter "Box States"

The screenshot shows the 'Box States' configuration tab. At the top are four tabs: 'Allgemein', 'EL6752', 'ADS', and 'Box States' (selected). Below the tabs is a table with two columns: 'Mac-ID' and 'BoxState'. The first row of the table contains the value '1' under 'Mac-ID' and 'No error' under 'BoxState'. Below the table are two buttons: 'Refresh' and 'Reset Counter'.

Abb. 33: Karteireiter „Box States“

Hier wird eine Übersicht aller aktuellen Box-States angezeigt.

EL6752-0010 – DeviceNet® Slave Klemme

Zunächst in der Baumstruktur der Systemkonfiguration mittels rechtem Mausklick auf E/A Geräte und "Gerät anfügen" die Auswahlliste der unterstützten Feldbuskarten öffnen.

EL6752-0010 CANOpenSlave auswählen. TwinCAT sucht nach der Klemme und zeigt die gefundenen Speicheradressen bzw. Slots an. Entsprechende Adresse auswählen und bestätigen.

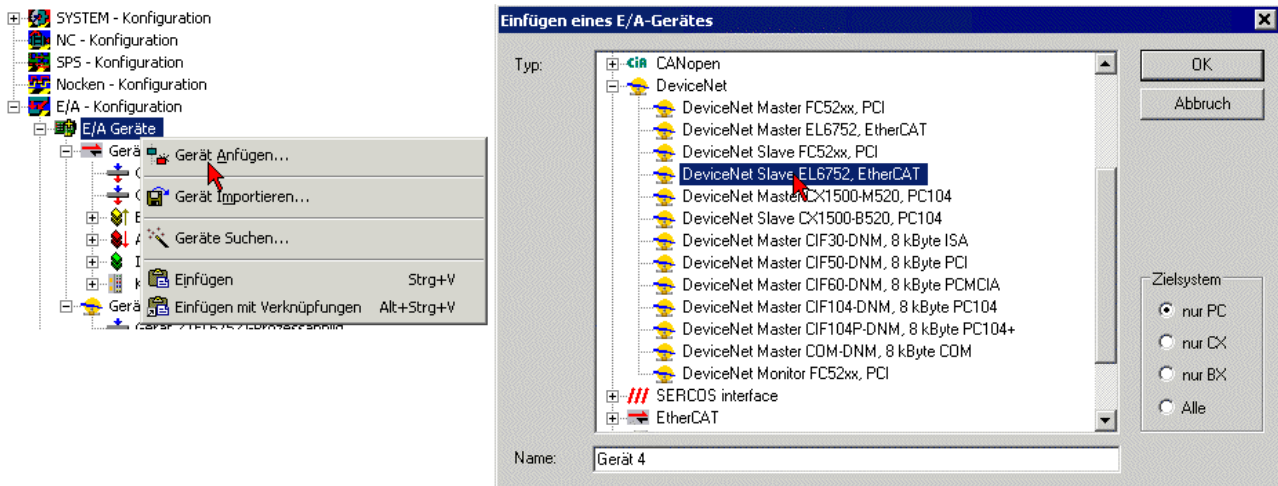


Abb. 34: Anfügen des Gerätes „DeviceNet Slave EL6752, EtherCAT“

Fügen Sie anschließend durch Rechtsklick bei "Gerät (EL6752-0010)" die Box für die EL6752-0010 ein:

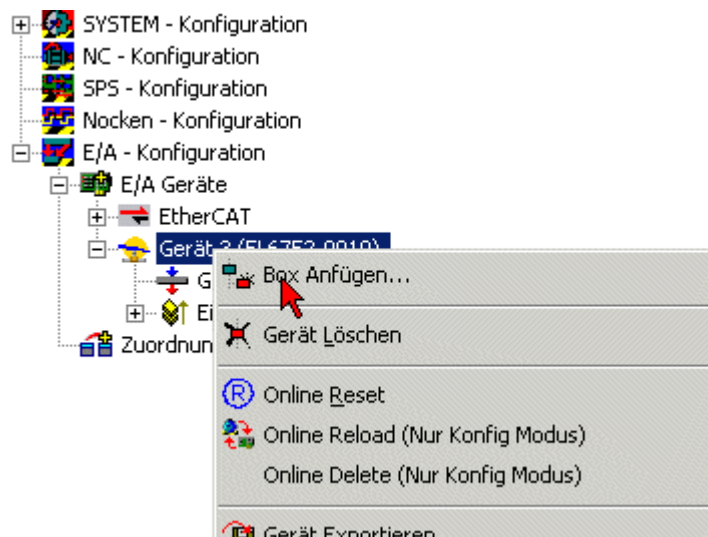


Abb. 35: Anfügen der Box „DeviceNet Slave EL6752, EtherCAT“

Bei Anwahl des E/A Gerätes für die EL6752-0010 in der Baumstruktur öffnet sich ein Dialog mit verschiedenen Konfigurationsmöglichkeiten:

Karteireiter "EL6752-0010"

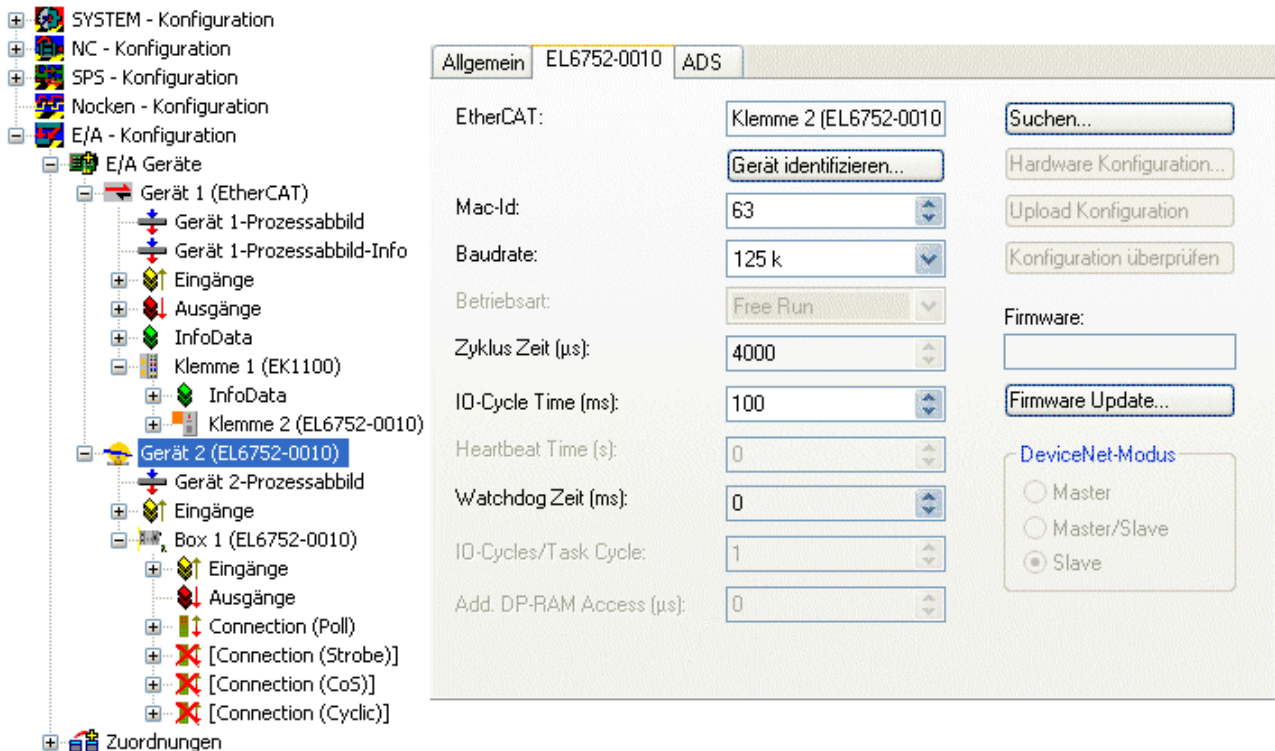


Abb. 36: Karteireiter „EL6752-0010“

EtherCAT

Bezeichnung der Klemme im Klemmenverbund.

MAC-ID

Jedes DeviceNet® Gerät benötigt eine eindeutige Stationsnummer die MAC-ID (Medium Access Identifier) - Wertebereich: 0...63.

Baudrate

Hier wird die Baudrate eingestellt.

Zyklus-Zeit

Hier wird die Zykluszeit der zugehörigen höchstpriorären Task angezeigt. Die Anzeige wird aktualisiert sobald das Mapping erzeugt wird. Die Netzwerkvariablen werden im Takt dieser Task aktualisiert.

Watchdog-Zeit

Zeit, bis der Watchdog triggert

Suchen...

Hierüber werden alle vorhandenen EL6752-0010 Kanäle gesucht, und es kann der gewünschte ausgewählt werden. Bei einer FC5102 erscheinen beide Kanäle A und B, die sich logisch wie zwei FC5101-Karten verhalten.

Firmware

Hier wird die aktuelle Firmware-Version der EL6752-0010 angezeigt.

Firmware Update...

Hierüber kann die Firmware der EL6752-0010 aktualisiert werden. Achtung: Das TwinCAT System muss hierzu gestoppt sein.

Karteireiter "ADS"

The screenshot shows the 'ADS' configuration tab. At the top, there are three tabs: 'Allgemein', 'EL6752-0010', and 'ADS'. The 'ADS' tab is active. Below the tabs, there is a checkbox 'Use Port' which is checked. To its right is a text field 'Port No.' containing '28674 (0x7002)' and a button 'Ändern...'. Below this is a text field 'NetId' containing '10.5.2.7.3.1'. To the right of 'NetId' is a text field 'Remote Name' containing 'Gerät 2 (EL6752-0010)'. Below these is a text field 'Add. NetIds' which is empty. To the right of this field are two buttons: 'Hinzufügen' and 'Löschen'.

Abb. 37: Karteireiter „ADS“

Die EL6752-0010 ist ein ADS-Device mit einer eigenen Net-ID, die hier verändert werden kann. Alle ADS-Dienste (Diagnose, azyklische Kommunikation), die an die EL6752-0010 gehen, müssen die Karte mittels dieser Net-ID adressieren. Zusätzliche ADS Net-IDs können eingetragen werden, um unterlagerte ADS Geräte (z. B. eine weitere Feldbus-Karte im gleichen PC) über die Karte anzusprechen.

Karteireiter "DPRAM (Online)"

Zu Diagnosezwecken kann lesend direkt auf das DPRAM der Karte zugegriffen werden.

Box EL6752-0010 Slave

Es wird automatisch eine Box "EL6752-0010 (DeviceNet Slave)" angelegt. Hier sind weitere Parameter einzustellen:

Karteireiter Box EL6752-0010:

The screenshot shows the 'Box EL6752-0010' configuration window. On the left, there is a tree view with 'Box 1 (EL6752-0010)' selected. Below it are several icons and labels: 'Eingänge', 'Ausgänge', 'Connection (Poll)', '[Connection (Strobe)]', '[Connection (CoS)]', and '[Connection (Cyclic)]'. The main area shows the 'Allgemein' tab. It contains a 'Name' field with 'Box 1 (EL6752-0010)', an 'Id' field with '3', a 'Typ' field which is empty, and a 'Kommentar' field with 'EDS for EL6752-0010'. At the bottom left is a 'Disabled' checkbox, and at the bottom right is a 'Symbole erzeugen' checkbox.

Abb. 38: Karteireiter "Allgemein", Box EL6752-0010

DeviceNet® IO-Betriebsarten

Die EL6752-0010 unterstützt die DeviceNet®-Betriebsarten zyklisches Polling, Change of State / Cyclic und Bit-Strobe. Die IO-Betriebsarten können entsprechend den DeviceNet®-Spezifikation selektiert werden.

Die DeviceNet® IO-Betriebsart zyklisches Polling wird per Default für die EL6752-0010 selektiert:

IO-Betriebsart	Eingangsdatenlänge / Byte	Ausgangsdatenlänge / Byte
Polling	0 - 255	0 - 255
Change of State	0 - 255	0 - 255
Cyclic	0 - 255	0 - 255
Bit-Srobe	1 Bit	0-8

Polling / Change of State (COS) / Cyclic

Die Betriebsart zyklisches Polling ist charakterisiert durch ein zyklisches Abfragen bzw. Pollen der IO-Daten durch den Master. Die Betriebsart Change of State ist charakterisiert durch das ereignisorientierte Versenden der IO-Daten. In der Betriebsart Cyclic werden die IO-Daten zyklisch nach den durch den Master konfigurierten Kommunikationsparametern versendet. Da die Kommunikationseinstellungen durch den Master vorgegeben werden sind hier keine weiteren Einstellungen mehr möglich. Mehr Informationen zu den Betriebsarten sind dem Kapitel DeviceNet® Kommunikation zu entnehmen. Die Einstellungen sind für die genannten Betriebsarten identisch.

Die Eingangs- und Ausgangsdatenlängen sind mit je 8 Byte vorinitialisiert:

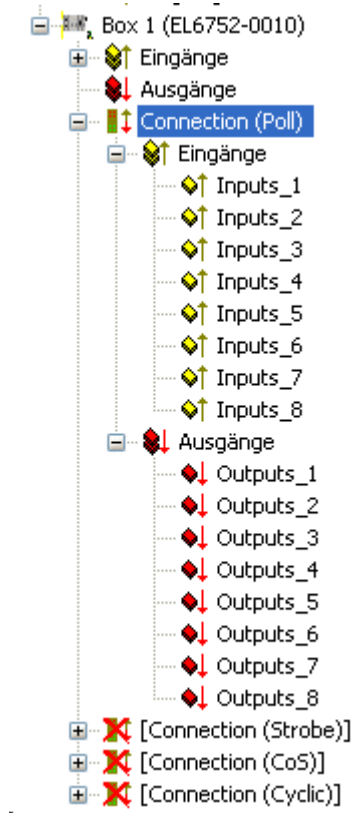


Abb. 39: Vorinitialisierte Ein- und Ausgangsdatenlängen bei Betriebsart Polling

Nach Bedarf und Applikation können weitere Eingangs- bzw. Ausgangsdaten durch Rechtsklick (Abb. Einfügen von weiteren Variablen) angefügt werden. Hierbei können beliebige Datentypen ausgewählt werden:

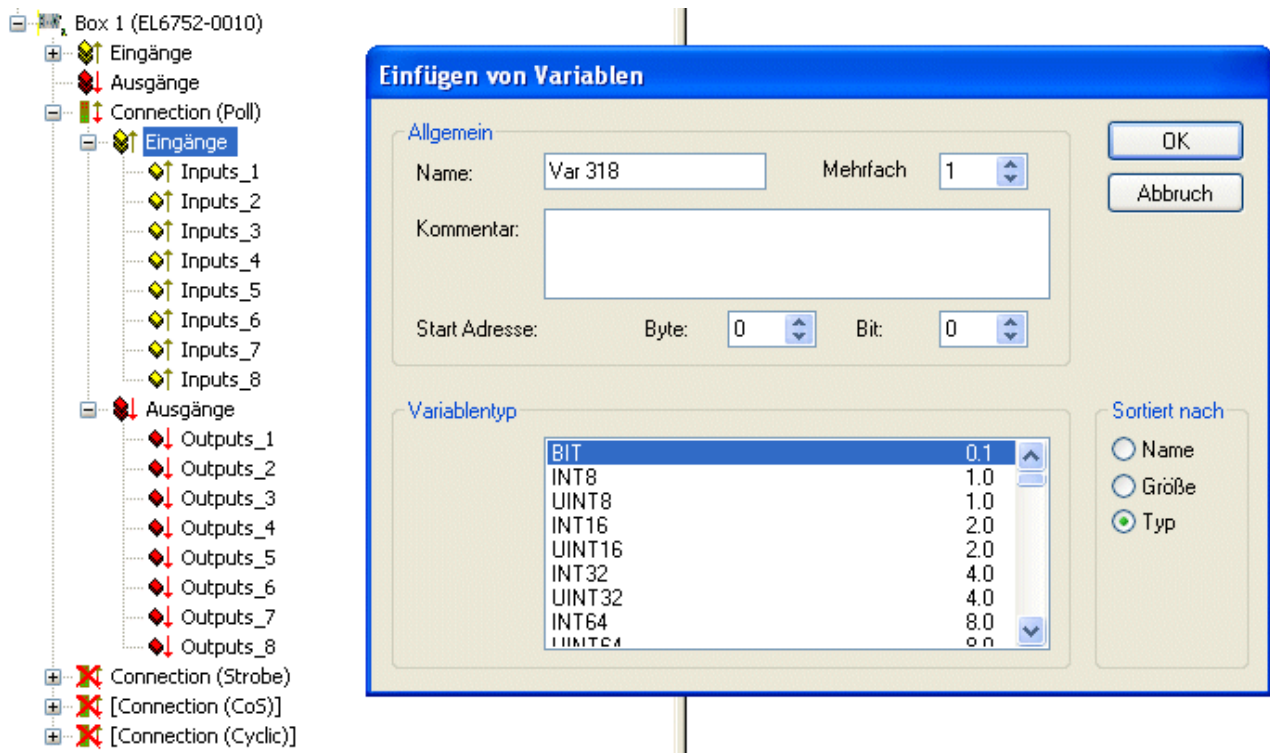


Abb. 40: Einfügen von weiteren Variablen

Die Datenlänge wird entsprechend der DeviceNet® Spezifikation in ein Byte-Stream umgerechnet und auf dem Karteireiter der entsprechenden Verbindung zur Anzeige gebracht:

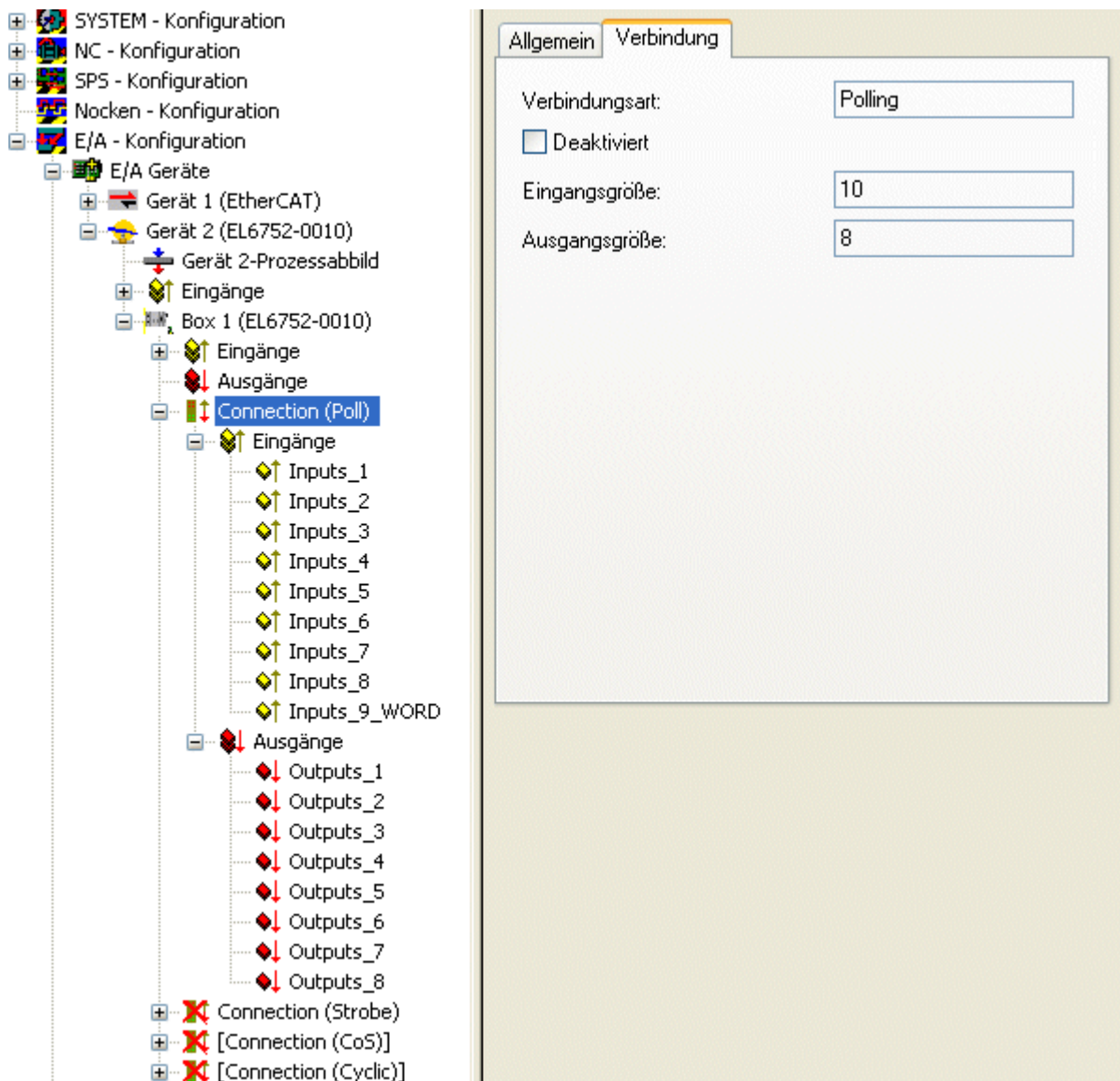


Abb. 41: Karteireiter „Verbindung“ mit Anzeige der Verbindungsart „Polling“ und der Ein- und Ausgangsgrößen

Maximale Ausgangs-Datenlänge

Die maximale Datenlänge darf 255 Bytes pro Datenrichtung betragen.

Die angezeigten Eingangs- und Ausgangsdatenlängen müssen beim zugehörigen DeviceNet®-Master entsprechend konfiguriert werden.

Bit Strobe

Die IO-Betriebsart Bit-Strobe beinhaltet ein 8-Byte Kommando des Master an die Slaves. Für jeden mögliche Adresse/MAC-ID (DeviceNet Adressraum: 64) ist entsprechend 1 Bit Nutzdaten reserviert. Die Antwort-Nachricht des Slaves darf maximal 8 Byte lang sein und wird unmittelbar nach Erhalt des Bit-Strobe Kommandos an den Master gesendet.

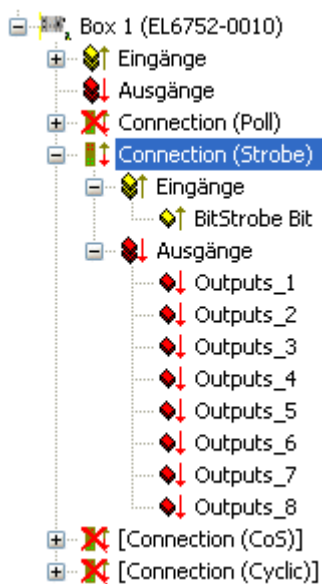


Abb. 42: Anzeige der Ausgangsgrößen in Verbindungsart „Bit Strobe“ im TwinCAT-Baum

● Maximale Ausgangs-Datenlänge

i Die maximale Ausgangs-Datenlänge darf 8 Bytes betragen. Die Eingangsdatenlänge ist fest vorgegeben.

Da die Kommunikationseinstellungen durch den Master vorgegeben werden sind hier keine weiteren Einstellungen mehr möglich.

6.5 Beckhoff DeviceNet® Buskoppler

Der Buskoppler BK52xx sowie die FeldbusBox IPxxx-B520 werden im **DeviceNet®** Bus eingesetzt. Nachfolgend werden die spezifischen Eigenschaften beschrieben, die sich von anderen Buskopplern bzw. Feldbus Box Modulen unterscheiden.

Typen	Beschreibung
BK5210	Economy Buskoppler
BK5220	Economy + Buskoppler
LC5200	Low-Cost Buskoppler
BK5250	Kompakt Buskoppler
BC5250	Kompakt Busklemmen Controller mit 48 kByte Programmspeicher
BX5200	BX Busklemmen Controller mit 256 kByte Programmspeicher
IPxxxx-B520	Feldbus Kompakt Box: DeviveNet Ein-/Ausgabebaugruppe in Schutzart IP67

Die Parametrierung findet mit folgenden Karteireitern statt:

- Karteireiter "BK52x0" [► 59]
- Karteireiter "Startup-Attributes" [► 61]
- Karteireiter "ADS" [► 62]
- Karteireiter "Parameter" [► 63]
- Karteireiter "Diag" [► 63]

Karteireiter "BK52x0"

Abb. 43: Karteireiter „BK52x0“

MAC-ID

Stellt die MAC-ID, d.h. die Geräteadresse des DeviceNet® Teilnehmers ein (zwischen 0 und 63). Dieser Wert muss mit dem am Buskoppler bzw. an der Kompakt Box eingestellten Wert übereinstimmen.

Cycle-Time

Stellt die Zyklus-Zeit der IO-Verbindung Polling und Bit-Strobe ein. Der Wert wird gemäß DeviceNet® Spezifikation als "Expected Packet Rate" Attribute des "Connection Objects" verwendet.

Electronic Key

Dient der Überprüfung der sich im Netz befindlichen Geräte beim Systemstart. Der Electronic Key wird bei jedem Systemstart aus den Geräten ausgelesen und mit der gespeicherten Konfiguration verglichen.

Polled**Produced/Consumed**

Aktivierung der Betriebsart "Polling", zyklisches Schreiben und Lesen von IO-Daten. Einstellung des Dateninhaltes der über die Polled IO Verbindung übertragenen Daten. Zur Auswahl stehen digitale Daten, analoge Daten oder beides. Die Auswahl hängt hierbei von den am BK52xx angeordneten Klemmen ab.

Bit-Strobed**Produced/Consumed**

Aktivierung der Betriebsart "Bit-Strobe". Mit einer Broadcast Message werden alle Knoten aufgefordert, ihre Bit-Strobe Message (bis 7 Bytes Eingangs- oder Statusdaten) zu senden. Einstellung des Dateninhaltes der über die Bit-Strobed IO Verbindung übertragenen Daten. Zur Auswahl stehen hierbei digitale Daten oder Diagnosedaten.

Change of State / Cyclic**Produced/Consumed**

Einstellung des Dateninhaltes der über die Change of State / Cyclic IO Verbindung übertragenen Daten. Zur Auswahl stehen digitale Daten, analoge Daten oder beides. Die Auswahl hängt hierbei von den am BK52xx angeordneten Klemmen ab.

Change of State / Cyclic

Auswahl der entsprechenden Betriebsart.

Heartbeat-Rate / Send-Rate

Die Heartbeat Rate gibt bei der Betriebsart "Change of State" die Cycle-Time an mit der IO-Daten untergelagert, d.h. zusätzlich zum ereignisgesteuerten senden, zyklisch gesendet werden. Die Send-Rate gibt bei der Betriebsart "Cyclic" die Cycle-Time an mit der IO-Daten gesendet werden.

Inhibit-Time

Verzögerungszeit bei der Betriebsart "Change of State", IO-Daten werden nach einem Zustandswechsel frühestens nach der hier eingestellten Zeit gesendet.

Acknowledge Timeout

Zeitspanne bis zur Sendewiederholung bei fehlendem Acknowledge auf eine Change of State / Cyclic Nachricht.

Acknowledge Retry Limit

Maximale Anzahl Sendewiederholungen bis IO-Verbindung in Fehlerzustand übergeht.

K-Bus Update

Berechnet die voraussichtliche Dauer für ein vollständiges Update des Klemmenbusses (ist abhängig von den angeschlossenen Klemmen).

Auto Device Replacement (ADR)

Nicht unterstützt.

Karteireiter "Startup-Attribute"

Class/Inst./Attr...	Length	Value	Name
100.1.1	1	01	Vendor Value 1
100.1.2	1	02	Vendor Value 2

Neu... Löschen... Bearbeiten...

Abb. 44: Karteireiter „Startup-Attribute“

Die Startup-Attribute werden vor dem zyklischen Datenaustausch zum Slave gesendet. Die Nachrichten werden vor dem eigentlichen IO-Datenverkehr gesendet.

Die Konfiguration erfolgt über den "Neu" oder "Bearbeiten" Button:

Edit Attribute Entry

Class Id (dez): 100

Instanz Id (dez): 1

Attribute Id (dez): 3

Länge (dez): 3

Wert (hex): 03 03 03

Name: Vendor Value 3

OK Abbruch

Abb. 45: Attribut Eintrag editieren

Die Attribute werden mit dem Class/Instance/Attribute initialisiert. Zu beachten ist hierbei die "Wert" - Angabe in hexadezimaler Form.

Karteireiter "ADS"

ADS Address (acyclic services): NetId: 10.5.2.7.3.1 Port: 4098 (0x1002)

ADS-Router on Box

☒ Enable Router

Net-Id: 10.5.2.7.3.6

Remote Name: Box 2 (BK5220)_Gerät 2 (EL6752)

Online-Access

Object Class: 0x03 Service Id:

Object Instance: 0x01 Attribute Id: 1

Read-Length: 1

Read-Data: 02

Write-Data:

Read Write ReadWrite

Abb. 46: Karteireiter „ADS“

Um DeviceNet® Objekte auch zur Laufzeit Schreiben und Lesen zu können (z. B. aus der SPS heraus), wird dem Knoten (Buskoppler) ein ADS-Port zugewiesen. Dieser kann bei Bedarf verändert werden. Eine detaillierte Beschreibung zur Ausführung von Expliziten Nachrichten ist im Kapitel "DeviceNet® Kommunikation" unter "[Explizite Nachrichten](#) [► 37]" beschrieben.

Über den Online Access kann auf DeviceNet® Objekte zugegriffen werden. Hierzu sind die DeviceNet-spezifischen Angaben wie Class/Instance/Attribute einzugeben.

Read

Lesen eines Objekt-Attributes via DeviceNet® "Get_Attribute_Single" Service. Eine Dienstkennung braucht hierbei nicht angegeben werden.

Write

Schreiben eines Objekt-Attributes via DeviceNet "Set_Attribute_Single" Service. Eine Dienstkennung braucht hierbei nicht angegeben werden.

Read/Write

Ausführen eines beliebigen DeviveNet Dienstes. Hierzu ist die Angabe der "Service-Id" erforderlich.

Karteireiter "Parameter"

N...	Name	Flags	Value
1	IO Error Action	u	Leave Local I/O Cycle + Reset O...
2	Input Data Bit Strobe	u	Discrete I/O
3	Input Size Poll Mode	ur	0 (0x0) Byte
4	Input Size COS/Cyc Mode	ur	0 (0x0) Byte
5	Data Size Bit Strobe	ur	0 (0x0) Byte
6	Output Size Poll/COS/Cyc	ur	0 (0x0) Byte
7	BK5220 Status	urm	0x0
8	Terminal No.	u	Coupler
9	Table No.	u	Table 0: Coupler or 1. Channel (T...
10	Register No.	u	0 (0x0)
11	Get Register data+status	ur	0x0
12	Set Register data	u	0x0
13	Device Diagnostics	u	OFF

Flags: u = unknown value; default value displayed, r = read only
 m = possibly modified by device in real time, * = modified by user

Schreiben Lesen Default setzen Alles wählen

Zu Startup Attributes kopieren All

Abb. 47: Karteireiter „Parameter“

Unter dem Karteireiter "Parameter" werden die aus der EDS-Datei ausgelesenen Parameter angezeigt. Hierbei besteht die Möglichkeit diese zu Lesen, zu Schreiben und in die Liste der Startup-Parameter zu übernehmen.

Karteireiter "Diag"

BoxState: No error
not implemented!

Refresh

Abb. 48: Karteireiter „Diag“

Der Karteireiter "Diag" gibt den Zustand der Box wieder. Eine weitere Diagnose ist nicht verfügbar.

6.6 Allgemeines DeviceNet® Gerät

DeviceNet® Teilnehmer werden als allgemeine DeviceNet® Geräte eingebunden.

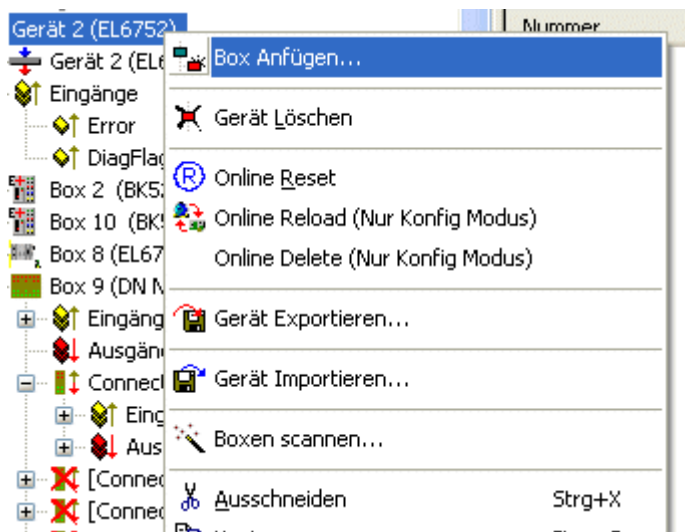


Abb. 49: Einfügen eines DeviceNet® Gerätes (E/A-Geräte -> Gerät n (EL6752) -> Rechte Maustaste -> Box anfügen...)

6.6.1 Einbinden eines DeviceNet® Gerätes mit EDS-File

Steht ein EDS-File für das einzubindende DeviceNet® Gerät zur Verfügung muss dieses in das **..TwinCAT/IO/DeviceNet®** Verzeichnis kopiert werden.

Im Anschluss erscheint das Gerät unter der Auswahl "Box Anfügen" s. Abb. *Einfügen eines DeviceNet® Gerätes (E/A-Geräte -> Gerät n (EL6752) -> Rechte Maustaste -> Box anfügen...)*, mit der Kennung der Herstellernamens:

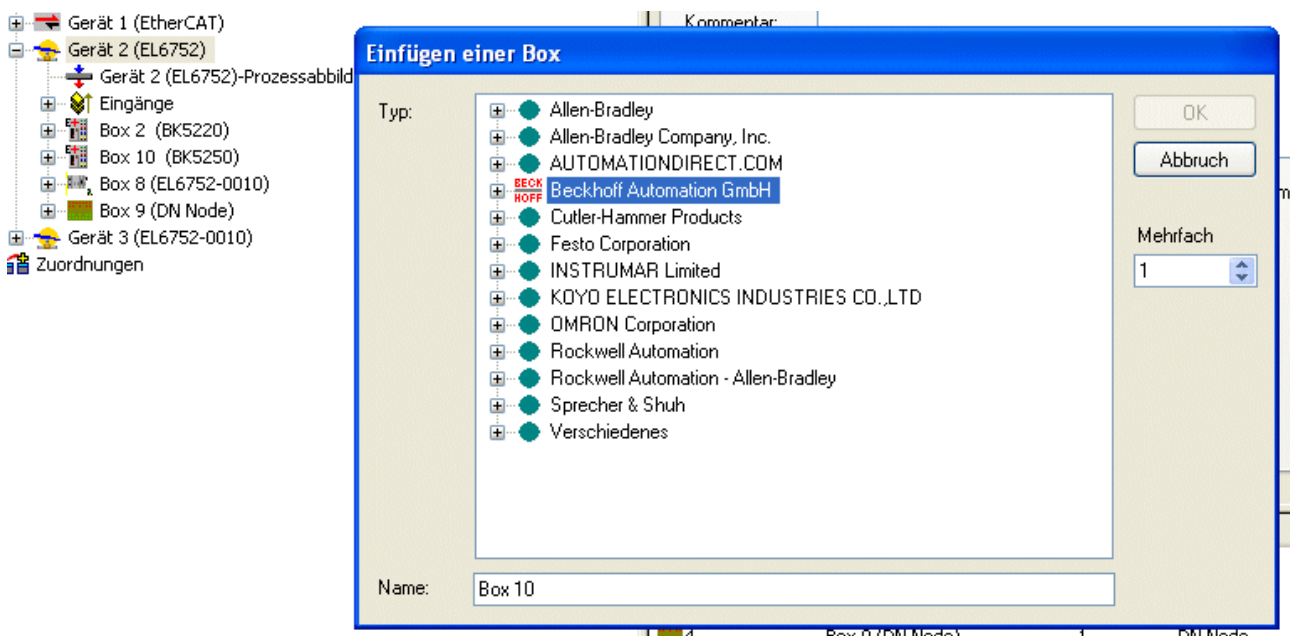


Abb. 50: Einfügen der Box mit der Kennung des Herstellernamens

Alternativ kann unter der Auswahl "Verschiedenes" ein DeviceNet® Gerät mit EDS-File eingebunden werden:

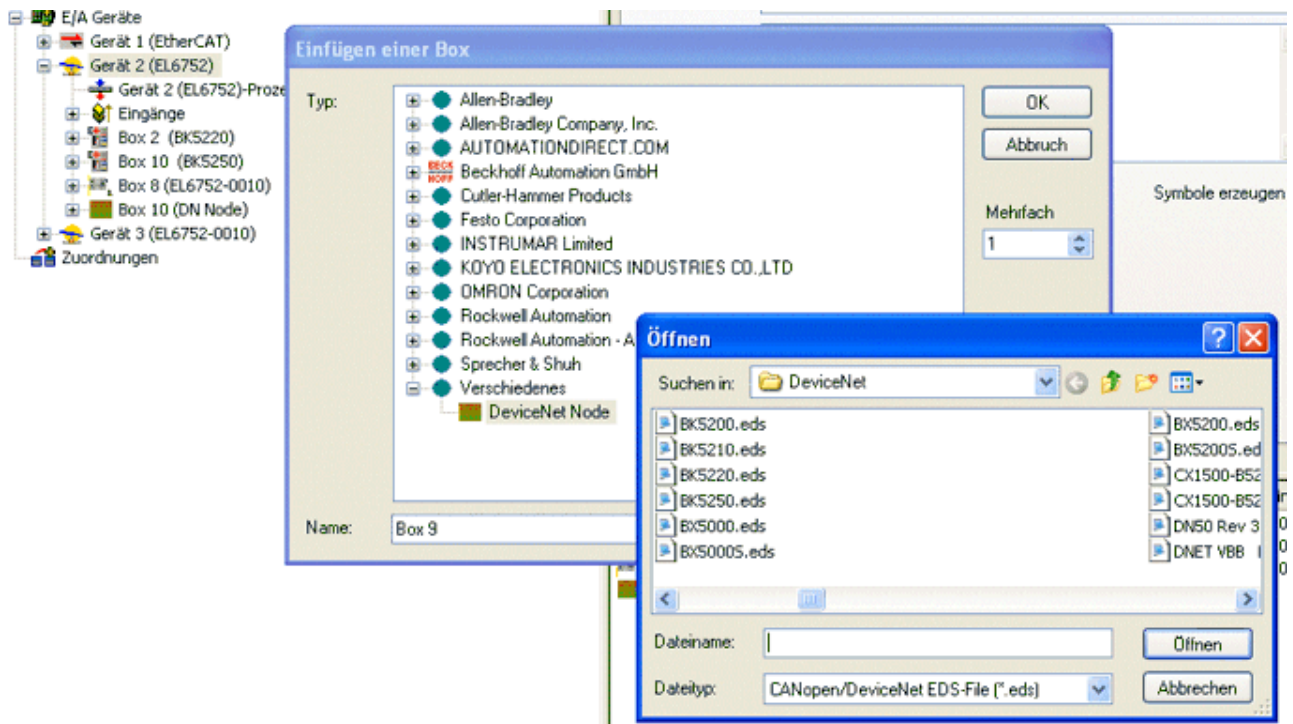


Abb. 51: Einfügen der Box ohne EDS-File

Je nach Angaben im EDS-File erscheint ein DeviceNet® Knoten mit/ohne Karteireiter Parameter.

Die IO-Betriebsart und die zugehörigen Datenlängen werden durch das EDS-File vorgegeben.

6.6.2 Einbinden eines DeviceNet® Gerätes ohne EDS-File

Unter der Auswahl "Verschiedenes" ist es möglich ein DeviceNet® Gerät ohne EDS-File einzubinden:

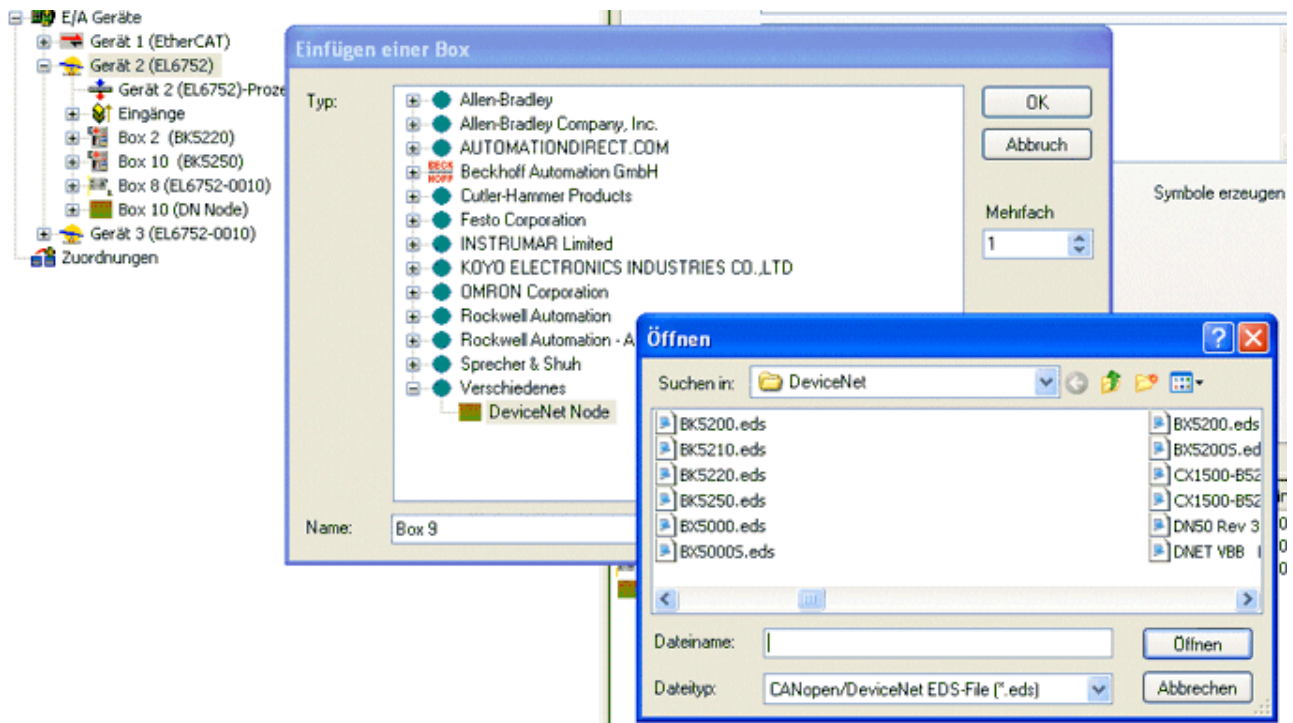


Abb. 52: Einfügen der Box ohne EDS-File („Abbrechen“ klicken)

Die Auswahl des EDS-Files muss mit "Abbrechen" beendet werden. Es wird ein allgemeines DeviceNet® Gerät angelegt.

Die Auswahl der IO-Betriebsart sowie der allgemeinen Konfiguration muss dann manuell vorgenommen werden.

DeviceNet® IO-Betriebsarten

Die EL6752 unterstützt für DeviceNet® Geräte die DeviceNet®-Betriebsarten zyklisches Polling, Change of State / Cyclic und Bit-Strobe. Die IO-Betriebsarten können entsprechend den DeviceNet®-Spezifikation selektiert werden.

Die DeviceNet® IO-Betriebsart zyklisches Polling wird per Default für die EL6752 selektiert:

IO-Betriebsart	Eingangsdatenlänge / Byte	Ausgangsdatenlänge / Byte
Polling	0 - 255	0 - 255
Change of State	0 - 255	0 - 255
Cyclic	0 - 255	0 - 255
Bit-Strobe	1 Bit	0-8
Summe alle EA-Daten	max. xxx Byte	max. xxx Byte

Polling / Change of State (COS) / Cyclic

Die Betriebsart zyklisches Polling ist charakterisiert durch ein zyklisches Abfragen bzw. Pollen der IO-Daten durch den Master. Die Betriebsart Change of State ist charakterisiert durch das ereignisorientierte Versenden der IO-Daten. In der Betriebsart Cyclic werden die IO-Daten zyklisch nach den durch den Master konfigurierten Kommunikationsparametern versendet. Die Einstellungen sind für die genannten Betriebsarten identisch.

Die Eingangs- und Ausgangsdatenlängen muss entsprechend der Gerätekonfiguration ergänzt werden:

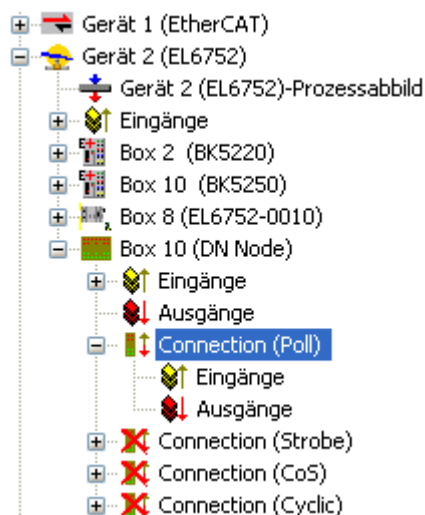


Abb. 53: Ergänzung der Ein- und Ausgangsdaten

Je nach Gerätekonfiguration müssen Eingangs- bzw. Ausgangsdaten angefügt werden. Hierbei können beliebige Datentypen ausgewählt werden:



Abb. 54: Einfügen von Variablen

Die Datenlänge wird entsprechend der DeviceNet® Spezifikation in ein Byte-Stream umgerechnet und auf dem Karteireiter der entsprechenden Verbindung zur Anzeige gebracht:

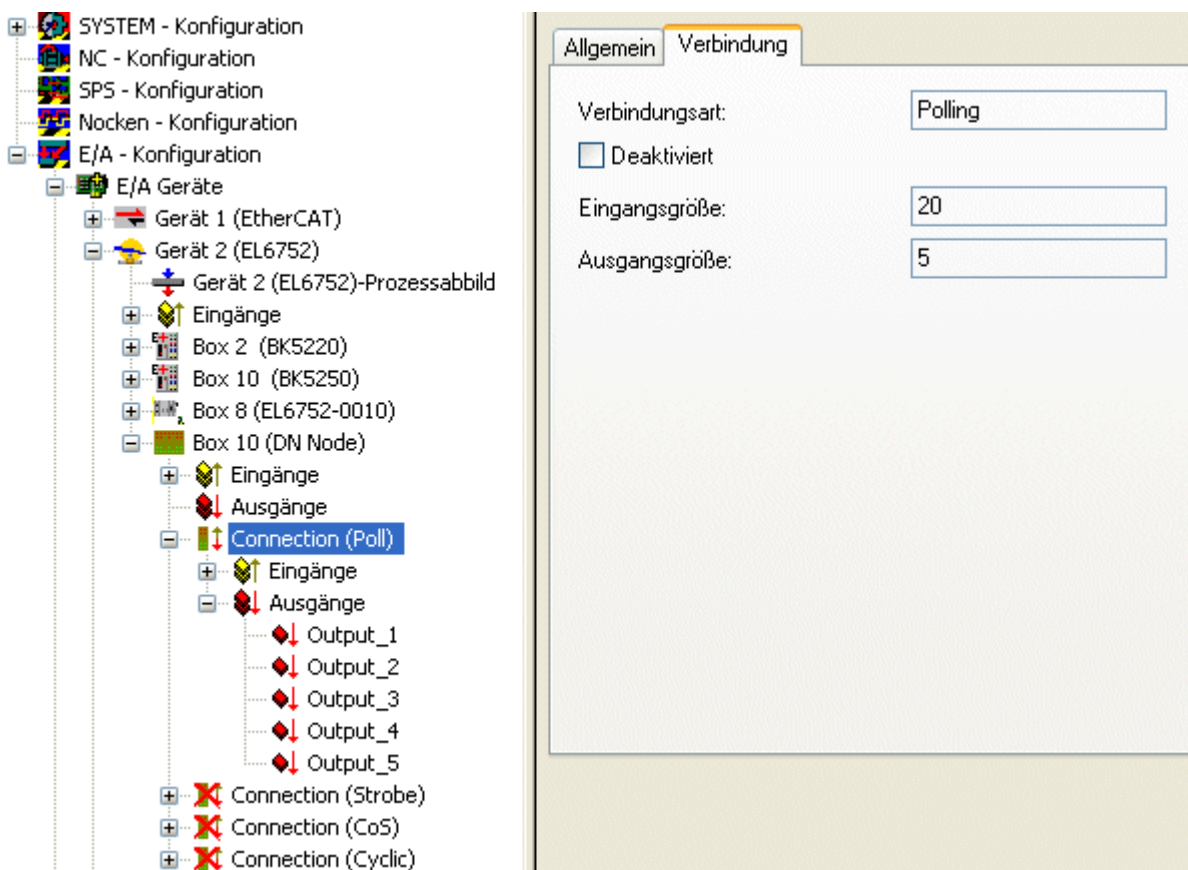


Abb. 55: Karteireiter „Verbindung“ mit Anzeige der Verbindungsart „Polling“ und der Ein- und Ausgangsgrößen

Maximale Datenlänge

Die maximale Datenlänge darf 255 Bytes pro Datenrichtung betragen.

Bit Strobe

Die IO-Betriebsart Bit-Strobe beinhaltet ein 8-Byte Kommando des Master an die Slaves. Für jeden mögliche Adresse/Mac-Id (DeviceNet Adressraum: 64) ist entsprechend 1 Bit Nutzdaten reserviert. Die Antwort-Nachricht des Slaves darf maximal 8 Byte lang sein und wird unmittelbar nach Erhalt des Bit-Strobe Kommandos an den Master gesendet.

Nach Auswahl der Bit-Strobe Betriebsart sind die Eingangsdaten entsprechend zu konfigurieren. Hierbei können beliebige Datentypen ausgewählt werden (siehe Polling/ COS / Cyclic). Die Datenlänge wird entsprechend der DeviceNet® Spezifikation in ein Byte-Stream umgerechnet und auf dem Karteireiter der Bit-Strobe Verbindung zur Anzeige gebracht:

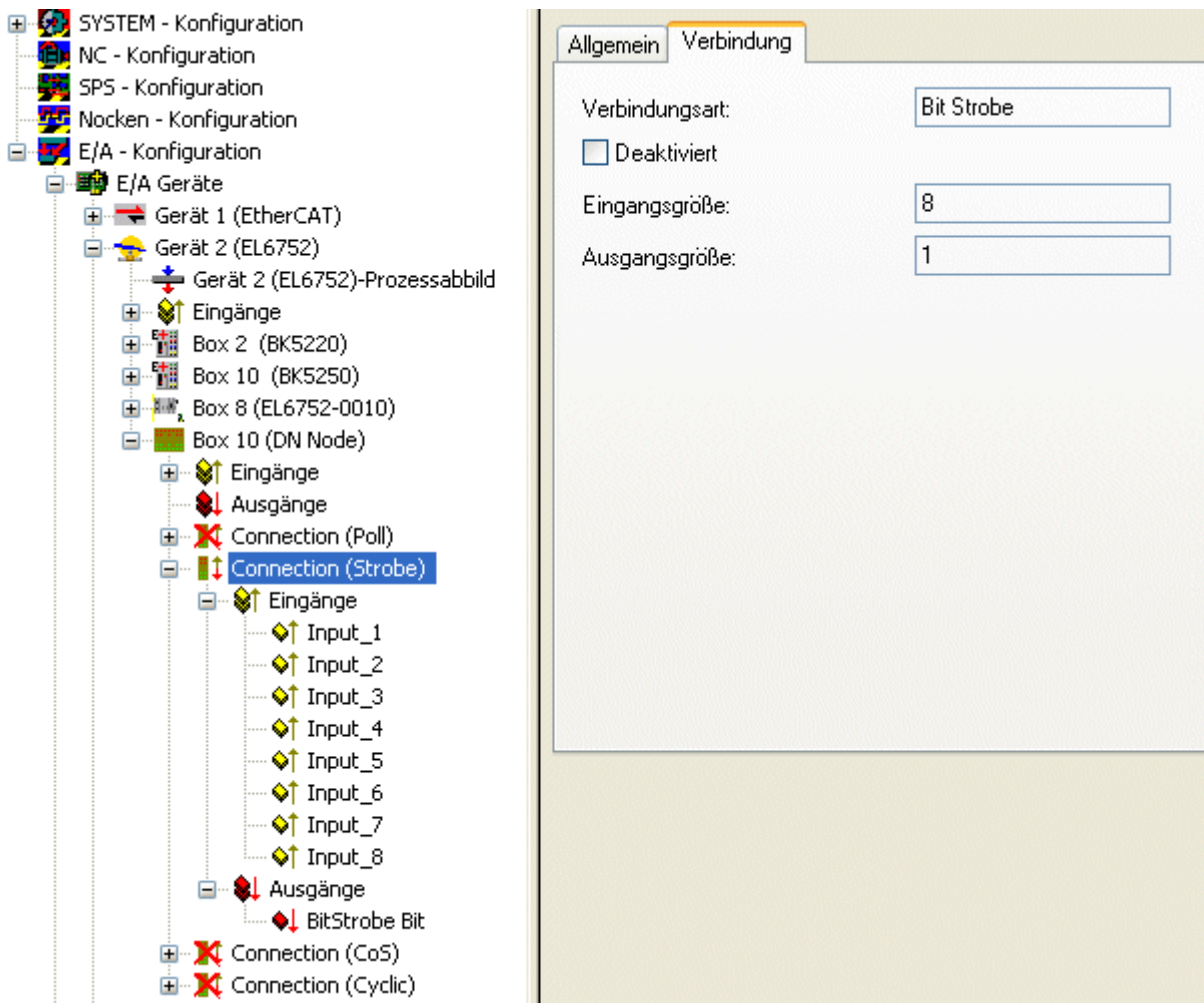


Abb. 56: Karteireiter „Verbindung“ mit Anzeige der Verbindungsart „Bit Strobe“ und der Ein- und Ausgangsgrößen

Maximale Datenlänge

Die maximale Eingangs-Datenlänge darf 8 Bytes betragen. Die Ausgangsdatenlänge ist fest vorgegeben.

Da die Kommunikationseinstellungen durch den Master vorgegeben werden sind hier keine weiteren Einstellungen mehr möglich.

6.6.3 Parametrierung eines DeviceNet® Gerätes

Die Parametrierung der DeviceNet Geräte findet mit folgenden Karteireitern statt:

- Karteireiter "DeviceNet Node" [► 69]
- Karteireiter "Startup-Attributes" [► 70]
- Karteireiter "ADS" [► 71]
- Karteireiter "Parameter" [► 72]
- Karteireiter "Diag" [► 72]

Karteireiter "DeviceNet Node"

Abb. 57: Karteireiter „DeviceNet Node“

MAC-ID

Stellt die MAC-ID, d.h. die Geräteadresse des DeviceNet® Teilnehmers ein (zwischen 0 und 63). Dieser Wert muss mit dem am Buskoppler bzw. an der Kompakt Box eingestellten Wert übereinstimmen.

Cycle-Time

Stellt die Zyklus-Zeit der IO-Verbindung Polling und Bit-Strobe ein. Der Wert wird gemäß DeviceNet® Spezifikation als "Expected Packet Rate" Attribute der "Connection Objects" verwendet.

Electronic Key

Dient der Überprüfung der sich im Netz befindlichen Geräte beim Systemstart. Der Electronic Key wird bei jedem Systemstart aus den Geräten ausgelesen und mit der gespeicherten Konfiguration verglichen.

Polled

Produced/Consumed

Aktivierung der Betriebsart "Polling", zyklisches Schreiben und Lesen von IO-Daten. Einstellung des Dateninhaltes der über die Polled IO Verbindung übertragenen Daten. Zur Auswahl stehen digitale Daten, analoge Daten oder beides. Die Auswahl hängt hierbei von den am BK52xx angeordneten Klemmen ab.

Bit-Strobed**Produced/Consumed**

Aktivierung der Betriebsart "Bit-Strobe". Mit einer Broadcast Message werden alle Knoten aufgefordert, ihre Bit-Strobe Message (bis 7 Bytes Eingangs- oder Statusdaten) zu senden. Einstellung des Dateninhaltes der über die Bit-Strobed IO Verbindung übertragenen Daten. Zur Auswahl stehen hierbei digitale Daten oder Diagnosedaten.

Change of State / Cyclic**Produced/Consumed**

Einstellung des Dateninhaltes der über die Change of State / Cyclic IO Verbindung übertragenen Daten. Zur Auswahl stehen digitale Daten, analoge Daten oder beides. Die Auswahl hängt hierbei von den am BK52xx angeordneten Klemmen ab.

Change of State / Cyclic

Auswahl der entsprechenden Betriebsart.

Heartbeat-Rate / Send-Rate

Die Heartbeat Rate gibt bei der Betriebsart "Change of State" die Cycle-Time an mit der IO-Daten untergelagert, d.h. zusätzlich zum ereignisgesteuerten senden, zyklisch gesendet werden. Die Send-Rate gibt bei der Betriebsart "Cyclic" die Cycle-Time an mit der IO-Daten gesendet werden.

Inhibit-Time

Verzögerungszeit bei der Betriebsart "Change of State", IO-Daten werden nach einem Zustandswechsel frühestens nach der hier eingestellten Zeit gesendet.

Acknowledge Timeout

Zeitspanne bis zur Sendewiederholung bei fehlendem Acknowledge auf eine Change of State / Cyclic Nachricht.

Acknowledge Retry Limit

Maximale Anzahl Sendewiederholungen bis IO-Verbindung in Fehlerzustand übergeht.

K-Bus Update

Berechnet die voraussichtliche Dauer für ein vollständiges Update des Klemmenbusses (ist abhängig von den angeschlossenen Klemmen).

Auto Device Replacement (ADR)

Nicht unterstützt.

Karteireiter "Startup-Attribute"

Class/Inst./Attr...	Length	Value	Name
100.1.1	1	01	Vendor Value 1
100.1.2	1	02	Vendor Value 2

Neu... Löschen... Bearbeiten...

Abb. 58: Karteireiter „Startup-Attribute“

Die Startup-Attribute werden vor dem zyklischen Datenaustausch zum Slave gesendet. Die Nachrichten werden vor dem eigentlichen IO-Datenverkehr gesendet.

Die Konfiguration erfolgt über den "Neu" oder "Bearbeiten" Button:

Edit Attribute Entry

Class Id (dez): 100

Instanz Id (dez): 1

Attribute Id (dez): 3

Länge (dez): 3

Wert (hex): 03 03 03

Name: Vendor Value 3

OK

Abbruch

Abb. 59: Attribut Eintrag editieren

Die Attribute werden mit dem Class/Instance/Attribute initialisiert. Zu beachten ist hierbei die "Wert" - Angabe in hexadezimaler Form.

Karteireiter "ADS"

Allgemein BK52x0/IX-B52x Startup Attribute **ADS** Parameter Diag

ADS Address (acyclic services): NetId: 10.5.2.7.3.1 Port: 4098 (0x1002)

ADS-Router on Box

☒ Enable Router

Net-Id: 10.5.2.7.3.6

Remote Name: Box 2 (BK5220)_Gerät 2 (EL6752)

Online-Access

Object Class: 0x03 Service Id:

Object Instance: 0x01 Attribute Id: 1

Read-Length: 1

Read-Data: 02

Write-Data:

Read Write ReadWrite

Abb. 60: Karteireiter „ADS“

Um DeviceNet® Objekte auch zur Laufzeit Schreiben und Lesen zu können (z. B. aus der SPS heraus), wird dem Knoten (Buskoppler) ein ADS-Port zugewiesen. Dieser kann bei Bedarf verändert werden. Eine detaillierte Beschreibung zur Ausführung von Expliziten Nachrichten ist im Kapitel "DeviceNet® Kommunikation" unter "Explizite Nachrichten" beschrieben.

Über den Online Access kann auf DeviceNet® Objekte zugegriffen werden. Hierzu sind die DeviceNet-spezifischen Angaben wie Class/Instance/Attribute einzugeben.

Read

Lesen eines Objekt-Attributes via DeviceNet® "Get_Attribute_Single" Service. Eine Dienstkennung braucht hierbei nicht angegeben werden.

Write

Schreiben eines Objekt-Attributes via DeviceNet® "Set_Attribute_Single" Service. Eine Dienstkennung braucht hierbei nicht angegeben werden.

Read/Write

Ausführen eines beliebigen DeviveNet Dienstes. Hierzu ist die Angabe der "Service-Id" erforderlich.

Karteireiter "Parameter"

N...	Name	Flags	Value
1	IO Error Action	u	Leave Local I/O Cycle + Reset O...
2	Input Data Bit Strobe	u	Discrete I/O
3	Input Size Poll Mode	ur	0 (0x0) Byte
4	Input Size COS/Cyc Mode	ur	0 (0x0) Byte
5	Data Size Bit Strobe	ur	0 (0x0) Byte
6	Output Size Poll/COS/Cyc	ur	0 (0x0) Byte
7	BK5220 Status	urm	0x0
8	Terminal No.	u	Coupler
9	Table No.	u	Table 0: Coupler or 1. Channel (T...
10	Register No.	u	0 (0x0)
11	Get Register data+status	ur	0x0
12	Set Register data	u	0x0
13	Device Diagnostics	u	OFF

Flags: u = unknown value; default value displayed, r = read only
m = possibly modified by device in real time, * = modified by user

Schreiben Lesen Default setzen Alles wählen

Zu Startup Attributes kopieren All

Abb. 61: Karteireiter „Parameter“

Unter dem Karteireiter "Parameter" werden die aus der EDS-Datei ausgelesenen Parameter angezeigt. Hierbei besteht die Möglichkeit diese zu Lesen, zu Schreiben und in die Liste der Startup-Parameter zu übernehmen.

Karteireiter "Diag"

BoxState: No error
not implemented!

Refresh

Abb. 62: Karteireiter „Diag“

Der Karteireiter "Diag" gibt den Zustand der Box wieder. Eine weitere Diagnose ist nicht verfügbar.

6.7 EtherCAT Beschreibung

6.7.1 Einführung

Die verschiedenen DeviceNet® Funktionalitäten sowie Konfigurationsmöglichkeiten sind in Abhängigkeit der unterschiedlichen EtherCAT Zustände veränderbar bzw. parametrierbar.

EtherCAT Zustände

Die EtherCAT Zustände (INIT, PREOP, SAFEOP, OP) haben entsprechend den feldbusspezifischen Funktionen folgende Bedeutung:

EtherCAT Zustand	Bedeutung
INIT	Feldbus läuft nicht
PREOP	Feldbuskonfiguration laden
SAFEOP	Feldbus zyklischer Betrieb, sicherer Zustand. Eingänge werden gelesen, Ausgänge werden nicht geschrieben
OP	Feldbus zyklischer Betrieb. Eingänge werden gelesen, Ausgänge werden geschrieben

Im Folgenden werden das Vorgehen und die Konfigurationsmöglichkeiten beschrieben

6.7.1.1 EL6752 - DeviceNet® Master Konfiguration

Die Konfiguration des DeviceNet® Masters sowie der zugehörigen DeviceNet® Slaves wird im EtherCAT-Zustand PREOP durchgeführt. Die DeviceNet® Master Parameter werden über das EtherCAT Objekt 0xF800 geschrieben, die Slave Parameter werden über die EtherCAT Objekte ab 0x80n0 [► 80] geschrieben, siehe Kapitel EtherCAT Objekt Beschreibung.

Die EtherCAT Zustände werden wie folgt auf DeviceNet® abgebildet:

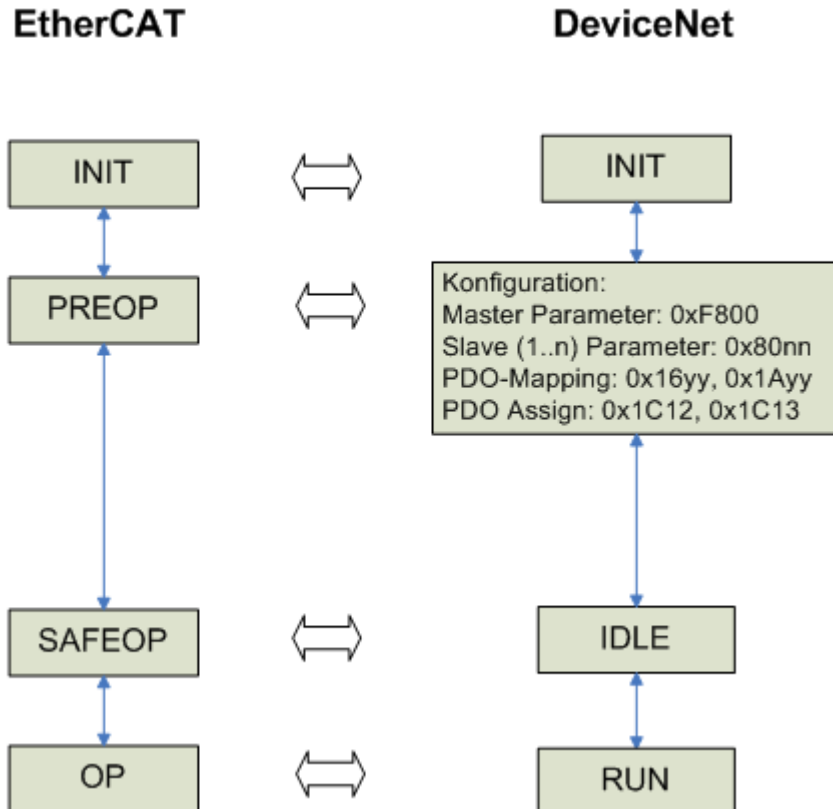


Abb. 63: EtherCAT Stati in Abbildung auf EL6752-0000

Das EtherCAT PDO-Mapping (EtherCAT Objekte 0x16yy, 0x1Ayy) sowie das PDO-Assignment (EtherCAT Objekte 0x1C12, 0x1C13) kann nach dem Schreiben der DeviceNet® Master Parameter und dem Schreiben der DeviceNet® Slave Parameter gelesen werden. Im Anschluss daran wird entsprechend das Prozessabbild generiert.

Nach dem die DeviceNet® Master Parameter über das EtherCAT Objekt 0xF800 [► 93] geschrieben worden sind, meldet sich der DeviceNet® Master am Netzwerk an und führt den Duplicate MAC-ID Check durch.

Starten des Feldbusses

Beim EtherCAT Zustandsübergang von PREOP nach SAFEOP startet der DeviceNet® Master die Datenkommunikation mit den Slaves und allokiert die konfigurierten Betriebsarten. Im EtherCAT Zustand SAFEOP befindet sich der DeviceNet® Master im IDLE Mode. Beim EtherCAT Zustandsübergang von SAFEOP nach OP schaltet der DeviceNet® Master in den RUN Mode.

Laden einer neuen Konfiguration

Eine neue DeviceNet® Konfiguration kann nur durch einen EtherCAT Zustandsübergang nach IDLE oder PREOP geladen werden. Im Anschluss müssen wieder die DeviceNet® Master Parameter und die DeviceNet® Slave Parameter geschrieben werden.

6.7.1.2 EL6752-0010 - DeviceNet® Slave Konfiguration

Die Konfiguration des DeviceNet® Slaves wird im EtherCAT-Zustand PREOP durchgeführt. Die allgemeinen DeviceNet® Slave Parameter werden über das EtherCAT Objekt 0xF800 geschrieben, die Slave Konfigurationsdaten, d.h. die Kommunikationseigenschaften und die IO-Konfiguration wird über das EtherCAT Objekt 0x8000 [► 96] geschrieben, siehe Kapitel EtherCAT Objekt Beschreibung.

Die EtherCAT Zustände werden wie folgt auf DeviceNet® abgebildet:

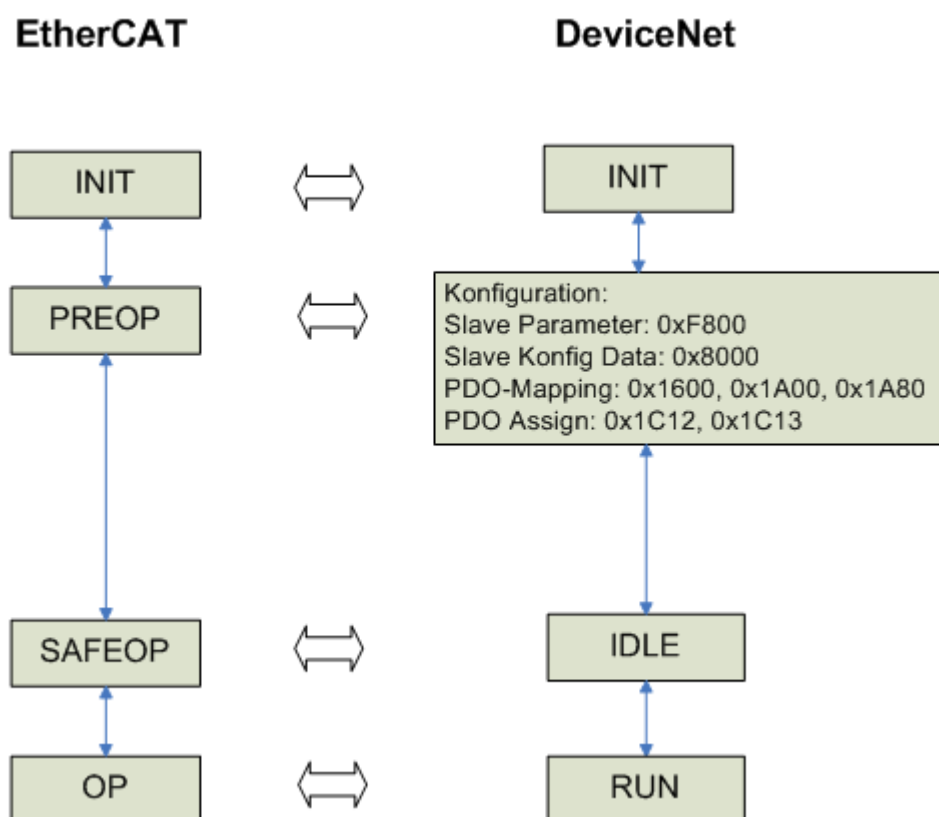


Abb. 64: EtherCAT Stati in Abbildung auf EL6752-0010

Das EtherCAT PDO-Mapping (EtherCAT Objekte 0x1600, 0x1A00, 0x1A80) sowie das PDO-Assignment (EtherCAT Objekte 0x1C12 [[► 99](#)], 0x1C13 [[► 99](#)]) kann nach dem Schreiben der DeviceNet® Slave Parameter und dem Schreiben der DeviceNet® Slave Konfigurationsdaten gelesen werden. Im Anschluss daran wird entsprechend das Prozessabbild generiert.

Nach dem die DeviceNet® Slave Parameter über das EtherCAT Objekt 0xF800 geschrieben worden sind meldet sich der DeviceNet® Slave am Netzwerk an und führt den Duplicate MAC-ID Check durch.

Starten des Feldbusses

Beim EtherCAT Zustandsübergang von PREOP nach SAFEOP startet der DeviceNet® Slave die Datenkommunikation, d.h. er ist nun bereit zur Kommunikation mit einem DeviceNet® Master. Im EtherCAT Zustand SAFEOP befindet sich der DeviceNet® Slave im IDLE Mode. Beim EtherCAT Zustandsübergang von SAFEOP nach OP schaltet der DeviceNet® Slave in den RUN Mode.

Laden einer neuen Konfiguration

Eine neue DeviceNet® Konfiguration kann nur durch einen EtherCAT Zustandsübergang nach IDLE oder PREOP geladen werden. Im Anschluss müssen wieder die DeviceNet® Slave Parameter und die DeviceNet® Slave Konfigurationsdaten geschrieben werden.

6.7.1.3 EL6752-0010 - DeviceNet® Adresse und Baudrate über ADS ändern

Die DeviceNet Adresse (MACId) und die Baudrate der DeviceNet Slave Klemme EL6752-0010 kann neben den bekannten Funktionen wie sie bereits im Kapitel "Konfiguration mit dem [TwinCAT System Manager](#) [[► 53](#)]" beschrieben sind auch über ein ADS-Kommando gesetzt werden

ADS-Kommando

Setzen der MAC-ID und der Baudrate über ADS

```
IDXGRP=0x1F480
Index Offset 0x00

LEN=6

DATA[0]=0x45
DATA[1]=0x23
DATA[2]=MACId (0 _ 63)
DATA[3]=0
DATA[4]=Baudrate (1=500k, 2=250k, 3=125k)
DATA[5]=0

Ams Net Id: die der EL6752
Ams Port: 200
```

Nach dem Schreiben des Kommandos muss die Klemme einmal in INIT und zurück in OP geschaltet werden. Die gesetzten Daten können im Objekt 0xF800 Index 1 (MAC ID) und Index 2 (Baudrate) gelesen werden.

Kommando am Beispiel des TwinCAT AMS ADS Viewer

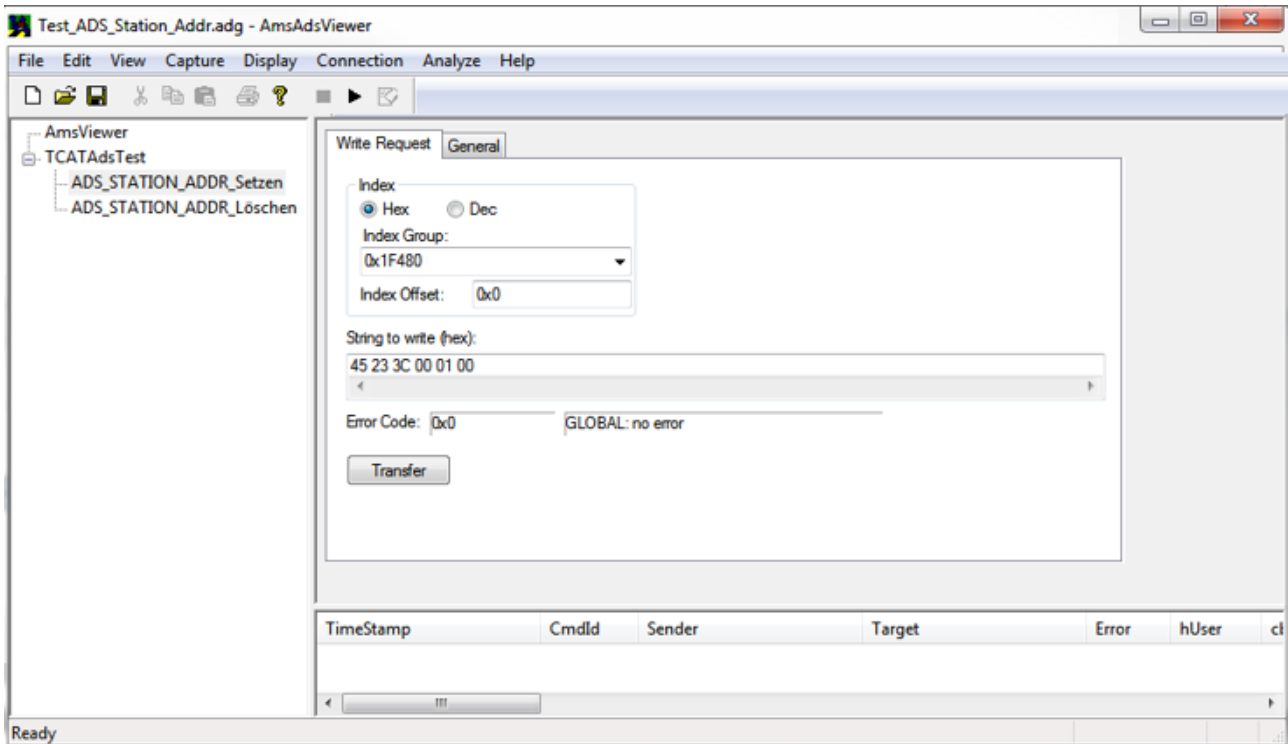


Abb. 65: ADS Kommando mit den Daten 3C - MACId (60dez) und 01 - Baudrate (500k)

Rücksetzen

Werden die MAC ID und die Baudrate einmal über den ADS Befehl gesetzt, speichert die Klemme die Informationen persistent ab. Sind diese Daten einmal geschrieben worden, werden die Einträge der Objekte 0x8000:01, 0xF800:01 und 0xF800:02 nicht beachtet. ! Dies betrifft die Startup Commands, welche dann von der Klemme ignoriert werden.

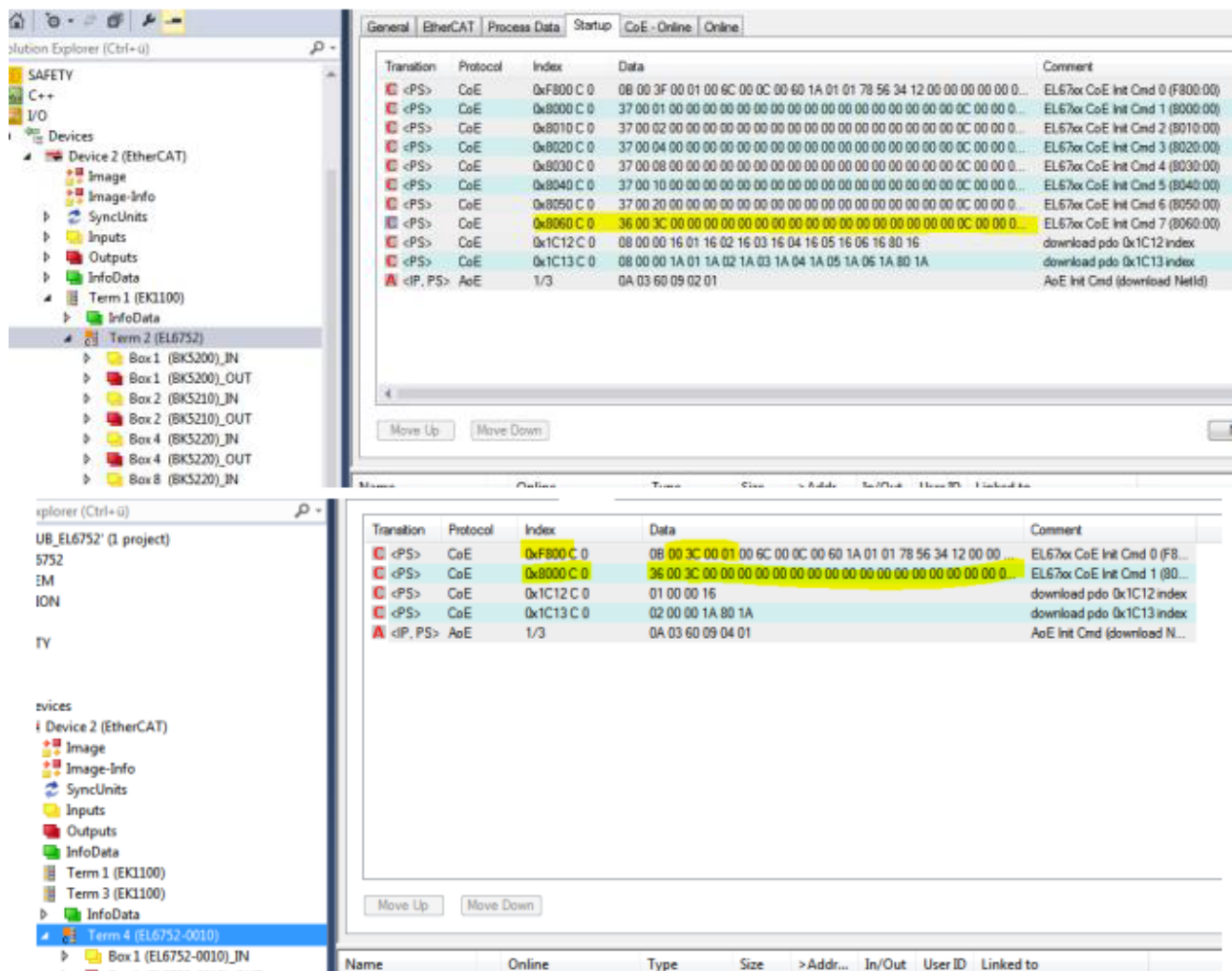


Abb. 66: Beispiel von Startup CMD, welche bei erfolgreich über ADS gesetzter MACId und Baudrate von der Slave-Klemme ignoriert werden (0x8000:01; 0xF800:01 und 0xF800:02)

ADS Kommando (Rücksetzen)

```
IDXGRP=0x1F480
Index Offset 0x00
```

```
LEN=6
```

```
DATA[0]=0
DATA[1]=0
DATA[2]=0
DATA[3]=0
DATA[4]=0
DATA[5]=0
```

```
Ams Net Id: die der EL6752
Ams Port: 200
```

So können die Daten wieder dauerhaft gelöscht werden und die Klemme verhält sich wie im Auslieferungszustand.

Rücksetz - Kommando am Beispiel des TwinCAT AMS ADS Viewer

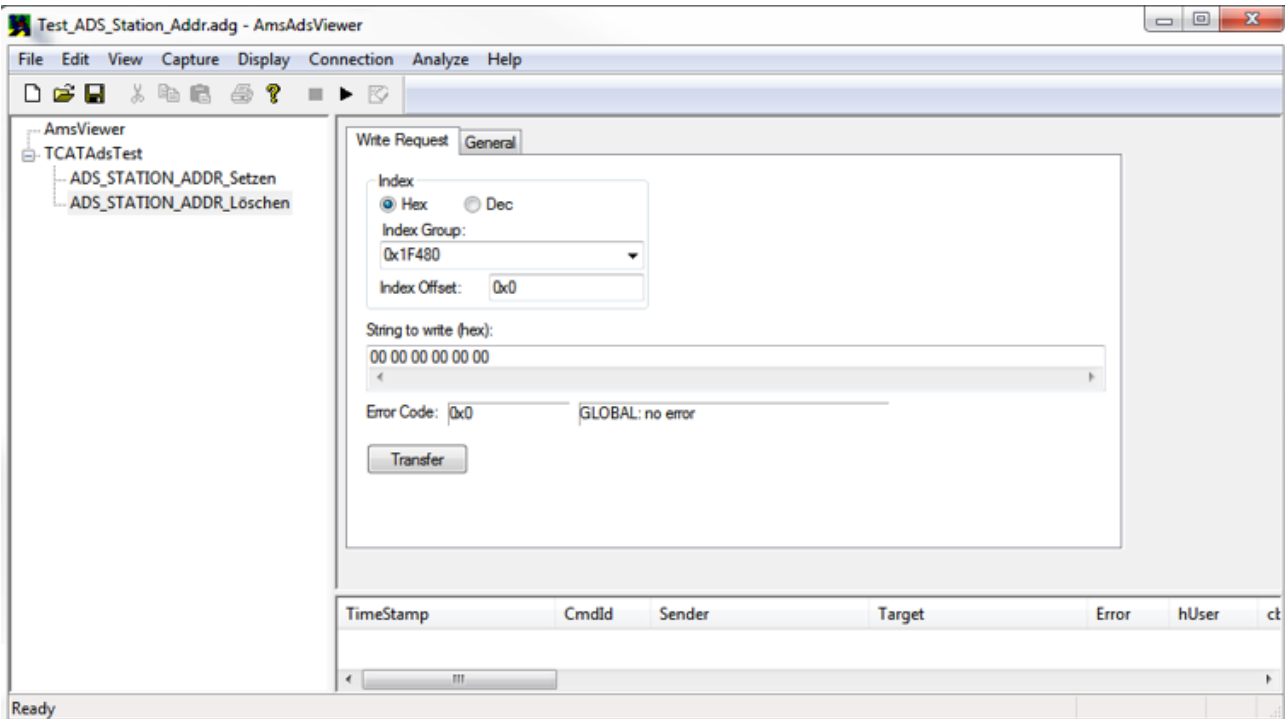


Abb. 67: Rücksetzen der persistenten Daten für MAC ID und Baudrate

6.7.2 Objektbeschreibung und Parametrierung

6.7.2.1 DeviceNet® Master - EL6752

● EtherCAT ESI Device Description (XML)

i Die Darstellung entspricht der Anzeige der CoE-Objekte aus der EtherCAT ESI Device Description (XML). Es wird empfohlen, die entsprechende aktuellste XML-Datei im Download-Bereich auf der [Beckhoff-Website](#) herunterzuladen und entsprechend der Installationsanweisungen zu installieren.

● Parametrierung über das CoE-Verzeichnis (CAN over EtherCAT)

i Die Parametrierung des EtherCAT-Gerätes wird über den CoE-Online Reiter (mit Doppelklick auf das entsprechende Objekt) bzw. über den Prozessdatenreiter (Zuordnung der PDOs) vorgenommen. Beachten Sie bei Verwendung/Manipulation der CoE-Parameter die allgemeinen CoE-Hinweise [► 39]:

- StartUp-Liste führen für den Austauschfall
- Unterscheidung zwischen Online/Offline Dictionary, Vorhandensein aktueller XML-Beschreibung
- „CoE-Reload“ zum Zurücksetzen der Veränderungen

Einführung

In der CoE-Übersicht sind Objekte mit verschiedenem Einsatzzweck enthalten:

- Objekte, die zu Parametrierung bei der Inbetriebnahme nötig sind
- Objekte, die interne Settings anzeigen und ggf. nicht veränderlich sind.

Im Folgenden werden zuerst die zur Parametrierung und zum normalen Betrieb benötigten Objekte vorgestellt. Alle weiteren Objekte, die für den normalen Anwendungsfall nicht erforderlich sind, sind im unteren Tabellenabschnitt zu finden.

6.7.2.1.1 Objekte für die Parametrierung

Index 8000-803E Configuration Data Slave 1

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
8000+n*16: 0	Configuration Data Slave 1	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x21 (33 _{dez})
(8000+n*16): :01	MAC ID	DeviceNet Geräte Adresse (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16): :02	ProductName	Produktname	OCTET-STRING[32]	RW	{0}
(8000+n*16): :03	Device Type	Gerätetyp (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16): :04	Vendor ID	Herstellerkennung (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16): :05	Product Code	Produktkennung (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16): :06	Revision Number	Versionsnummer (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16): :07	Serial Number	Seriennummer (siehe DeviceNet-Spezifikation)	UINT32	RW	0x00000000 (0 _{dez})
(8000+n*16): :08	Network Flags	reserviert für AMS über DeviceNet	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16): :09	Network Port	reserviert für AMS über DeviceNet	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16): :0A	Network Segment Address	reserviert für AMS über DeviceNet	OCTET-STRING[6]	RW	{0}
(8000+n*16): :0B	Reserved	reserviert	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16): :0C	Electronic Key	Electronic Key Bit-Maske: Bit 0: Check Vendor Id Bit 1: Check DeviceType Bit 2: Check Product Code Bit 3: Check Revision Bit 4: reserved(0) Bit 5: reserved(0) Bit 6: reserved(0) Bit 7: reserved(0)	UINT16	RW	0x0100 (256 _{dez})
(8000+n*16): :0D	Allocation Choice	DeviceNet Betriebsarten-Auswahl (siehe DeviceNet-Spezifikation) Bit 0: reserved (0) Bit1: Polled Bit2: Bit-Strobed Bit3: reserved (0) Bit4: Change of State Bit5: Cyclic Bit6: Acknowledge Suppression Bit7: reserved(0)	UINT16	RW	0x0100 (256 _{dez})
(8000+n*16): :0E	Expected Packet Rate - Poll	Timing-Parameter der Poll-Verbindung (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16): :0F	Expected Packet Rate - Bit Strobe	Timing-Parameter der Bit-Strobe-Verbindung (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16): :10	Expected Packet Rate - COS/Cyclic	Timing-Parameter der COS/Cyclic-Verbindung (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16): :11	Produced Data Size - Poll	Datenlänge im Poll-Modus	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16): :12	Produced Data Size - Bit Strobe	Datenlänge im Bit-Strobe-Modus	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16): :13	Produced Data Size - COS/Cyclic	Datenlänge im Change of State / Cyclic Modus	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16): :14	Reserved	reserviert	UINT16	RW	0x0000 (0 _{dez})

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
(8000+n*16):15	Reserved	reserviert	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16):16	Reserved	reserviert	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16):17	Consumed Data Size - Poll	Datenlänge im Poll-Modus	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16):18	Consumed Data Size - Bit Strobe	Datenlänge im Bit-Strobe-Modus	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16):19	Consumed Data Size - COS/Cyclic	Datenlänge im Change of State / Cyclic Modus	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16):1A	Reserved	reserviert	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16):1B	Reserved	reserviert	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16):1C	Reserved	reserviert	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16):1D	Inhibit Time	Timing-Parameter der COS/Cyclic-Verbindung (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16):1E	Acknowledge Timer	Timing-Parameter der COS/Cyclic-Verbindung (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16):1F	Acknowledge Retry Limit	Timing-Parameter der COS/Cyclic-Verbindung (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0100 (256 _{dez})
(8000+n*16):20	Reserved	reserviert	UINT16	RW	0x0000 (0 _{dez})
(8000+n*16):21	Reserved	reserviert	UINT16	RW	0x0000 (0 _{dez})

Index 8001-803F StartUp Attributes Slave 1

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
8001+n*16:0	StartUp Attributes Slave 1	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x01 (1 _{dez})
(8001+n*16):01	Start Up Attribute.000		OCTET-STRING[32]	RW	{0}

6.7.2.1.2 Objekte für interne Settings

6.7.2.1.2.1 Standardobjekte (0x1000-0x1FFF)

Index 1000 Device type

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1000:0	Device type	Geräte-Typ des EtherCAT-Slaves: Das Lo-Word enthält das verwendete CoE Profil (5001). Das Hi-Word enthält das Modul Profil entsprechend des Modular Device Profile.	UINT32	RO	0x14501389 (340792201 _{dez})

Index 1008 Device name

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1008:0	Device name	Geräte-Name des EtherCAT-Slave	STRING	RO	EL6752

Index 1009 Hardware version

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1009:0	Hardware version	Hardware-Version des EtherCAT-Slaves	STRING	RO	00

Index 100A Software version

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
100A:0	Software version	Firmware-Version des EtherCAT-Slaves	STRING	RO	00

Index 1018 Identity

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1018:0	Identity	Informationen, um den Slave zu identifizieren	UINT8	RO	0x04 (4 _{dez})
1018:01	Vendor ID	Hersteller-ID des EtherCAT-Slaves	UINT32	RO	0x00000002 (2 _{dez})
1018:02	Product code	Produkt-Code des EtherCAT-Slaves	UINT32	RO	0x1A603052 (442511442 _{dez})
1018:03	Revision	Revisionsnummer des EtherCAT-Slaves, das Low-Word (Bit 0-15) kennzeichnet die Sonderklemmennummer, das High-Word (Bit 16-31) verweist auf die Gerätebeschreibung	UINT32	RO	0x00100000 (1048576 _{dez})
1018:04	Serial number	Seriennummer des EtherCAT-Slaves, das Low-Byte (Bit 0-7) des Low-Words enthält das Produktionsjahr, das High-Byte (Bit 8-15) des Low-Words enthält die Produktionswoche, das High-Word (Bit 16-31) ist 0	UINT32	RO	0x00000000 (0 _{dez})

Index 10E2 Manufacturer-specific Identification Code

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
10E2:0	Manufacturer-specific Identification Code	Herstellerspezifischer Identifizierung Code	UINT8	RO	0x01 (1 _{dez})
10E2:01	SubIndex 001	reserviert	STRING	RO	

Index 1600-163E DNS RxPDO-Map Slave 0

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1600+n:0	DNS RxPDO-Map Slave 0	PDO Mapping RxPDO 1 (je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1600+n):01					
...					
(1600+n):FF					

Index 1650-168E DNS RxPDO-Map Slave 5

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1650+n:0	DNS RxPDO-Map Slave 5	PDO Mapping RxPDO 81 (je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1650+n):01					
...					
(1650+n):FF					

Index 16A0-16DE DNS RxPDO-Map Slave 10

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
16A0+n:0	DNS RxPDO-Map Slave 10	PDO Mapping RxPDO 161 (je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(16A0+n):01					
...					
(16A0+n):F F					

Index 16F0-172E DNS RxPDO-Map Slave 15

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
16F0+n:0	DNS RxPDO-Map Slave 15	PDO Mapping RxPDO 241 (je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(16F0+n):01					
...					
(16F0+n):FF					

Index 1740-177E DNS RxPDO-Map Slave 20

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1740+n:0	DNS RxPDO-Map Slave 20	PDO Mapping RxPDO 321 (je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1740+n):01					
...					
(1740+n):FF					

Index 1790-17CE DNS RxPDO-Map Slave 25

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1790+n:0	DNS RxPDO-Map Slave 25	PDO Mapping RxPDO 401 (je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1790+n):01					
...					
(1790+n):FF					

Index 17E0-181E DNS RxPDO-Map Slave 30

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
17E0+n:0	DNS RxPDO-Map Slave 30	PDO Mapping RxPDO 481 (je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(17E0+n):01					
...					
(17E0+n):F F					

Index 1830-186E DNS RxPDO-Map Slave 35

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1830+n:0	DNS RxPDO-Map Slave 35	PDO Parameter TxPDO 49 (je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1830+n):01					
...					
(1830+n):FF					

Index 1880-18BE DNS RxPDO-Map Slave 40

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1880+n:0	DNS RxPDO-Map Slave 40	PDO Parameter TxPDO 129 (je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1880+n):01					
...					
(1880+n):FF					

Index 18D0-190E DNS RxPDO-Map Slave 45

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
18D0+n:0	DNS RxPDO-Map Slave 45	PDO Parameter TxPDO 209 (je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(18D0+n):01					
...					
(18D0+n):F F					

Index 1920-195E DNS RxPDO-Map Slave 50

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1920+n:0	DNS RxPDO-Map Slave 50	PDO Parameter TxPDO 289 (je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1920+n):01					
...					
(1920+n):FF					

Index 1970-19AE DNS RxPDO-Map Slave 55

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1970+n:0	DNS RxPDO-Map Slave 55	PDO Parameter TxPDO 369 (je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1970+n):01					
...					
(1970+n):FF					

Index 19C0-19FE DNS RxPDO-Map Slave 60

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
19C0+n:0	DNS RxPDO-Map Slave 60	PDO Parameter TxPDO 449 (je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(19C0+n):01					
...					
(19C0+n):F F					

Index 1AB0-1AEE DNS TxPDO-Map Slave 11

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1AB0+n:0	DNS TxPDO-Map Slave 11	PDO Mapping TxPDO 177 (je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1AB0+n):0 1					
...					
(1AB0+n):F F					

Index 1B00-1B3E DNS TxPDO-Map Slave 16

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1B00+n:0	DNS TxPDO-Map Slave 16	PDO Mapping TxPDO 257 (je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1B00+n):01					
...					

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
(1B00+n):F F					

Index 1B50-1B8E DNS TxPDO-Map Slave 21

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1B50+n:0	DNS TxPDO-Map Slave 21	PDO Mapping TxPDO 337 (je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1B50+n):01					
...					
(1B50+n):F F					

Index 1BA0-1BDE DNS TxPDO-Map Slave 26

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1BA0+n:0	DNS TxPDO-Map Slave 26	PDO Mapping TxPDO 417 (je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1BA0+n):0 1					
...					
(1BA0+n):F F					

Index 1BF0-1C2E DNS TxPDO-Map Slave 31

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1BF0+n:0	DNS TxPDO-Map Slave 31	PDO Mapping TxPDO 497 (je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1BF0+n):01					
...					
(1BF0+n):F F					

Index 1C00 Sync manager type

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1C00:0	Sync manager type	Benutzung der Sync Manager	UINT8	RO	0x04 (4 _{dez})
1C00:01	SubIndex 001	Sync-Manager Type Channel 1: Mailbox Write	UINT8	RO	0x01 (1 _{dez})
1C00:02	SubIndex 002	Sync-Manager Type Channel 2: Mailbox Read	UINT8	RO	0x02 (2 _{dez})
1C00:03	SubIndex 003	Sync-Manager Type Channel 3: Process Data Write (Outputs)	UINT8	RO	0x03 (3 _{dez})
1C00:04	SubIndex 004	Sync-Manager Type Channel 4: Process Data Read (Inputs)	UINT8	RO	0x04 (4 _{dez})

Index 1C32 SM output parameter

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1C32:0	SM output parameter	Synchronisierungsparameter der Outputs	UINT8	RO	0x20 (32 _{dez})
1C32:01	Sync mode	Aktuelle Synchronisierungsbetriebsart: <ul style="list-style-type: none"> 0: Free Run 1: Synchron with SM 2 Event 2: DC-Mode - Synchron with SYNC0 Event 3: DC-Mode - Synchron with SYNC1 Event 	UINT16	RW	0x0000 (0 _{dez})
1C32:02	Cycle time	Zykluszeit (in ns): <ul style="list-style-type: none"> Free Run: Zykluszeit des lokalen Timers Synchron with SM 2 Event: Zykluszeit des Masters DC-Mode: SYNC0/SYNC1 Cycle Time	UINT32	RW	0x00000000 (0 _{dez})
1C32:03	Shift time	Zeit zwischen SYNC0 Event und Ausgabe der Outputs (in ns, nur DC-Mode)	UINT32	RO	0x00000000 (0 _{dez})
1C32:04	Sync modes supported	Unterstützte Synchronisierungsbetriebsarten: <ul style="list-style-type: none"> Bit 0 = 1: Free Run wird unterstützt 	UINT16	RO	0x4001 (16385 _{dez})

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
		<ul style="list-style-type: none"> • Bit 1 = 1: Synchron with SM 2 Event wird unterstützt • Bit 2-3 = 01: DC-Mode wird unterstützt • Bit 4-5 = 10: Output Shift mit SYNC1 Event (nur DC-Mode) Bit 14 = 1: dynamische Zeiten (Messen durch Beschreiben von 1C32:08)			
1C32:05	Minimum cycle time	Minimale Zykluszeit (in ns)	UINT32	RO	0x00000000 (0 _{dez})
1C32:06	Calc and copy time	Minimale Zeit zwischen SYNC0 und SYNC1 Event (in ns, nur DC-Mode)	UINT32	RO	0x00000000 (0 _{dez})
1C32:07	Minimum delay time		UINT32	RO	0x00000000 (0 _{dez})
1C32:08	Get Cycle Time	<ul style="list-style-type: none"> • 0: Messung der lokalen Zykluszeit wird gestoppt • 1: Messung der lokalen Zykluszeit wird gestartet Die Entries 1C32:03 , 1C32:05 , 1C32:06 , 1C32:09 , 1C33:03 , 1C33:06 , 1C33:09 werden mit den maximal gemessenen Werten aktualisiert. Wenn erneut gemessen wird, werden die Messwerte zurückgesetzt	UINT16	RW	0x0000 (0 _{dez})
1C32:09	Maximum delay time	Zeit zwischen SYNC1 Event und Ausgabe der Outputs (in ns, nur DC-Mode)	UINT32	RO	0x00000000 (0 _{dez})
1C32:0B	SM event missed counter	Anzahl der ausgefallenen SM-Events im OPERATIONAL (nur im DC Mode)	UINT16	RO	0x0000 (0 _{dez})
1C32:0C	Cycle exceeded counter	Anzahl der Zykluszeitverletzungen im OPERATIONAL (Zyklus wurde nicht rechtzeitig fertig bzw. der nächste Zyklus kam zu früh)	UINT16	RO	0x0000 (0 _{dez})
1C32:0D	Shift too short counter	Anzahl der zu kurzen Abstände zwischen SYNC0 und SYNC1 Event (nur im DC Mode)	UINT16	RO	0x0000 (0 _{dez})
1C32:20	Sync error	Im letzten Zyklus war die Synchronisierung nicht korrekt (Ausgänge wurden zu spät ausgegeben, nur im DC Mode)	BOOLEAN	RO	0x00 (0 _{dez})

Index 1C33 SM input parameter

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1C33:0	SM input parameter	Synchronisierungsparameter der Inputs	UINT8	RO	0x20 (32 _{dez})
1C33:01	Sync mode	Aktuelle Synchronisierungsbetriebsart: <ul style="list-style-type: none"> • 0: Free Run • 1: Synchron with SM 3 Event (keine Outputs vorhanden) • 2: DC - Synchron with SYNC0 Event • 3: DC - Synchron with SYNC1 Event 34: Synchron with SM 2 Event (Outputs vorhanden) 	UINT16	RW	0x0000 (0 _{dez})
1C33:02	Cycle time	wie 1C32:02	UINT32	RW	0x00000000 (0 _{dez})
1C33:03	Shift time	Zeit zwischen SYNC0-Event und Einlesen der Inputs (in ns, nur DC-Mode)	UINT32	RO	0x00000000 (0 _{dez})
1C33:04	Sync modes supported	Unterstützte Synchronisierungsbetriebsarten: <ul style="list-style-type: none"> • Bit 0: Free Run wird unterstützt • Bit 1: Synchron with SM 2 Event wird unterstützt (Outputs vorhanden) • Bit 1: Synchron with SM 3 Event wird unterstützt (keine Outputs vorhanden) • Bit 2-3 = 01: DC-Mode wird unterstützt • Bit 4-5 = 01: Input Shift durch lokales Ereignis (Outputs vorhanden) • Bit 4-5 = 10: Input Shift mit SYNC1 Event (keine Outputs vorhanden) Bit 14 = 1: dynamische Zeiten (Messen durch Beschreiben von 1C32:08 oder 1C33:08)	UINT16	RO	0x4001 (16385 _{dez})
1C33:05	Minimum cycle time	wie 1C32:05	UINT32	RO	0x00000000 (0 _{dez})

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1C33:06	Calc and copy time	Zeit zwischen Einlesen der Eingänge und Verfügbarkeit der Eingänge für den Master (in ns, nur DC-Mode)	UINT32	RO	0x00000000 (0 _{dez})
1C33:07	Minimum delay time		UINT32	RO	0x00000000 (0 _{dez})
1C33:08	Get Cycle Time	wie 1C32:08	UINT16	RW	0x0000 (0 _{dez})
1C33:09	Maximum delay time	Zeit zwischen SYNC1-Event und Einlesen der Eingänge (in ns, nur DC-Mode)	UINT32	RO	0x00000000 (0 _{dez})
1C33:0B	SM event missed counter	wie 1C32:11	UINT16	RO	0x0000 (0 _{dez})
1C33:0C	Cycle exceeded counter	wie 1C32:12	UINT16	RO	0x0000 (0 _{dez})
1C33:0D	Shift too short counter	wie 1C32:13	UINT16	RO	0x0000 (0 _{dez})
1C33:20	Sync error	wie 1C32:32	BOOLEAN	RO	0x00 (0 _{dez})

Index 1C40-1C7E DNS TxPDO-Map Slave 36

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1C40+n:0	DNS TxPDO-Map Slave 36	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1C40+n):01					
...					
(1C40+n):F					

Index 1C50-1C8E DNS TxPDO-Map Slave 37

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1C50+n:0	DNS TxPDO-Map Slave 37	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1C50+n):01					
...					
(1C50+n):F					

Index 1C60-1C9E DNS TxPDO-Map Slave 38

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1C60+n:0	DNS TxPDO-Map Slave 38	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1C60+n):01					
...					
(1C60+n):F					

Index 1C70-1CAE DNS TxPDO-Map Slave 39

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1C70+n:0	DNS TxPDO-Map Slave 39	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1C70+n):01					
...					
(1C70+n):F					

Index 1C80-1CBE DNS TxPDO-Map Slave 40

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1C80+n:0	DNS TxPDO-Map Slave 40	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1C80+n):01					
...					
(1C80+n):F					

Index 1C90-1CCE DNS TxPDO-Map Slave 41

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1C90+n:0	DNS TxPDO-Map Slave 41	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1C90+n):01					
...					
(1C90+n):F					

Index 1CA0-1CDE DNS TxPDO-Map Slave 42

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1CA0+n:0	DNS TxPDO-Map Slave 42	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1CA0+n):01					
...					
(1CA0+n):F					

Index 1CB0-1CEE DNS TxPDO-Map Slave 43

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1CB0+n:0	DNS TxPDO-Map Slave 43	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1CB0+n):01					
...					
(1CB0+n):F					

Index 1CC0-1CFE DNS TxPDO-Map Slave 44

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1CC0+n:0	DNS TxPDO-Map Slave 44	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1CC0+n):01					
...					
(1CC0+n):F					

Index 1CD0-1D0E DNS TxPDO-Map Slave 45

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1CD0+n:0	DNS TxPDO-Map Slave 45	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1CD0+n):01					
...					
(1CD0+n):F					

Index 1CE0-1D1E DNS TxPDO-Map Slave 46

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1CE0+n:0	DNS TxPDO-Map Slave 46	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1CE0+n):01					
...					
(1CE0+n):F					

Index 1CF0-1D2E DNS TxPDO-Map Slave 47

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1CF0+n:0	DNS TxPDO-Map Slave 47	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1CF0+n):01					
...					
(1CF0+n):F					

Index 1D00-1D3E DNS TxPDO-Map Slave 48

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1D00+n:0	DNS TxPDO-Map Slave 48	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1D00+n):01					
...					
(1D00+n):F					

Index 1D10-1D4E DNS TxPDO-Map Slave 49

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1D10+n:0	DNS TxPDO-Map Slave 49	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1D10+n):01					
...					
(1D10+n):F					

Index 1D20-1D5E DNS TxPDO-Map Slave 50

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1D20+n:0	DNS TxPDO-Map Slave 50	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1D20+n):01					
...					
(1D20+n):F					

Index 1D30-1D6E DNS TxPDO-Map Slave 51

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1D30+n:0	DNS TxPDO-Map Slave 51	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1D30+n):01					
...					
(1D30+n):F					

Index 1D40-1D7E DNS TxPDO-Map Slave 52

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1D40+n:0	DNS TxPDO-Map Slave 52	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1D40+n):01					
...					
(1D40+n):F					

Index 1D50-1D8E DNS TxPDO-Map Slave 53

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1D50+n:0	DNS TxPDO-Map Slave 53	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
(1D50+n):01					
...					
(1D50+n):F F					

Index 1D60-1D9E DNS TxPDO-Map Slave 54

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1D60+n:0	DNS TxPDO-Map Slave 54	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1D60+n):01					
...					
(1D60+n):F F					

Index 1D70-1DAE DNS TxPDO-Map Slave 55

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1D70+n:0	DNS TxPDO-Map Slave 55	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1D70+n):01					
...					
(1D70+n):F F					

Index 1D80-1DBE DNS TxPDO-Map Slave 56

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1D80+n:0	DNS TxPDO-Map Slave 56	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1D80+n):01					
...					
(1D80+n):F F					

Index 1D90-1DCE DNS TxPDO-Map Slave 57

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1D90+n:0	DNS TxPDO-Map Slave 57	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1D90+n):01					
...					
(1D90+n):F F					

Index 1DA0-1DDE DNS TxPDO-Map Slave 58

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1DA0+n:0	DNS TxPDO-Map Slave 58	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1DA0+n):0 1					
...					
(1DA0+n):F F					

Index 1DB0-1DEE DNS TxPDO-Map Slave 59

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1DB0+n:0	DNS TxPDO-Map Slave 59	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1DB0+n):0 1					
...					

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
(1DB0+n):F F					

Index 1DC0-1DFE DNS TxPDO-Map Slave 60

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1DC0+n:0	DNS TxPDO-Map Slave 60	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1DC0+n):0 1					
...					
(1DC0+n):F F					

Index 1DD0-1E0E DNS TxPDO-Map Slave 61

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1DD0+n:0	DNS TxPDO-Map Slave 61	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1DD0+n):0 1					
...					
(1DD0+n):F F					

Index 1DE0-1E1E DNS TxPDO-Map Slave 62

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1DE0+n:0	DNS TxPDO-Map Slave 62	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RW	0x00 (0 _{dez})
(1DE0+n):0 1					
...					
(1DE0+n):F F					

Index F008 Code word

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F008:0	Code word	reserviert	UINT32	RW	0x00000000 (0 _{dez})

6.7.2.1.2.2 Profilspezifische Objekte (0x6000-0xFFFF)

Die profilspezifischen Objekte haben für alle EtherCAT Slaves, die das Profil 5001 unterstützen, die gleiche Bedeutung.

Index 6000-603E Produced Data Slave 1

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
6000+n*16:0	Produced Data Slave 1	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RO	0x01 (1 _{dez})
(6000+n*16):01					
...					
(6000+n*16):FF					

Index 7000-703E Consumed Data Slave 1

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
7000+n*16:0	Consumed Data Slave 1	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RO	0x01 (1 _{dez})
(7000+n*16):01					
...					
(7000+n*16):FF					

Index A000-A03E Status Data Slave 1

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
A000+n*16:0	Status Data Slave 1	(je Modul ein Objekt (0 ≤ n < maximale Anzahl Module))	UINT8	RO	0x02 (2 _{dez})
(A000+n*16):01					
...					
(A000+n*16):FF					

Index F000 Modular device profile

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F000:0	Modular Device Profile	Allgemeine Informationen des Modular Device Profiles	UINT8	RO	0x02 (2 _{dez})
F000:01	Index distance	Indexabstand der Objekte der einzelnen Kanäle	UINT16	RO	0x0010 (16 _{dez})
F000:02	Maximum number of modules	Anzahl der Kanäle	UINT16	RO	0x003F (63 _{dez})
F000:03	General configuration		UINT32	RO	0x7FFFFFFF (2147483647 _{dez})

Index F010 Module list

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F010:0	Module list	Liste der an der EL6752 angeschlossenen DeviceNet Slaves.	UINT8	RW	0x00 (0 _{dez})
F010:01		Product-Code des ersten DeviceNet-Slaves	UINT16	RO	0x00 (0 _{dez})
...					
F010:3F					

Index F100 DeviceNet status

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F100:0	DeviceNet status	Max. Subindex	UINT8	RO	0x10 (16 _{dez})
F100:01	Communication status	erlaubte Werte:	UINT8	RO	0x00 (0 _{dez})
		0 No error			

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
		1			
		2			
		18			
		49			
F100:10	TxPdoState	Status der Tx-PDO	BOOLEAN	RO	0x00 (0 _{dez})

Index F101 Network status

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F101:0	Network status	Max. Subindex	UINT8	RO	0x11 (17 _{dez})
F101:01	Device status	0: RUN MODE 1: IDLE MODE 2: Duplicate MacId Check failed, MAC ID used 3: Status: Selftest 4: Status: Standby 5: Status:Major Recoverable Fault 6: Status:Minor Recoverable Fault 7: DeviceNet Voltage Error 8: DeviceNet Access Error	UINT8	RO	0x00 (0 _{dez})
F101:09	CAN BUS-OFF	CAN-Controller der EL6752 ist im Zustand Bus-Off	BOOLEAN	RO	0x00 (0 _{dez})
F101:0A	CAN warning limit	CAN-Controller der EL6752 hat das Warning Limit überschritten	BOOLEAN	RO	0x00 (0 _{dez})
F101:0B	CAN Overrun	CAN-Controller der EL6752 ist im Zustand Bus-Off	BOOLEAN	RO	0x00 (0 _{dez})
F101:11	CAN BUS load	CAN Busauslastung 0 - 100%	UINT16	RO	0x0000 (0 _{dez})

Index F102 Status - MacState

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F102:0	Status - MacState		UINT8	RO	0x01 (1 _{dez})
F102:01					
...					
F102:FF					

Index F103 Diag data

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F103:0	Diag data		UINT8	RO	0x01 (1 _{dez})
F103:01					
...					
F103:FF					

Index F200 BitStrobe Bits

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F200:0	BitStrobe Bits		UINT8	RO	0x40 (64 _{dez})
F200:01					
...					
F200:FF					

Index F800 Bus Parameter set

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F800:0	Bus Parameter set	Max. Subindex	UINT8	RW	0x0B (11 _{dez})
F800:01	MAC ID	Geräteadresse des DeviceNet Gerätes (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
F800:02	Baud rate	erlaubte Werte:	UINT16	RW	0x0100 (256 _{dez})
		0			
		1			
		2			
		3			
F800:03	Vendor ID	Herstellerkennung des DeviceNet Gerätes (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F800:04	DeviceType	Gerätetyp des DeviceNet Gerätes (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
F800:05	Product Code	Produktkennung des DeviceNet Gerätes (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
F800:06	Revision Number	Versionsnummer des DeviceNet Gerätes (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
F800:07	Serial Number	Seriennummer des DeviceNet Gerätes (siehe DeviceNet-Spezifikation)	UINT32	RW	0x00350000 (3473408 _{dez})
F800:08	Product Name	Produktname des DeviceNet Gerätes (siehe DeviceNet-Spezifikation)	OCTET-STRING[32]	RW	{0}
F800:09	Quick Connect	erlaubte Werte:	UINT16	RW	0x0000 (0 _{dez})
		0 Quick Connect Disabled			
		1 Quick Connect DEnabled			
F800:0A	Reserved	reserviert	UINT16	RW	0x0000 (0 _{dez})
F800:0B	Reserved2	reserviert	UINT16	RW	0x0000 (0 _{dez})

Index F880 Vendor data

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F880:0	Vendor data	Max. Subindex	UINT8	RO	0x01 (1 _{dez})
F880:01	Serial Number	Serial Number	UINT32	RW	0x00000000 (0 _{dez})

6.7.2.2 DeviceNet® Slave - EL6752-0010

i EtherCAT ESI Device Description (XML)

Die Darstellung entspricht der Anzeige der CoE-Objekte aus der EtherCAT ESI Device Description ([XML](#)). Es wird empfohlen, die entsprechende aktuellste XML-Datei im Download-Bereich auf der [Beckhoff-Website](#) herunterzuladen und entsprechend der Installationsanweisungen zu installieren.

i Parametrierung über das CoE-Verzeichnis (CAN over EtherCAT)

Die Parametrierung des EtherCAT-Gerätes wird über den CoE-Online Reiter (mit Doppelklick auf das entsprechende Objekt) bzw. über den Prozessdatenreiter (Zuordnung der PDOs) vorgenommen. Beachten Sie bei Verwendung/Manipulation der CoE-Parameter die allgemeinen [CoE-Hinweise](#) [[► 39](#)]:

- StartUp-Liste führen für den Austauschfall
 - Unterscheidung zwischen Online/Offline Dictionary, Vorhandensein aktueller XML-Beschreibung
 - „CoE-Reload“ zum Zurücksetzen der Veränderungen
-

Einführung

In der CoE-Übersicht sind Objekte mit verschiedenem Einsatzzweck enthalten:

- Objekte, die zu Parametrierung bei der Inbetriebnahme nötig sind
- Objekte, die interne Settings anzeigen und ggf. nicht veränderlich sind.

Im Folgenden werden zuerst die zur Parametrierung und zum normalen Betrieb benötigten Objekte vorgestellt. Alle weiteren Objekte, die für den normalen Anwendungsfall nicht erforderlich sind, sind im unteren Tabellenabschnitt zu finden.

6.7.2.2.1 Objekte für die Parametrierung

Index 8000 Configuration Data

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
8000:0	Configuration Data	Max. Subindex	UINT8	RW	0x33 (51 _{dez})
8000:01	MAC ID	DeviceNet Geräte Adresse (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
8000:02	ProductName	Produktname	OCTET-STRING[16]	RW	{0}
8000:03	Device Type	Gerätetyp (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
8000:04	Vendor ID	Herstellerkennung (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
8000:05	Product Code	Produktkennung (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
8000:06	Revision Number	Versionsnummer (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
8000:07	Serial Number	Seriennummer (siehe DeviceNet-Spezifikation)	UINT32	RW	0x00000000 (0 _{dez})
8000:08	Network Flags	reserviert für AMS über DeviceNet	UINT16	RW	0x0000 (0 _{dez})
8000:09	Network Port	reserviert für AMS über DeviceNet	UINT16	RW	0x0000 (0 _{dez})
8000:0A	Network Segment Address	reserviert für AMS über DeviceNet	OCTET-STRING[6]	RW	{0}
8000:0A	Reserved	reserviert	UINT16	RW	0x0000 (0 _{dez})
8000:0B	Electronic Key	Electronic Key Bit-Maske: Bit 0: Check Vendor Id Bit 1: Check DeviceType Bit 2: Check Product Code Bit 3: Check Revision Bit 4: reserved(0) Bit 5: reserved(0) Bit 6: reserved(0) Bit 7: reserved(0)	UINT16	RW	0x0100 (256 _{dez})
8000:0D	Allocation Choice	DeviceNet Betriebsarten-Auswahl (siehe DeviceNet-Spezifikation) Bit 0: reserved (0) Bit 1: Polled Bit 2: Bit-Strobed Bit 3: reserved (0) Bit 4: Change of State Bit 5: Cyclic Bit 6: Acknowledge Suppression Bit 7: reserved(0)	UINT16	RW	0x0100 (256 _{dez})
8000:0E	Expected Packet Rate - Poll	Timing-Parameter der Poll-Verbindung (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
8000:0F	Expected Packet Rate - Bit Strobe	Timing-Parameter der Bit-Strobe-Verbindung (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
8000:10	Expected Packet Rate - COS/Cyclic	Timing-Parameter der COS/Cyclic-Verbindung (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
8000:11	Produced Data Size - Poll	Datenlänge im Poll-Modus	UINT16	RW	0x0000 (0 _{dez})
8000:12	Produced Data Size - Bit Strobe	Datenlänge im Bit-Strobe-Modus	UINT16	RW	0x0000 (0 _{dez})
8000:13	Produced Data Size - COS/Cyclic	Datenlänge im Change of State / Cyclic Modus	UINT16	RW	0x0000 (0 _{dez})
8000:14	Reserved1	reserviert	UINT16	RW	0x0000 (0 _{dez})
8000:15	Reserved2	reserviert	UINT16	RW	0x0000 (0 _{dez})
8000:16	Reserved3	reserviert	UINT16	RW	0x0000 (0 _{dez})
8000:17	Consumed Data Size - Poll	Datenlänge im Poll-Modus	UINT16	RW	0x0000 (0 _{dez})
8000:18	Consumed Data Size - Bit Strobe	Datenlänge im Bit-Strobe-Modus	UINT16	RW	0x0000 (0 _{dez})
8000:19	Consumed Data Size - COS/Cyclic	Datenlänge im Change of State / Cyclic Modus	UINT16	RW	0x0000 (0 _{dez})

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
8000:1A	Reserved4	reserviert	UINT16	RW	0x0000 (0 _{dez})
8000:1B	Reserved5	reserviert	UINT16	RW	0x0000 (0 _{dez})
8000:1C	Reserved6	reserviert	UINT16	RW	0x0000 (0 _{dez})
8000:1D	Inhibit Time	Timing-Parameter der COS/Cyclic-Verbindung (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
8000:1E	Acknowledge Timer	Timing-Parameter der COS/Cyclic-Verbindung (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
8000:1F	Acknowledge Retry Limit	Timing-Parameter der COS/Cyclic-Verbindung (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
8000:20	Reserved7	reserviert	UINT16	RW	0x0000 (0 _{dez})
8000:21	Reserved8	reserviert	UINT16	RW	0x0000 (0 _{dez})

6.7.2.2.2 Objekte für interne Settings

6.7.2.2.2.1 Standardobjekte (0x1000-0x1FFF)

Die Standardobjekte haben für alle EtherCAT-Slaves die gleiche Bedeutung.

Index 1000 Device type

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1000:0	Device type	Geräte-Typ des EtherCAT-Slaves: Das Lo-Word enthält das verwendete CoE Profil (5001). Das Hi-Word enthält das Modul Profil entsprechend des Modular Device Profile.	UINT32	RO	0x145A1389 (341447561 _{dez})

Index 1008 Device name

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1008:0	Device name	Geräte-Name des EtherCAT-Slave	STRING	RO	EL6752-0010

Index 1009 Hardware version

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1009:0	Hardware version	Hardware-Version des EtherCAT-Slaves	STRING	RO	00

Index 100A Software version

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
100A:0	Software version	Firmware-Version des EtherCAT-Slaves	STRING	RO	00

Index 1018 Identity

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1018:0	Identity	Informationen, um den Slave zu identifizieren	UINT8	RO	0x04 (4 _{dez})
1018:01	Vendor ID	Hersteller-ID des EtherCAT-Slaves	UINT32	RO	0x00000002 (2 _{dez})
1018:02	Product code	Produkt-Code des EtherCAT-Slaves	UINT32	RO	0x1A603052 (442511442 _{dez})
1018:03	Revision	Revisionsnummer des EtherCAT-Slaves, das Low-Word (Bit 0-15) kennzeichnet die Sonderklemmennummer, das High-Word (Bit 16-31) verweist auf die Gerätebeschreibung	UINT32	RO	0x00000000 (0 _{dez})
1018:04	Serial number	Seriennummer des EtherCAT-Slaves, das Low-Byte (Bit 0-7) des Low-Words enthält das Produktionsjahr, das High-Byte (Bit 8-15) des Low-Words enthält die Produktionswoche, das High-Word (Bit 16-31) ist 0	UINT32	RO	0x00000000 (0 _{dez})

Index 10E2 Manufacturer-specific Identification Code

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
10E2:0	Manufacturer-specific Identification Code	Herstellerspezifischer Identifizierung Code	UINT8	RO	0x01 (1 _{dez})
10E2:01	SubIndex 001	reserviert	STRING	RO	

Index 1A10 DNM TxPDO-Map DeviceNet State

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1A10:0	DNM TxPDO-Map DeviceNet State	PDO Mapping TxPDO 17	UINT8	RO	0x06 (6 _{dez})
1A10:01	SubIndex 001	1. PDO Mapping entry (object 0xA000 (Status Data Slave), entry 0x01)	UINT32	RO	0xA000:01, 8
1A10:02	SubIndex 002	2. PDO Mapping entry (object 0xF100 (DeviceNet status), entry 0x01 (Communication status))	UINT32	RO	0xF100:01, 8
1A10:03	SubIndex 003	3. PDO Mapping entry (object 0xF100 (DeviceNet status), entry 0x10 (TxPdoState))	UINT32	RO	0xF100:10, 1

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1A10:04	SubIndex 004	4. PDO Mapping entry (object 0xF103 (Diag data), entry 0x01)	UINT32	RO	0xF103:01, 1
1A10:05	SubIndex 005	5. PDO Mapping entry (object 0xF600 (BitStrobe Bits), entry 0x01)	UINT32	RO	0xF600:01, 1
1A10:06	SubIndex 006	6. PDO Mapping entry (13 bits align)	UINT32	RO	0x0000:00, 13

Index 1C00 Sync manager type

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1C00:0	Sync manager type	Benutzung der Sync Manager	UINT8	RO	0x04 (4 _{dez})
1C00:01	SubIndex 001	Sync-Manager Type Channel 1: Mailbox Write	UINT8	RO	0x01 (1 _{dez})
1C00:02	SubIndex 002	Sync-Manager Type Channel 2: Mailbox Read	UINT8	RO	0x02 (2 _{dez})
1C00:03	SubIndex 003	Sync-Manager Type Channel 3: Process Data Write (Outputs)	UINT8	RO	0x03 (3 _{dez})
1C00:04	SubIndex 004	Sync-Manager Type Channel 4: Process Data Read (Inputs)	UINT8	RO	0x04 (4 _{dez})

Index 1C12 RxPDO assign

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RW	0x00 (0 _{dez})
1C12:01	SubIndex 001	1. zugeordnete RxPDO (enthält den Index des zugehörigen RxPDO Mapping Objekts)	UINT16	RW	0x0000 (0 _{dez})

Index 1C13 TxPDO assign

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RW	0x02 (2 _{dez})
1C13:01	SubIndex 001	1. zugeordnete TxPDO (enthält den Index des zugehörigen TxPDO Mapping Objekts)	UINT16	RW	0x1A10 (6672 _{dez})
1C13:02	SubIndex 002	2. zugeordnete TxPDO (enthält den Index des zugehörigen TxPDO Mapping Objekts)	UINT16	RW	0x0000 (0 _{dez})

Index 1C32 SM output parameter

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1C32:0	SM output parameter	Synchronisierungsparameter der Outputs	UINT8	RO	0x20 (32 _{dez})
1C32:01	Sync mode	Aktuelle Synchronisierungsbetriebsart: <ul style="list-style-type: none"> 0: Free Run 1: Synchron with SM 2 Event 2: DC-Mode - Synchron with SYNC0 Event 3: DC-Mode - Synchron with SYNC1 Event 	UINT16	RW	0x0000 (0 _{dez})
1C32:02	Cycle time	Zykluszeit (in ns): <ul style="list-style-type: none"> Free Run: Zykluszeit des lokalen Timers Synchron with SM 2 Event: Zykluszeit des Masters DC-Mode: SYNC0/SYNC1 Cycle Time	UINT32	RW	0x00000000 (0 _{dez})
1C32:03	Shift time	Zeit zwischen SYNC0 Event und Ausgabe der Outputs (in ns, nur DC-Mode)	UINT32	RO	0x00000000 (0 _{dez})
1C32:04	Sync modes supported	Unterstützte Synchronisierungsbetriebsarten: <ul style="list-style-type: none"> Bit 0 = 1: Free Run wird unterstützt Bit 1 = 1: Synchron with SM 2 Event wird unterstützt Bit 2-3 = 01: DC-Mode wird unterstützt Bit 4-5 = 10: Output Shift mit SYNC1 Event (nur DC-Mode) Bit 14 = 1: dynamische Zeiten (Messen durch Beschreiben von 1C32:08)	UINT16	RO	0x4001 (16385 _{dez})
1C32:05	Minimum cycle time	Minimale Zykluszeit (in ns)	UINT32	RO	0x00000000 (0 _{dez})
1C32:06	Calc and copy time	Minimale Zeit zwischen SYNC0 und SYNC1 Event (in ns, nur DC-Mode)	UINT32	RO	0x00000000 (0 _{dez})
1C32:07	Minimum delay time		UINT32	RO	0x00000000 (0 _{dez})

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1C32:08	Get Cycle Time	<ul style="list-style-type: none"> 0: Messung der lokalen Zykluszeit wird gestoppt 1: Messung der lokalen Zykluszeit wird gestartet <p>Die Entries 1C32:03, 1C32:05, 1C32:06, 1C32:09, 1C33:03, 1C33:06, 1C33:09 werden mit den maximal gemessenen Werten aktualisiert. Wenn erneut gemessen wird, werden die Messwerte zurückgesetzt</p>	UINT16	RW	0x0000 (0 _{dez})
1C32:09	Maximum delay time	Zeit zwischen SYNC1 Event und Ausgabe der Outputs (in ns, nur DC-Mode)	UINT32	RO	0x00000000 (0 _{dez})
1C32:0B	SM event missed counter	Anzahl der ausgefallenen SM-Events im OPERATIONAL (nur im DC Mode)	UINT16	RO	0x0000 (0 _{dez})
1C32:0C	Cycle exceeded counter	Anzahl der Zykluszeitverletzungen im OPERATIONAL (Zyklus wurde nicht rechtzeitig fertig bzw. der nächste Zyklus kam zu früh)	UINT16	RO	0x0000 (0 _{dez})
1C32:0D	Shift too short counter	Anzahl der zu kurzen Abstände zwischen SYNC0 und SYNC1 Event (nur im DC Mode)	UINT16	RO	0x0000 (0 _{dez})
1C32:20	Sync error	Im letzten Zyklus war die Synchronisierung nicht korrekt (Ausgänge wurden zu spät ausgegeben, nur im DC Mode)	BOOLEAN	RO	0x00 (0 _{dez})

Index 1C33 SM input parameter

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1C33:0	SM input parameter	Synchronisierungsparameter der Inputs	UINT8	RO	0x20 (32 _{dez})
1C33:01	Sync mode	<p>Aktuelle Synchronisierungsbetriebsart:</p> <ul style="list-style-type: none"> 0: Free Run 1: Synchron with SM 3 Event (keine Outputs vorhanden) 2: DC - Synchron with SYNC0 Event 3: DC - Synchron with SYNC1 Event 34: Synchron with SM 2 Event (Outputs vorhanden) 	UINT16	RW	0x0000 (0 _{dez})
1C33:02	Cycle time	wie 1C32:02	UINT32	RW	0x00000000 (0 _{dez})
1C33:03	Shift time	Zeit zwischen SYNC0-Event und Einlesen der Inputs (in ns, nur DC-Mode)	UINT32	RO	0x00000000 (0 _{dez})
1C33:04	Sync modes supported	<p>Unterstützte Synchronisierungsbetriebsarten:</p> <ul style="list-style-type: none"> Bit 0: Free Run wird unterstützt Bit 1: Synchron with SM 2 Event wird unterstützt (Outputs vorhanden) Bit 1: Synchron with SM 3 Event wird unterstützt (keine Outputs vorhanden) Bit 2-3 = 01: DC-Mode wird unterstützt Bit 4-5 = 01: Input Shift durch lokales Ereignis (Outputs vorhanden) Bit 4-5 = 10: Input Shift mit SYNC1 Event (keine Outputs vorhanden) <p>Bit 14 = 1: dynamische Zeiten (Messen durch Beschreiben von 1C32:08 oder 1C33:08)</p>	UINT16	RO	0x4001 (16385 _{dez})
1C33:05	Minimum cycle time	wie 1C32:05	UINT32	RO	0x00000000 (0 _{dez})
1C33:06	Calc and copy time	Zeit zwischen Einlesen der Eingänge und Verfügbarkeit der Eingänge für den Master (in ns, nur DC-Mode)	UINT32	RO	0x00000000 (0 _{dez})
1C33:07	Minimum delay time		UINT32	RO	0x00000000 (0 _{dez})
1C33:08	Get Cycle Time	wie 1C32:08	UINT16	RW	0x0000 (0 _{dez})
1C33:09	Maximum delay time	Zeit zwischen SYNC1-Event und Einlesen der Eingänge (in ns, nur DC-Mode)	UINT32	RO	0x00000000 (0 _{dez})
1C33:0B	SM event missed counter	wie 1C32:11	UINT16	RO	0x0000 (0 _{dez})
1C33:0C	Cycle exceeded counter	wie 1C32:12	UINT16	RO	0x0000 (0 _{dez})
1C33:0D	Shift too short counter	wie 1C32:13	UINT16	RO	0x0000 (0 _{dez})

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
1C33:20	Sync error	wie <u>1C32:32</u>	BOOLEAN	RO	0x00 (0 _{dez})

6.7.2.2.2 Profilspezifische Objekte (0x6000-0xFFFF)

Die profilspezifischen Objekte haben für alle EtherCAT Slaves, die das Profil 5001 unterstützen, die gleiche Bedeutung.

Index A000 Status Data Slave

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
A000	Status Data Slave	max. Subindex	UINT8	RO	0x02 (2 _{dez})
A000:01					
...					
A000:FF					

Index F000 Modular device profile

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F000:0	Modular device profile	Allgemeine Informationen des Modular Device Profiles	UINT8	RO	0x02 (2 _{dez})
F000:01	Module index distance	Indexabstand der Objekte der einzelnen Kanäle	UINT16	RO	0x0010 (16 _{dez})
F000:02	Maximum number of modules	Anzahl der Kanäle	UINT16	RO	0x0001 (1 _{dez})

Index F008 Code word

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F008:0	Code word	reserved	UINT32	RW	0x00000000 (0 _{dez})

Index F100 DeviceNet status

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F100:0	DeviceNet status	DeviceNet-seitiger Status der EL6752-0010	UINT8	RO	0x10 (16 _{dez})
F100:01	Communication status	Kommunikationszustand der EL6752-0010: 0 = No error 1 = Station deactivated 2 = Station not exists 18 = Station ready 31 = only for EtherCAT gateways: WC-State of cyclic EtherCAT frame is 1	UINT8	RO	0x00 (0 _{dez})
F100:10	TxPdoState	Status der Tx-PDO	BOOLEAN	RO	0x00 (0 _{dez})

Index F101 Network status

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F101:0	Network status	Max. Subindex	UINT8	RO	0x11 (17 _{dez})
F101:01	Device status	0: RUN MODE 1: IDLE MODE 2: Duplicate MacId Check failed, MAC ID used 3: Status: Selftest 4: Status: Standby 5: Status:Major Recoverable Fault 6: Status:Minor Recoverable Fault 7: DeviceNet Voltage Error 8: DeviceNet Access Error	UINT8	RO	0x00 (0 _{dez})
F101:09	CAN BUS-OFF	CAN-Controller der EL6752-0010 ist im Zustand Bus-Off	BOOLEAN	RO	0x00 (0 _{dez})
F101:0A	CAN warning limit	CAN-Controller der EL6752-0010 hat das Warning Limit überschritten	BOOLEAN	RO	0x00 (0 _{dez})
F101:0B	CAN Overrun	CAN-Controller der EL6752-0010 ist im Zustand Bus-Off	BOOLEAN	RO	0x00 (0 _{dez})
F101:11	CAN BUS load	CAN Busauslastung 0 - 100%	UINT16	RO	0x0000 (0 _{dez})

Index F102 Status - MacState

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F102:0	Status - MacState		UINT8	RO	0x01 (1 _{dez})
F102:01					
...					

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F102:FF					

Index F103 Diag data

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F103:0	Diag data		UINT8	RO	0x01 (1 _{dez})
F103:01					
...					
F103:FF					

Index F600 BitStrobe Bits

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F600:0	BitStrobe Bits		UINT8	RO	0x01 (01 _{dez})
F600:01					
...					
F600:FF					

Index F800 Bus Parameter set

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F800:0	Bus Parameter set	Max. Subindex	UINT8	RW	0x0B (11 _{dez})
F800:01	MAC ID	Geräteadresse des DeviceNet Gerätes (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
F800:02	Baud rate	erlaubte Werte:	UINT16	RW	0x0100 (256 _{dez})
		0 invalid			
		1 500 KBAud			
		2 250 KBAud			
		3 125 KBAud			
F800:03	Vendor ID	Herstellerkennung des DeviceNet Gerätes (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
F800:04	DeviceType	Gerätetyp des DeviceNet Gerätes (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
F800:05	Product Code	Produktkennung des DeviceNet Gerätes (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
F800:06	Revision Number	Versionsnummer des DeviceNet Gerätes (siehe DeviceNet-Spezifikation)	UINT16	RW	0x0000 (0 _{dez})
F800:07	Serial Number	Seriennummer des DeviceNet Gerätes (siehe DeviceNet-Spezifikation)	UINT32	RW	0x00350000 (3473408 _{dez})
F800:08	Product Name	Produktname des DeviceNet Gerätes (siehe DeviceNet-Spezifikation)	OCTET-STRING[32]	RW	{0}
F800:09	Quick Connect	erlaubte Werte:	UINT16	RW	0x0000 (0 _{dez})
		0 Quick Connect Disabled			
		1 Quick Connect DEnabled			
F800:0A	Reserved	reserviert	UINT16	RW	0x0000 (0 _{dez})
F800:0B	Reserved2	reserviert	UINT16	RW	0x0000 (0 _{dez})

Index F880 Vendor data

Index (hex)	Name	Bedeutung	Datentyp	Flags	Default
F880:0	Vendor data	Max. Subindex	UINT8	RO	0x01 (1 _{dez})
F880:01	Serial Number	Serial Number	UINT32	RW	0x00000000 (0 _{dez})

7 Fehlerbehandlung und Diagnose

7.1 EL6752 - LED Beschreibung

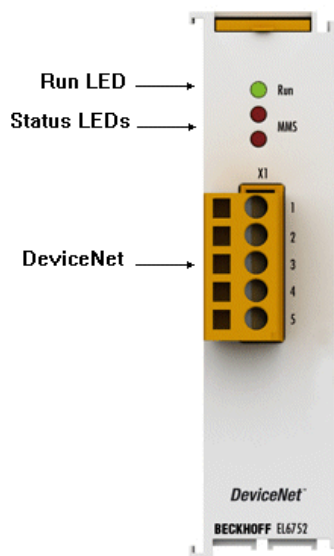


Abb. 68: LEDs

LED Verhalten

Anhand der LED's lassen sich die wichtigsten Zustände der Klemme schnell diagnostizieren:

EL6752-0000 (DeviceNet® Master Klemme)

LED	Farbe	Bedeutung
RUN	grün	Diese LED gibt den Betriebszustand der Klemme wieder:
		aus Zustand der EtherCAT State Machine: INIT = Initialisierung der Klemme; BOOTSTRAP = Funktion für Firmware-Updates der Klemme
		blinkend Zustand der EtherCAT State Machine: PREOP = Funktion für Mailbox-Kommunikation und abweichende Standard-Einstellungen gesetzt
		Einzelblitz Zustand der EtherCAT State Machine: SAFEOP = Überprüfung der Kanäle des Sync-Managers und der Distributed Clocks. Ausgänge bleiben im sicheren Zustand
MNS grün	grün	an Zustand der EtherCAT State Machine: OP = normaler Betriebszustand; Mailbox- und Prozessdatenkommunikation ist möglich
		aus Master ist offline
		blinkend Master ist online und vollzieht den Duplicate MAC-ID Check
MNS rot	rot	an Master ist online und kommuniziert mit den konfigurierten Slaves
		blinkend Kommunikationsstörung des Masters mit einem der konfigurierten Slaves
		an DeviceNet Bus OFF, DeviceNet Voltage Error, Master failed Duplicate MAC-ID Check

EL6752-0010 (DeviceNet® Slave Klemme)

LED	Color	Meaning
RUN	grün	Diese LED gibt den Betriebszustand der Klemme wieder:
		aus Zustand der EtherCAT State Machine: INIT = Initialisierung der Klemme; BOOTSTRAP = Funktion für Firmware-Updates der Klemme
		blinkend Zustand der EtherCAT State Machine: PREOP = Funktion für Mailbox-Kommunikation und abweichende Standard-Einstellungen gesetzt

LED	Color	Meaning	
		Einzelblitz	Zustand der EtherCAT State Machine: SAFEOP = Überprüfung der Kanäle des Sync-Managers und der Distributed Clocks. Ausgänge bleiben im sicheren Zustand
		an	Zustand der EtherCAT State Machine: OP = normaler Betriebszustand; Mailbox- und Prozessdatenkommunikation ist möglich
MNS grün	grün	aus	Slave ist offline
		blinkend	Slave Port hat den Duplicate MAC-ID Check beendet (Netzwerk OK), Kommunikationsstörung mit dem Master.
		an	Slave Port ist online und kommuniziert mit den Master.
MNS rot	rot	blinkend	Kommunikationsstörung des Slave Port mit dem Master, Timeout des Slave Port
		an	DeviceNet Bus OFF, DeviceNet Voltage Error, Slave Port Fehler, Fehler Duplicate MAC-ID Check

7.2 EL6752/-0010 Diagnose

Die EL6752/-0010 verfügen über verschiedene Diagnosevariablen, die den Zustand der Klemme sowie des DeviceNet® beschreiben und die in der SPS verknüpft werden können:



Es wird empfohlen, folgende Prozessdaten in jedem Zyklus zu überwachen:

- **WcState:** wenn $\neq 0$, dieser EtherCAT Teilnehmer nimmt nicht am Prozessdatenverkehr teil
- **State:** wenn $\neq 8$, der EtherCAT Teilnehmer ist nicht im OP (Operational) Status



Es wird empfohlen, folgende Prozessdaten zu überwachen:

- **Error:** wenn $\neq 0$, die angezeigte Anzahl DeviceNet® Geräte hat einen BoxState ungleich Null, d.h. es muss geprüft werden welche DeviceNet® Geräte nicht korrekt am Bus arbeiten
- **DiagFlag:** zeigt eine anstehende Diagnose an

7.2.1 EL6752/-0010 - WC-State

Zur Überwachung der EtherCAT Kommunikation muss der WC-State (Working-Counter) der EL6752/-0010 überprüft werden. Bei einem WC-State ungleich "0" ist die EtherCAT-Kommunikation gestört, d.h. Daten die zum Slave bzw. zum Master gesendet werden, werden nicht mehr korrekt übertragen und sind nicht gültig.

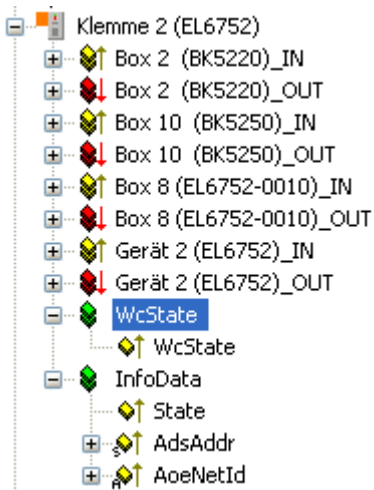


Abb. 69: WCState im TwinCAT Baum

WcState

0: Daten sind gültig

1: Daten sind nicht gültig, es liegt ein Problem in der EtherCAT Kommunikation vor

7.2.2 EL6752/-0010 - State

Die State-Diagnose Variable gibt den aktuellen EtherCAT-Zustand der EL6752/-0010 wieder.

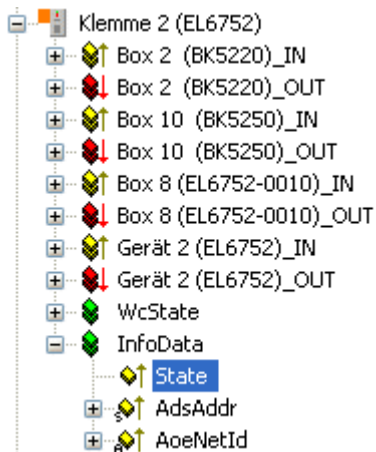


Abb. 70: State Diagnose Variable im TwinCAT Baum

State

- 0x___1 = Slave in 'INIT' state
- 0x___2 = Slave in 'PREOP' state
- 0x___3 = Slave in 'BOOT' state
- 0x___4 = Slave in 'SAFEOP' state
- 0x___8 = Slave in 'OP' state
- 0x001_ = Slave signals error
- 0x002_ = Invalid vendorId, productCode... read
- 0x004_ = Initialization error occurred
- 0x010_ = Slave not present
- 0x020_ = Slave signals link error
- 0x040_ = Slave signals missing link
- 0x080_ = Slave signals unexpected link
- 0x100_ = Communication port A
- 0x200_ = Communication port B
- 0x400_ = Communication port C
- 0x800_ = Communication port D

7.2.3 EL6752/-0010 - Error / DiagFlag

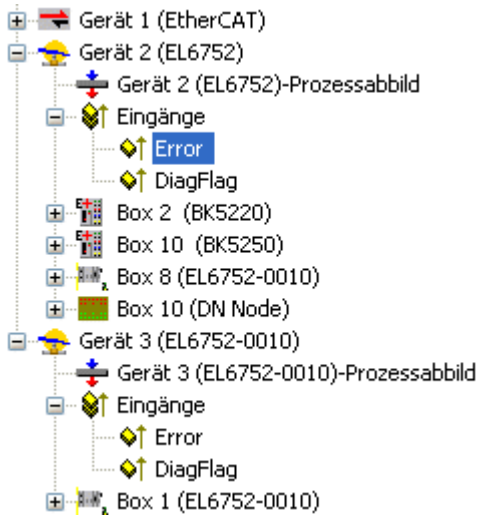


Abb. 71: Error und DiagFlag im TwinCAT Baum

Error

0: alle DeviceNet® Geräte haben BoxState gleich Null
 >0: Anzahl der DeviceNet®-Geräte mit BoxState ungleich Null.

DiagFlag

0 = es liegen keine Diagnose-Daten an
 1 = es liegen Diagnose-Daten vor, diese können über ADSRead-Dienste ausgelesen werden

7.3 DeviceNet® Geräte Diagnose

DeviceNet® Slave Geräte verfügen über verschiedene Diagnosevariablen die den DeviceNet®-Kommunikationszustand beschreiben und die in der SPS verknüpft werden können:



Es wird empfohlen, folgende Prozessdaten in jedem Zyklus zu überwachen:

- **MacState:** wenn $\neq 0$, dieser DeviceNet® Teilnehmer nimmt nicht korrekt am Prozessdatenverkehr teil
- **CouplerState:** für Beckhoff Buskoppler wenn $\neq 0$, ist die Klemmenkommunikation des Buskopplers evtl. gestört oder es liegen Diagnose-Daten an

7.3.1 DeviceNet® Slave Gerät / EL6752-0010 - MacState

Zur Überwachung der DeviceNet Kommunikation muss der MacState der DeviceNet Teilnehmer / EL6752-0010 überprüft werden. Ist der MacState ungleich Null, nimmt der DeviceNet Slave nicht korrekt am DeviceNet Datenaustausch teil.

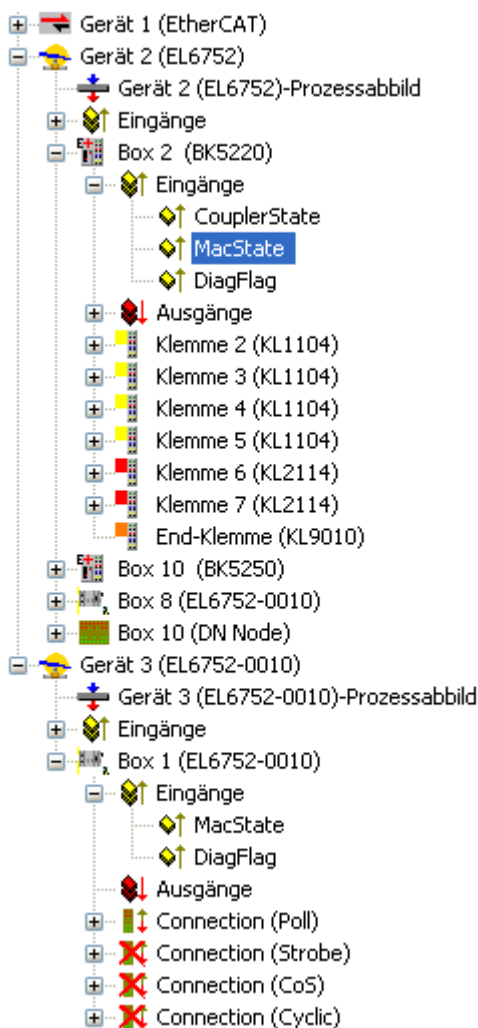


Abb. 72: MacState im TwinCAT Baum

MacState

- 0 = No error
- 1 = Station deactivated
- 2 = Station not exists
- 18 = Station ready
- 40 = Heartbeat Message not received
- 41 = Shutdown Message received
- 42 = Electronic Key Fault: Vendor Id

43 = Electronic Key Fault: Device Type
44 = Electronic Key Fault: Product Code
45 = Electronic Key Fault: Revision
46 = Fault while writing Start-Up Attributes
47 = wrong Produced IO-Data Size
48 = wrong Consumed IO-Data Size
49 = Idle Mode

7.3.2 DeviceNet® Slave Gerät / EL6752-0010 - DiagFlag

Das DiagFlag zeigt eine anstehende Diagnose an. Liegen Diagnosedaten an, so können diese über ein ADSRead Kommando ausgelesen werden.

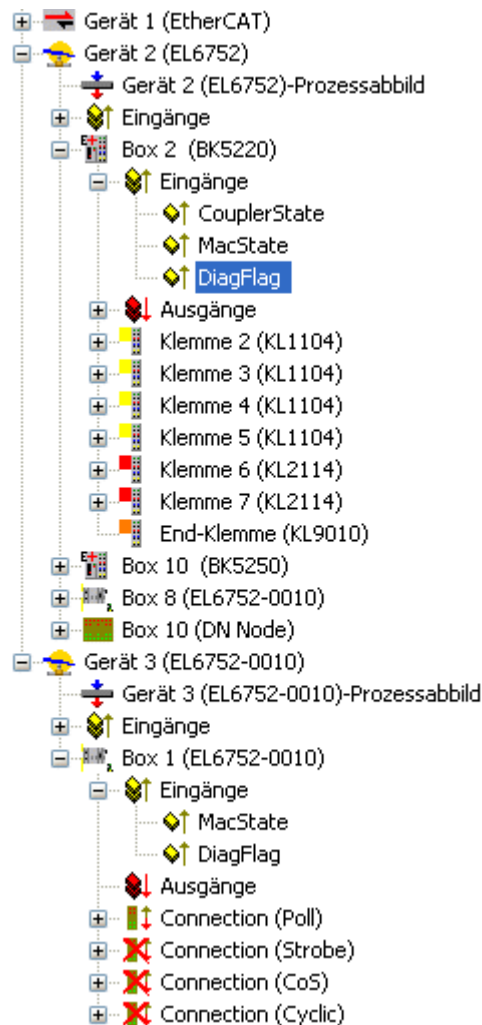


Abb. 73: DiagFlag im TwinCAT Baum

DiagFlag

0 = es liegen keine Diagnose-Daten an

1 = es liegen Diagnose-Daten vor, diese können über ADSRead-Dienste ausgelesen werden

7.3.3 Beckhoff DeviceNet® Slave Gerät - CouplerState

Der CouplerState gibt Information über die Klemmenbus-Kommunikation des Beckhoff-Buskoppler wieder. Diese Information steht für Beckhoff Buskoppler BK52x0, Geräte der IP-Box-Familie IPxxxx-B520 und der IP-Link-Familie zur Verfügung.

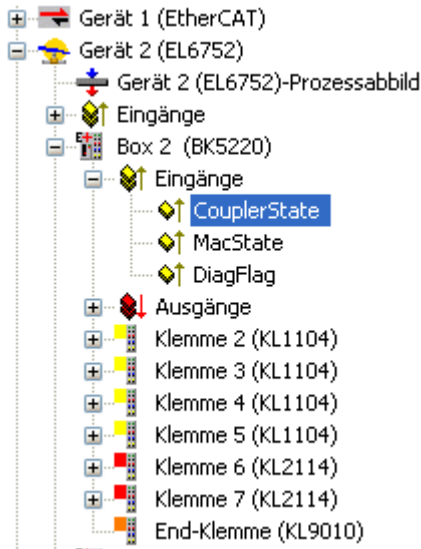


Abb. 74: CouplerState im TwinCAT Baum

CouplerState

0x00 = I/O Run

0x01 = I/O Error (KBus, IO or Terminal Error)

0x80 = I/O Idle-Mode / Fieldbus Error, es werden keine Ausgangsdaten geschrieben

0x08= es liegt eine Diagnose-Information einer Analogen/- Sonderfunktionsklemme an. Diese Funktion muss zuvor an den Kopplern aktiviert werden. Die Diagnose-Daten können dann in den entsprechenden Registern der Klemmen, IP/-IL-Module ausgelesen werden

7.4 EL6752/-0010 - ADS Error Codes

Die ADS Error Codes haben folgende Bedeutung:

Error	Beschreibung
Fehler bei der ADS/AMS - Datenaustausch	
0x1001	nicht genügend Speicher für AMS-Kommando
0x1101	falsche Datenlänge bei StartFieldbus
0x1102	falscher DeviceState bei StartFieldbus
0x1103	Device kann nicht von INIT nach RUN wechseln
0x1104	falscher AdsState im Zustand INIT
0x1105	falscher DeviceState bei StopFieldbus
0x1106	Device kann nicht STOP nach RUN wechseln, wenn keine CDL definiert ist
0x1107	Device kann nicht STOP nach RUN wechseln, wenn keine Box definiert ist
0x1108	falsche Datenlänge bei StartDataTransfer
0x1109	falscher DeviceState bei StartDataTransfer
0x110A	falscher AdsState im Zustand STOP
0x110B	Device kann nicht von RUN nach INIT wechseln
0x110C	falsche Datenlänge bei StopDataTransfer
0x110D	falscher DeviceState bei StopDataTransfer
0x1110	falscher AdsState im Zustand RUN
0x1111	Laden der Device-Parameter nur im Zustand INIT erlaubt
0x1112	falsche Datenlänge bei SetDeviceState
0x1113	AddBox im Zustand INIT nicht erlaubt
0x1114	falsche Datenlänge bei AddBox
0x1115	DeleteBox im Zustand INIT nicht erlaubt
0x1116	falscher IndexOffset bei DeleteBox
0x1117	falsche Datenlänge bei DeleteBox
0x1118	ReadBox nur mit AdsRead
0x1119	AddCdl im Zustand INIT nicht erlaubt
0x111A	falsche Datenlänge bei AddCdl
0x111B	DeleteCdl im Zustand INIT nicht erlaubt
0x111C	falscher IndexOffset bei DeleteCdl
0x111D	falsche Datenlänge bei DeleteCdl
0x111E	falsche IndexGroup bei AdsWrite
0x111F	Device-Parameter können nicht gelesen werden

Error	Beschreibung
Fehler bei der ADS/AMS - Datenaustausch	
0x1120	Box-Parameter können nicht gelesen werden
0x1121	Cdl-Parameter können nicht gelesen werden
0x1122	DeleteBox bzw. DeleteCdl nur mit AdsWrite
0x1123	ReadBox nur im Zustand STOP möglich
0x1124	falscher IndexOffset bei ReadBox
0x1125	falsche Datenlänge bei ReadBox
0x1126	falsche IndexGroup bei AdsRead
0x1127	AddDeviceNotification im Zustand INIT nicht erlaubt
0x1128	DelDeviceNotification im Zustand INIT nicht erlaubt
0x1129	IndexOffset zu groß beim Lesen der Device-Diagnosedaten
0x112B	IndexOffset zu groß beim Lesen der Box-Diagnosedaten
0x112F	nicht genügend Speicher für ReadBox-Response
0x1201	AddCdl: Cdl-No ist zu groß
0x1202	DeleteCdl nur möglich, wenn CDL gestoppt ist
0x1203	DeleteCdl nicht möglich, da keine CDL definiert
0x1204	Zyklus konnte innerhalb der internen Watchdog-Zeit nicht beendet werden
0x1301	AddCdl: IO-Access-Multiplier ist zu groß
0x1302	AddCdl: Start-Cycle muss kleiner als IO-Access-Multiplier sein
0x1303	AddCdl: falsche Datenlänge der Output-Area
0x1304	AddCdl: falsche Datenoffset der Output-Area

Error	Beschreibung
Fehler bei der ADS/AMS - Datenaustausch	
0x1305	AddCdl: Output-Area ist bereits definiert
0x1306	AddCdl: falsche Datenlänge der Input-Area
0x1307	AddCdl: falsche Datenoffset der Input-Area
0x1308	AddCdl: Input-Area ist bereits definiert
0x1309	AddCdl: falscher Area-Typ
0x130A	AddCdl: BoxNo wurde nicht mit AddBox definiert
0x130B	AddCdl: falscher Aktions-Typ
0x130C	AddCdl: nicht genügend Speicher für Poll-Liste
0x130D	AddCdl: nicht genügend Speicher für Poll-Listen-Array
0x130E	AddCdl: nicht genügend Speicher für Aktionen
0x130F	AddCdl: CdlNo existiert bereits

Error	Beschreibung
Fehler bei der ADS/AMS - Datenaustausch	
0x1310	DeleteCdl: Cdl ist nicht gestoppt
0x1311	AddCdl: nicht genügend Speicher für asynchrone Sende-Liste
0x1312	AddCdl: nicht genügend Speicher für synchrone Empfangs-Liste
0x1313	AddCdl: nicht genügend Speicher für asynchrone Empfangs-Liste
0x1316	AddCdl: nicht genügend Speicher für synchrone Empfangs-Liste
0x1318	AddCdl: nur Slave-Aktion erlaubt
0x1319	AddCdl: nicht genügend Speicher für Slave-Liste
0x1601	AddBox: BoxNo ist zu groß
0x1602	AddBox: nicht genügend Speicher für ADS-StartUp-Telegramme
0x1604	DeleteBox: Box ist nicht gestoppt
0x1605	AddBox: nicht genügend Speicher für CDL-Telegramme
0x1606	AddBox: Anzahl der CDL-Telegramme ist zu groß
0x1607	BoxRestart: Box ist nicht gestoppt
0x1608	BoxRestart: Syntaxfehler AdsWriteControl
0x1609	BoxRestart: falscher AdsState
0x160A	Syntaxfehler bei AdsWrite an Box-Port
0x160B	AMS-CmdId wird von Box-Port nicht unterstützt
0x160E	AdsReadState wird von Box-Port nicht unterstützt
0x160F	AddBox: nicht genügend Speicher für das ADS-Interface
0x1610	AddBox: AMS-Channel ist ungültig
0x1611	Fehler Kommunikation zu einer AMS-Box
0x1613	Fehler Kommunikation zu einer AMS-Box: Offset ist falsch
0x1614	Fehler Kommunikation zu einer AMS-Box: Häppchen ist zu groß
0x1615	Fehler Kommunikation zu einer AMS-Box: AMS-Kommando ist zu groß
0x1616	Fehler Kommunikation zu einer AMS-Box: erstes Häppchen ist zu groß
0x1617	Fehler Kommunikation zu einer AMS-Box: erster Offset ist falsch

Error	Beschreibung
Fehler bei der ADS/AMS - Datenaustausch	
0x1701	AddDeviceNotification: Länge der Device-Diagnosedaten zu klein
0x1702	AddDeviceNotification: Länge der Device-Diagnosedaten zu groß
0x1703	AddDeviceNotification: Länge der Box-Diagnosedaten zu klein
0x1704	AddDeviceNotification: Länge der Box-Diagnosedaten zu groß
0x1705	AddDeviceNotification: Box ist nicht definiert
0x1706	AddDeviceNotification: falsche IndexGroup
0x1707	AddDeviceNotification: keine Ressourcen mehr für Client
0x1708	DelDeviceNotification: falscher Handle
0x1801	StartFieldbus: Im Equidistant-Betrieb müssen Shift-Time + Safety-Time + 2*PLL-Sync-Time größer als die Cycle-Time sein
0x1802	StartFieldbus: Cycle-Time ist zu groß
0x1803	StartFieldbus: Cycle-Time ist zu groß
0x1804	StartFieldbus: Shift-Time ist zu groß
0x1805	StartFieldbus: PLL-Sync-Time ist zu groß

Error	Beschreibung
Fehler bei der ADS/AMS - Datenaustausch	
0x1806	StartFieldbus: Safety-Time ist zu groß
0x1807	StartFieldbus: Cycle-Times kleiner 1 ms müssen ein ganzzahliger Teiler von 1 ms sein
0x1A01	Speicher vom Huge-Heap konnte nicht alloziert werden, da er größer als 0x8000 Bytes war
0x1A02	Speicher vom Near-Heap konnte nicht alloziert werden, da er größer als 0x1000 Bytes war
0x1A03	Speicher vom Huge-Heap konnte nicht alloziert werden, da er 0 Bytes war
0x1A04	Speicher vom Near-Heap konnte nicht alloziert werden, da er 0 Bytes war
Fehler bei der Initialisieren der DeviceNet®-Konfiguration	
0x2001 .. 0x2xxx	
Fehler bei Expliziten DeviceNet® Datenaustausch	
0x2300	GENERR_RESUNAVAILABLE
0x2301	ADSERR_DEVICE_SRVNOTSUPP
0x2302	GENERR_INVALIDATTRVAL
0x2303	GENERR_ALRERADYINREQU
0x2304	GENERR_OBJECTSTATECONF
0x2305	GENERR_ATTRNOTSETABLE
0x2306	GENERR_PRIVVIOLATION
0x2307	GENERR_REPLDATTOOLARGE
0x2308	GENERR_NOTENOUGHDATA
0x2309	GENERR_ATTRNOTSUPP
0x230A	GENERR_TOOMUCHDATA
0x230B	GENERR_OBJECTNOTEXIST
0x230C	GENERR_NOSTOREATTRDATA
0x230D	GENERR_STOREOPFAIL
0x230E	GENERR_VENDORSPEC
0x230F	GENERR_INVALIDPARAM
0x2310	GENERR_INVALIDMEMBERID
0x2311	GENERR_MEMBERNOTSET
0x2312	ADSERR_DEVICE_SYMBOLNOTFOUND
0x2313	GENERR_OBJECTSTATECONF

7.5 DeviceNet® / CAN Trouble Shooting

Error Frames

Fehler in der CAN-Verkabelung, der Adressvergabe und der Baud-Rateneinstellung zeigen sich u.a. durch eine erhöhte Anzahl an Error Frames: die Diagnose LEDs zeigen dann *Warning Limit wird überschritten* oder *Bus-Off-Zustand erreicht*.



DeviceNet® / CAN Netzwerkanalyse

Überschrittenes CAN Warning Limit, Error Passive oder Bus-Off Zustand werden zunächst bei dem Knoten angezeigt, der die meisten Fehler entdeckt hat. Dieser Knoten muss nicht unbedingt die Ursache für das Auftreten dieser Error Frames sein! Wenn z. B. ein Knoten überdurchschnittlich stark zum Busverkehr beiträgt (z. B. weil er als einziger über analoge Eingänge verfügt, deren Daten in kurzen Abständen ereignisgesteuerte Nachrichten auslösen), so werden auch seine Telegramme mit großer Wahrscheinlichkeit zunächst gestört - entsprechend erreicht sein Fehlerzähler als erster kritische Zustände.

MAC-ID / Baud Rate Einstellung

Es muss sorgfältig darauf geachtet werden, dass keine Knotenadresse / MAC-ID doppelt vergeben ist.

Test 1

MAC-ID überprüfen. Falls die DeviceNet® Kommunikation wenigstens zeitweise funktioniert und alle Geräte den Duplicat MAC-ID Check unterstützen, so kann die Adressvergabe auch durch Aufzeichnen der Duplicat MAC-ID Check -Nachrichten nach dem Einschalten der Geräte überprüft werden - hierdurch wird aber kein Vertauschen von Knotenadressen erkannt.

Test 2

Überprüfen, ob überall die gleiche Baud-Rate eingestellt ist.

Test der DeviceNet®/CAN-Verkabelung

Diese Tests nicht ausführen, wenn das Netzwerk aktiv ist: Während der Tests sollte keine Kommunikation stattfinden. Die folgenden Tests sollten in der angegebenen Reihenfolge ausgeführt werden, da manche Tests davon ausgehen, dass der vorhergehende Test erfolgreich war. In der Regel sind nicht alle Tests notwendig.

Netzwerkabschluss und Signalleitungen

Für diesen Test sollten die Knoten ausgeschaltet oder die CAN-Leitung abgesteckt sein, da die Messergebnisse sonst durch die aktiven CAN-Transceiver verfälscht werden können.

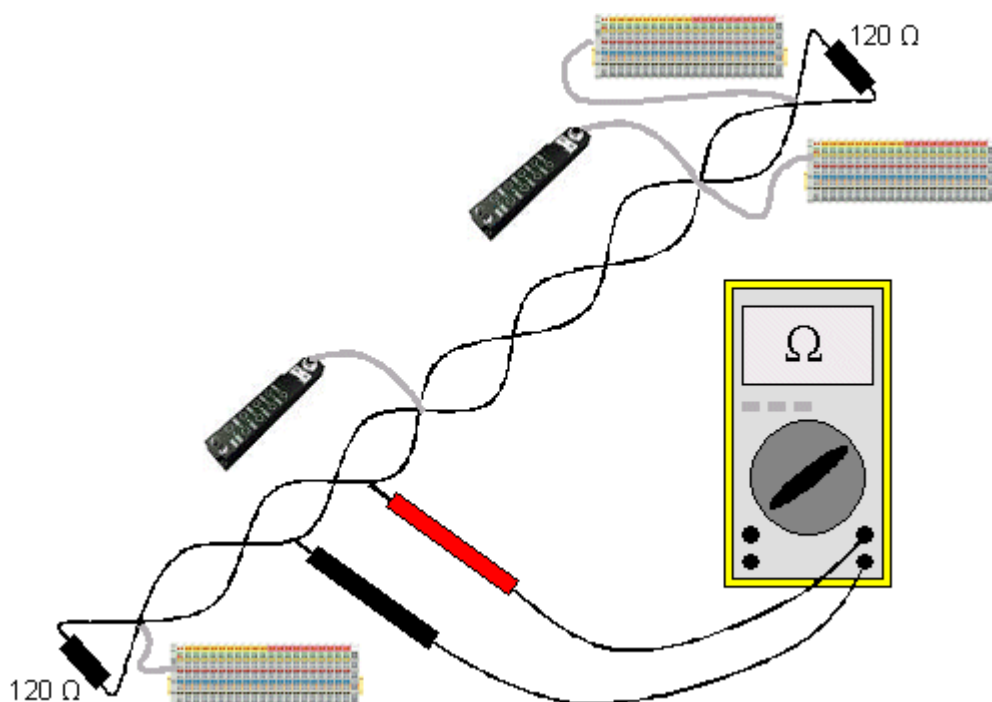


Abb. 75: Verdrahtungsplan für Testaufbau

Test 3

Widerstand zwischen CAN-high und CAN-low ermitteln - ggf. bei jedem Gerät.

Wenn der Messwert über 65 Ohm liegt, deutet dies auf fehlende Abschlusswiderstände oder den Bruch einer Signalleitung hin. Wenn der Messwert kleiner 50 Ohm ist, nach Kurzschluss zwischen CAN-Leitung, überzähligen Abschlusswiderständen oder fehlerhaften Transceivern suchen.

Test 4

Auf Kurzschluss zwischen CAN-Ground und den Signalleitungen sowie zwischen Schirm und Signalleitungen prüfen.

Test 5

Erdung von CAN-Ground und Schirm auftrennen. Auf Kurzschluss zwischen CAN-Ground und Schirm prüfen.

Topologie

Die Leitungslänge bei CAN Netzwerken hängt stark von der gewählten Baud-Rate ab. CAN toleriert dabei kurze Stichleitungen - ebenfalls in Abhängigkeit von der Baud-Rate. Die erlaubte Stichleitungslänge sollte nicht überschritten werden. Häufig wird die verlegte Leitungslänge unterschätzt - die Schätzung liegt teilweise Faktor 10 unter der tatsächlichen Länge. Deshalb empfiehlt sich folgender Test:

Test 6

Die Stichleitungslängen sowie die Busgesamtlänge nachmessen (nicht nur grob schätzen!) und mit den Topologieregeln (abhängig von der Baud-Rate) vergleichen.

Schirmung und Erdung

Stromversorgung und Schirm sollten sorgfältig, einmalig und niederohmig beim Netzteil geerdet werden. Alle Verbindungsstellen, Abzweige etc. im CAN-Kabel müssen neben den Signalleitungen (und evtl. CAN-GND) auch den Schirm durchverbinden. In den Beckhoff IP20 Buskopplern wird der Schirm über ein R/C-Glied hochfrequenzmäßig geerdet.

Test 7

Mit DC-Strommessgerät (16 A max.) Strom zwischen Spannungsversorgungs-Masse und Schirm am vom Netzteil entfernten Ende des Netzes messen. Es sollte ein Ausgleichsstrom vorhanden sein. Wenn kein Strom vorhanden ist, so ist der Schirm nicht durchgängig verbunden oder das Netzteil ist nicht richtig geerdet. Wenn das Netzteil in der Mitte des Netzwerkes angeordnet ist, so sollte an beiden Enden gemessen werden. Dieser Test kann u.U. auch an den Stichleitungsenden durchgeführt werden.

Test 8

Den Schirm an mehreren Stellen auftrennen und den Verbindungsstrom messen. Wenn ein Stromfluss vorhanden ist, so ist der Schirm an mehreren Stellen geerdet (Erdschleife).

Potentialunterschiede

Der Schirm muss für diesen Test durchgängig sein und darf keinen Strom führen (vorher getestet).

Test 9

Spannung zwischen Schirm und Spannungsversorgungs-Erde an jedem Knoten ermitteln und notieren. Der maximale Potentialunterschied zwischen zwei beliebigen Geräten sollte kleiner als 5 Volt sein.

Fehler erkennen und lokalisieren

Die Regel des "Low-tech-Ansatzes" ist die beste Lokalisierungsmethode: Teile des Netzes abhängen und beobachten, wann der Fehler verschwindet.

Aber: das Vorgehen nach dieser Methode ist unzureichend bei Problemen wie zu großen Potentialunterschieden, Masseschleifen, EMV und Signalverfälschung da die Verkleinerung des Netzes häufig das Problem löst, ohne dass der „fehlende“ Teil ursächlich war. Auch die Buslast ändert sich beim Verkleinern des Netzes - damit können externe Störungen seltener CAN-Telegramme "treffen".

Die Diagnose mittels Oszilloskop führt meist nicht zum Erfolg: CAN Signale sehen auch im ungestörten Zustand teilweise recht wirr aus. Unter Umständen kann mit einem Speicheroszilloskop auf Error Frames getriggert werden - diese Art der Diagnose ist aber Messtechnik-Experten vorbehalten.

Protokollprobleme

In seltenen Fällen sind auch Protokollprobleme (z. B. fehlerhafte oder unvollständige DeviceNet®-Implementierung, unglückliches Timing im Boot-Up etc.) Ursache von Störungen. Hier ist dann ein Mitschrieb (Trace) des Busverkehrs mit anschließender Auswertung durch DeviceNet® Experten erforderlich - das Beckhoff Support Team [► 134] kann hier helfen.

Für solch einen Trace eignet sich z. B. ein freier Kanal einer Beckhoff FC5102 CANopen PCI-Karte - die erforderliche Trace-Software stellt Beckhoff im Internet zur Verfügung. Alternativ kann selbstverständlich auch ein handelsübliches CAN Analysetool eingesetzt werden.

8 Anhang

8.1 EtherCAT AL Status Codes

Detaillierte Informationen hierzu entnehmen Sie bitte der vollständigen [EtherCAT-Systembeschreibung](#).

8.2 Firmware Kompatibilität

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

Anmerkung

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

HINWEIS

Beschädigung des Gerätes möglich!

Beachten Sie die Hinweise zum Firmware Update auf der gesonderten Seite [► 121]. Wird ein Gerät in den BOOTSTRAP-Mode zum Firmware-Update versetzt, prüft es u.U. beim Download nicht, ob die neue Firmware geeignet ist. Dadurch kann es zur Beschädigung des Gerätes kommen! Vergewissern Sie sich daher immer, ob die Firmware für den Hardware-Stand des Gerätes geeignet ist!

EL6752			
Hardware (HW)	Firmware (FW)	Revision-Nr.	Releasedatum
06 - 24*	07	EL6752-0000-0016	2008/06
	08		2008/11
	09		2010/05
		EL6752-0000-0017	2011/10
	10		2012/01
		EL6752-0000-0018	2012/10
	11	EL6752-0000-0019	2014/07
	12	EL6752-0000-0020	2014/06
	13		2015/02
	14		2019/04
	15		2020/06
	16*		2025/04

EL6752-0010			
Hardware (HW)	Firmware (FW)	Revision-Nr.	Releasedatum
06 - 24*	06	EL6752-0010-0016	2008/04
	07		2008/06
	08		2008/11
		EL6752-0010-0017	2011/10
	09		2012/01
	10		2012/05
		EL6752-0010-0018	2012/10
	11	EL6752-0010-0019	2014/07
	12	EL6752-0010-0020	2014/06
	13		2015/02
	14		2019/04
	15		2020/03
	16*		2024/02

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

8.3 Firmware Update EL/ES/ELM/EM/EP/EPP/ERPxxxx

Dieses Kapitel beschreibt das Geräte-Update für Beckhoff EtherCAT-Slaves der Serien EL/ES, ELM, EM, EK, EP, EPP und ERP. Ein FW-Update sollte nur nach Rücksprache mit dem Beckhoff Support durchgeführt werden.

HINWEIS

Nur TwinCAT 3 Software verwenden!

Ein Firmware-Update von Beckhoff IO Geräten ist ausschließlich mit einer TwinCAT 3-Installation durchzuführen. Es empfiehlt sich ein möglichst aktuelles Build, kostenlos zum Download verfügbar auf der [Beckhoff-Website](#).

Zum Firmware-Update kann TwinCAT im sog. FreeRun-Modus betrieben werden, eine kostenpflichtige Lizenz ist dazu nicht nötig.

Das für das Update vorgesehene Gerät kann in der Regel am Einbauort verbleiben; TwinCAT ist jedoch im FreeRun zu betreiben. Zudem ist auf eine störungsfreie EtherCAT Kommunikation zu achten (keine „LostFrames“ etc.).

Andere EtherCAT-Master-Software wie z. B. der EtherCAT-Konfigurator sind nicht zu verwenden, da sie unter Umständen nicht die komplexen Zusammenhänge beim Update von Firmware, EEPROM und ggf. weiteren Gerätebestandteilen unterstützen.

Speicherorte

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

- Jeder EtherCAT-Slave hat eine Gerätebeschreibung, bestehend aus Identität (Name, Productcode), Timing-Vorgaben, Kommunikationseinstellungen u. a.
Diese Gerätebeschreibung (ESI; EtherCAT-Slave Information) kann von der Beckhoff Website im Downloadbereich als [Zip-Datei](#) heruntergeladen werden und in EtherCAT-Mastern zur Offline-Konfiguration verwendet werden, z. B. in TwinCAT.
Vor allem aber trägt jeder EtherCAT-Slave seine Gerätebeschreibung (ESI) elektronisch auslesbar in einem lokalen Speicherchip, dem einem sog. **ESI-EEPROM**. Beim Einschalten wird diese Beschreibung einerseits im Slave lokal geladen und teilt ihm seine Kommunikationskonfiguration mit, andererseits kann der EtherCAT-Master den Slave so identifizieren und u. a. die EtherCAT Kommunikation entsprechend einrichten.

HINWEIS

Applikationsspezifisches Beschreiben des ESI-EEPROM

Die ESI wird vom Gerätehersteller nach ETG-Standard entwickelt und für das entsprechende Produkt freigegeben.

- Bedeutung für die ESI-Datei: Eine applikationsseitige Veränderung (also durch den Anwender) ist nicht zulässig.

- Bedeutung für das ESI-EEPROM: Auch wenn technisch eine Beschreibbarkeit gegeben ist, dürfen die ESI-Teile im EEPROM und ggf. noch vorhandene freie Speicherbereiche über den normalen Update-Vorgang hinaus nicht verändert werden. Insbesondere für zyklische Speichervorgänge (Betriebsstundenzähler u. ä.) sind dezidierte Speicherprodukte wie EL6080 oder IPC-eigener NOVRAM zu verwenden.

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

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

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

Vereinfachtes Update per Bundle-Firmware

Bequemer ist das Update per sog. **Bundle-Firmware**: hier sind die Controller-Firmware und die ESI-Beschreibung in einer *.efw-Datei zusammengefasst, beim Update wird in der Klemme sowohl die Firmware, als auch die ESI verändert. Dazu ist erforderlich

- dass die Firmware in dem gepackten Format vorliegt: erkenntlich an dem Dateinamen der auch die Revisionsnummer enthält, z. B. ELxxxx-xxxx_REV0016_SW01.efw
- dass im Download-Dialog das Passwort = 1 angegeben wird. Bei Passwort = 0 (default Einstellung) wird nur das Firmware-Update durchgeführt, ohne ESI-Update.
- dass das Gerät diese Funktion unterstützt. Die Funktion kann in der Regel nicht nachgerüstet werden, sie wird Bestandteil vieler Neuentwicklungen ab Baujahr 2016.

Nach dem Update sollte eine Erfolgskontrolle durchgeführt werden

- ESI/Revision: z. B. durch einen Online-Scan im TwinCAT ConfigMode/FreeRun – dadurch wird die Revision bequem ermittelt
- Firmware: z. B. durch einen Blick ins Online-CoE des Gerätes

HINWEIS

Beschädigung des Gerätes möglich!

- ✓ Beim Herunterladen von neuen Gerätedateien ist zu beachten
 - a) Das Herunterladen der Firmware auf ein EtherCAT-Gerät darf nicht unterbrochen werden.
 - b) Eine einwandfreie EtherCAT-Kommunikation muss sichergestellt sein, CRC-Fehler oder LostFrames dürfen nicht auftreten.
 - c) Die Spannungsversorgung muss ausreichend dimensioniert, die Pegel entsprechend der Vorgabe sein.
- ⇒ Bei Störungen während des Update-Vorgangs kann das EtherCAT-Gerät ggf. nur vom Hersteller wieder in Betrieb genommen werden!

8.3.1 Gerätebeschreibung ESI-File/XML

HINWEIS

ACHTUNG bei Update der ESI-Beschreibung/EEPROM

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

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

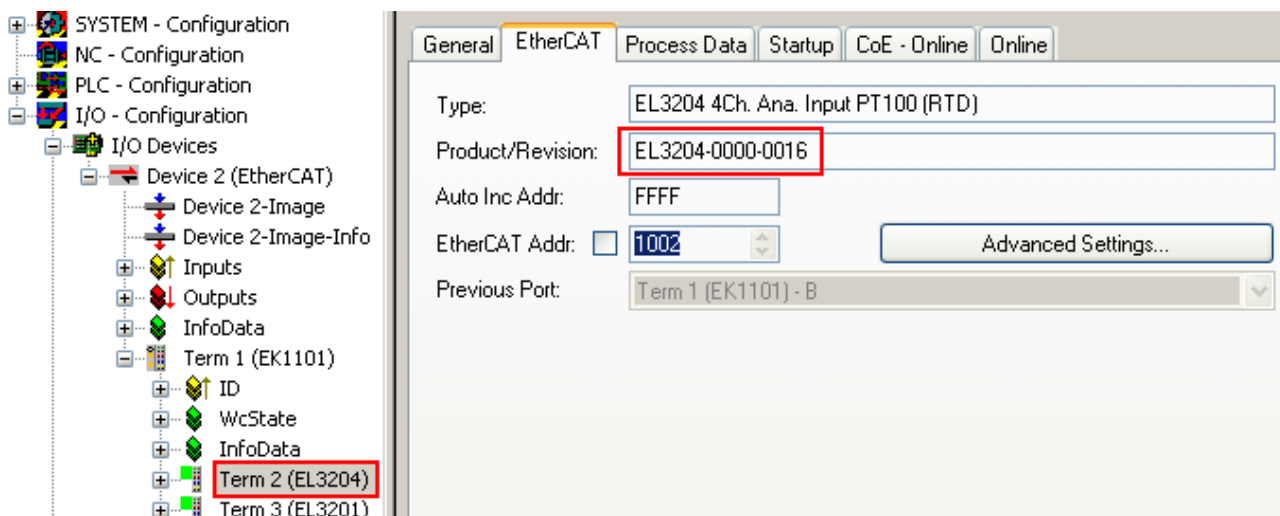


Abb. 76: Geräteerkennung aus Name EL3204-0000 und Revision -0016

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

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

i Update von XML/ESI-Beschreibung

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

Anzeige der Slave-Kennung ESI

Der einfachste Weg die Übereinstimmung von konfigurierter und tatsächlicher Gerätebeschreibung festzustellen, ist im TwinCAT-Modus Config/FreeRun das Scannen der EtherCAT-Boxen auszuführen:

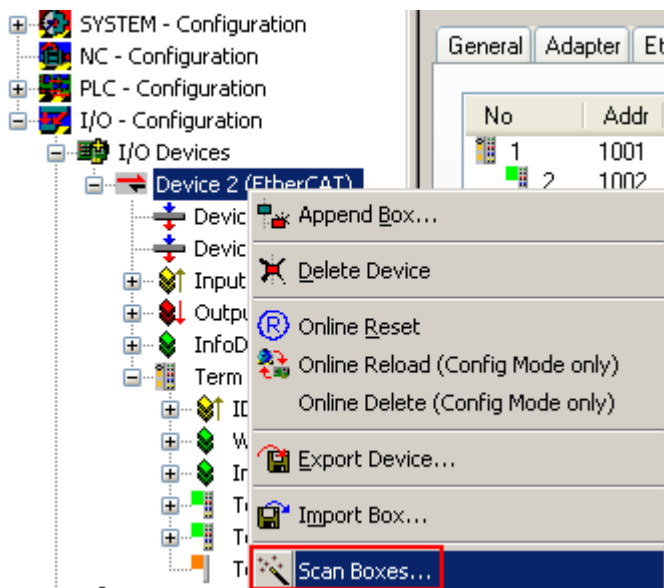


Abb. 77: Rechtsklick auf das EtherCAT-Gerät bewirkt das Scannen des unterlagerten Feldes

Wenn das gefundene Feld mit dem konfigurierten übereinstimmt, erscheint



Abb. 78: Konfiguration identisch

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

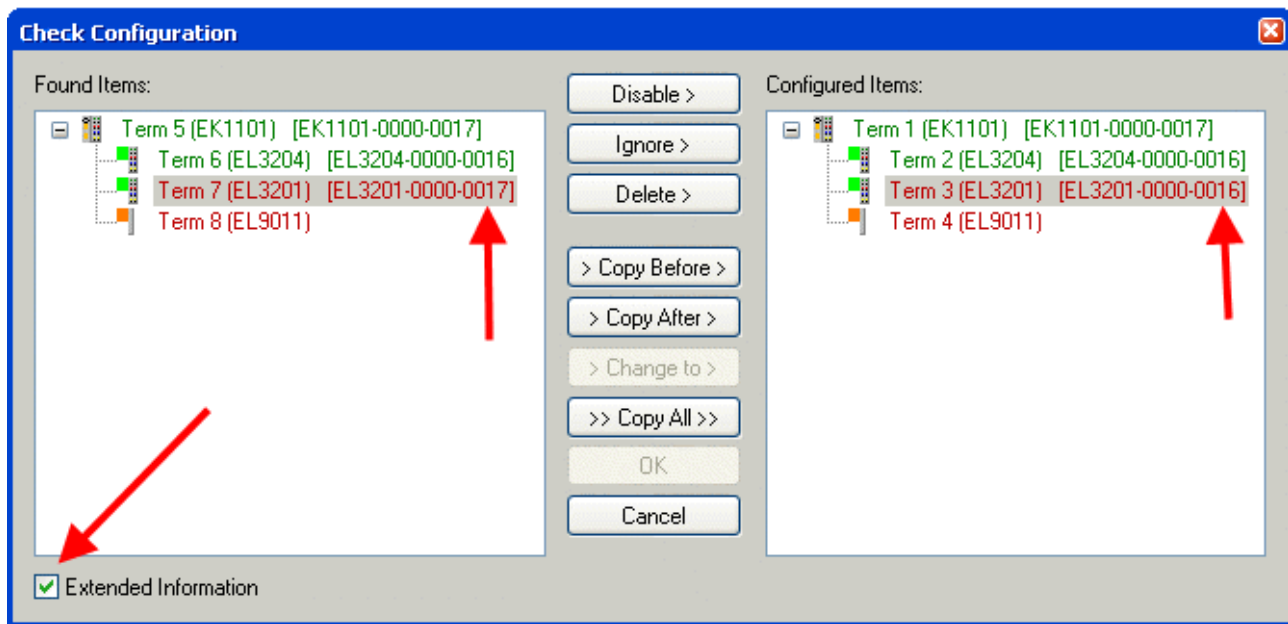


Abb. 79: Änderungsdialog

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

Änderung der Slave-Kennung ESI

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

- Es muss eine einwandfreie EtherCAT-Kommunikation zum Slave hergestellt werden
- Der State des Slave ist unerheblich
- Rechtsklick auf den Slave in der Online-Anzeige führt zum Dialog *EEPROM Update*, Abb. *EEPROM Update*

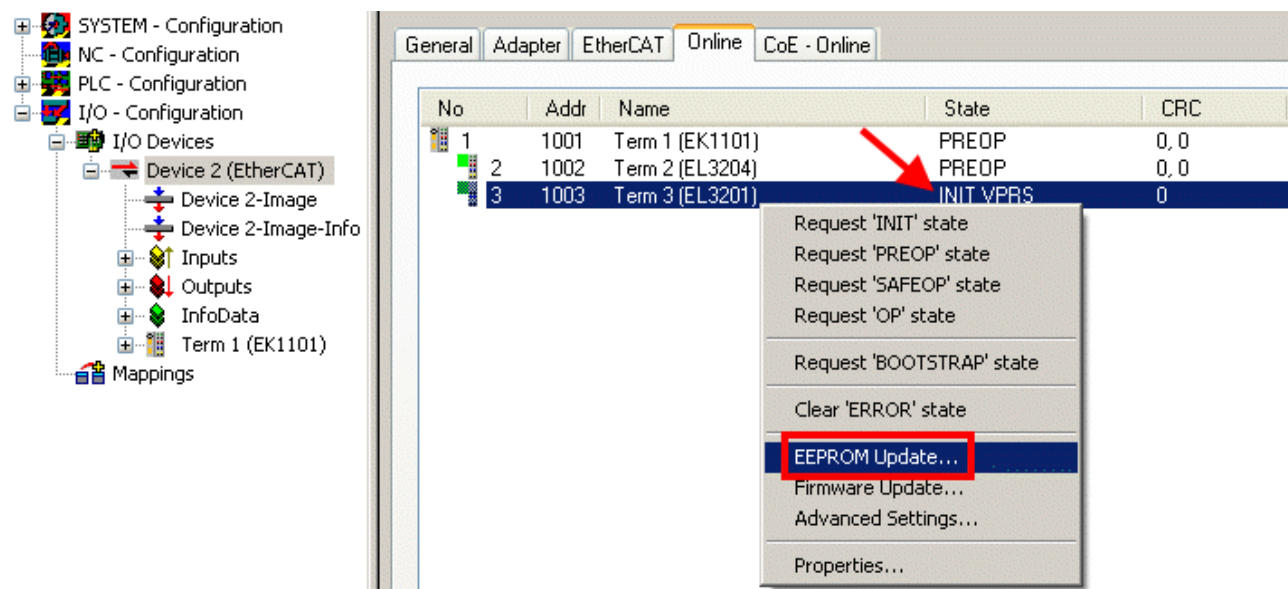


Abb. 80: EEPROM Update

Im folgenden Dialog wird die neue ESI-Beschreibung ausgewählt, s. Abb. *Auswahl des neuen ESI*. Die CheckBox *Show Hidden Devices* zeigt auch ältere, normalerweise ausgeblendete Ausgaben eines Slave.

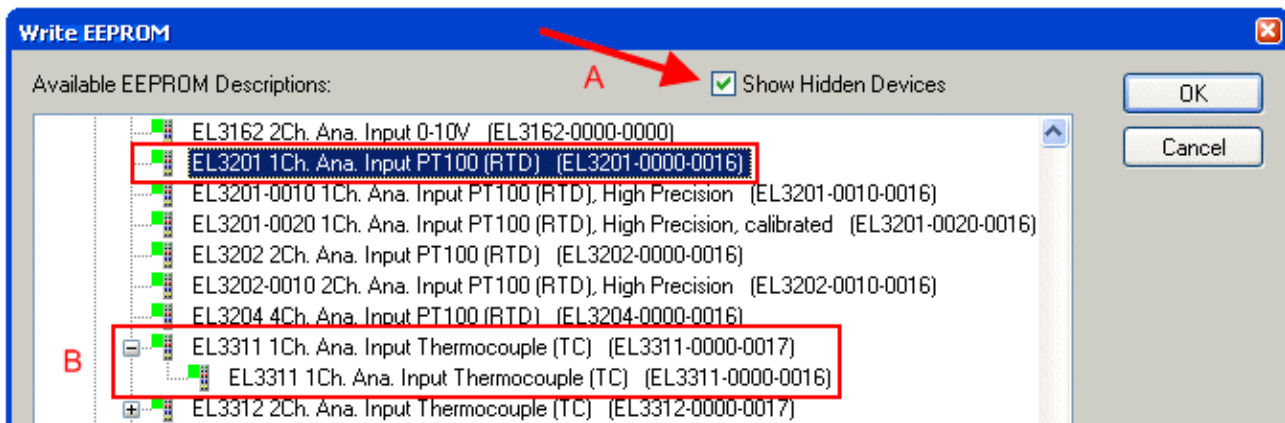


Abb. 81: Auswahl des neuen ESI

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

i Änderung erst nach Neustart wirksam

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

8.3.2 Erläuterungen zur Firmware

Versionsbestimmung der Firmware

Versionsbestimmung mit dem TwinCAT System Manager

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

i CoE-Online und Offline-CoE

Es existieren zwei CoE-Verzeichnisse:

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

Die Umschaltung zwischen beiden Ansichten kann über den Button *Advanced* vorgenommen werden.

In Abb. *Anzeige FW-Stand EL3204* wird der FW-Stand der markierten EL3204 in CoE-Eintrag 0x100A mit 03 angezeigt.

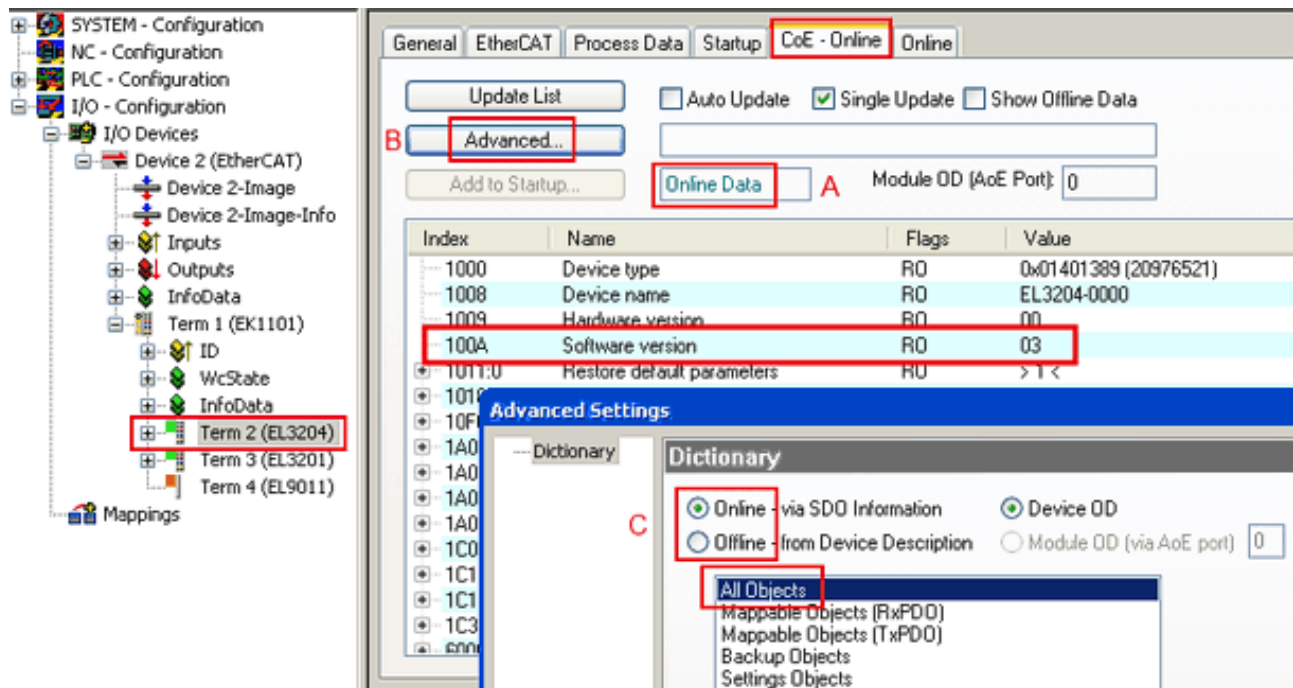


Abb. 82: Anzeige FW-Stand EL3204

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

8.3.3 Update Controller-Firmware *.efw



CoE-Verzeichnis

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

Um die Controller-Firmware eines Slave zu aktualisieren, wechseln Sie zum Karteireiter *Online*, s. Abb. *Firmware Update*.

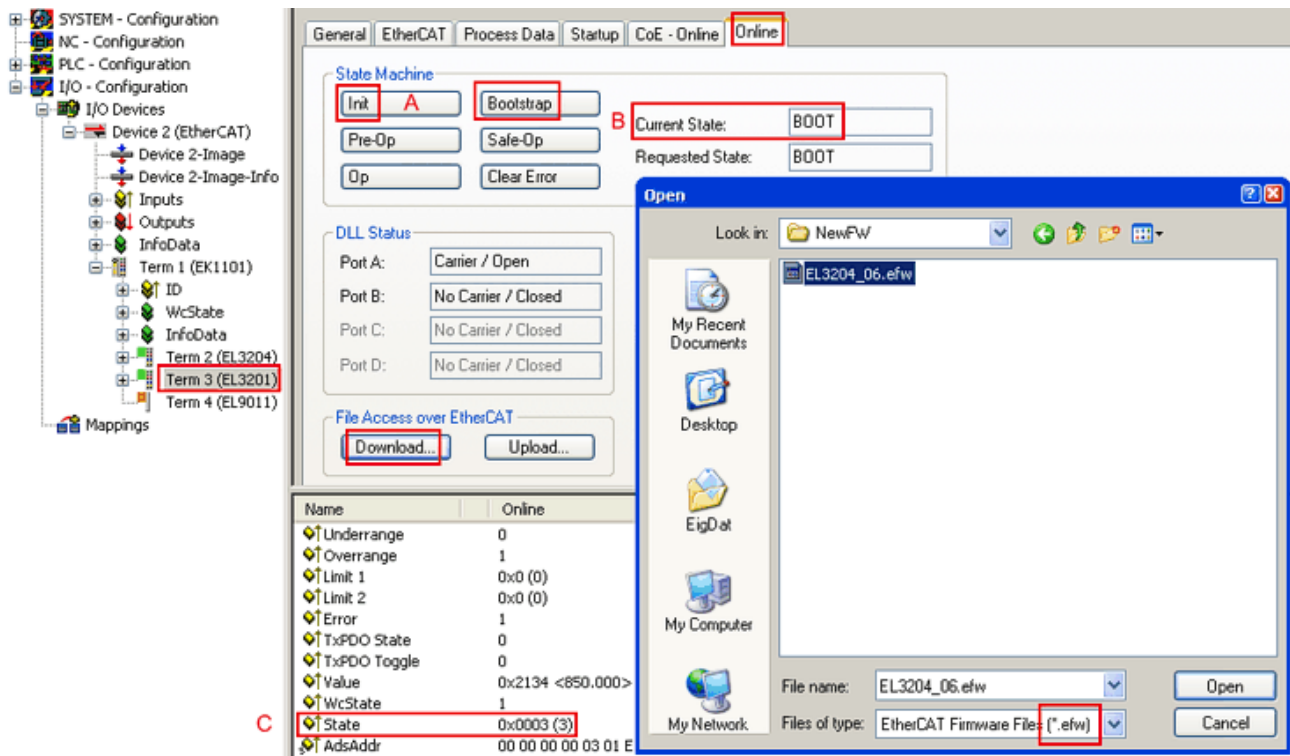
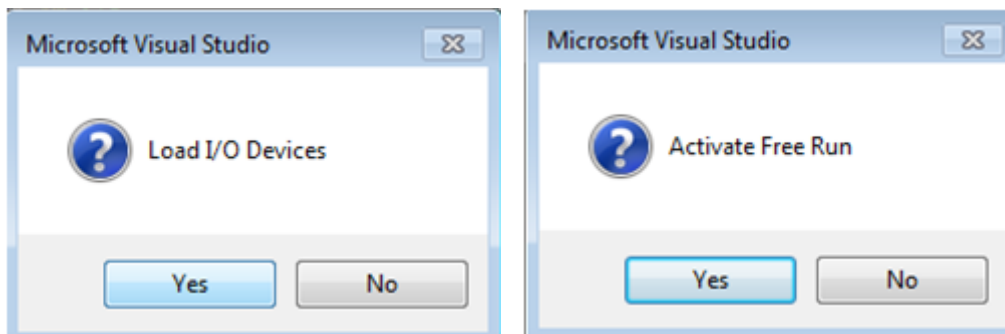


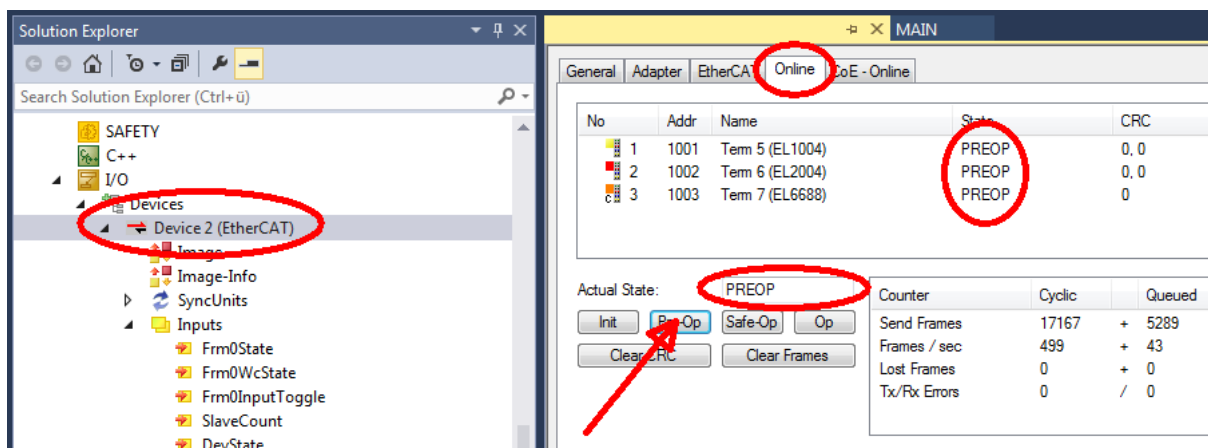
Abb. 83: Firmware Update

Es ist folgender Ablauf einzuhalten, wenn keine anderen Angaben z. B. durch den Beckhoff Support vorliegen. Gültig für TwinCAT 2 und 3 als EtherCAT-Master.

- TwinCAT System in ConfigMode/FreeRun mit Zykluszeit ≥ 1 ms schalten (default sind im ConfigMode 4 ms). Ein FW-Update während Echtzeitbetrieb ist nicht zu empfehlen.

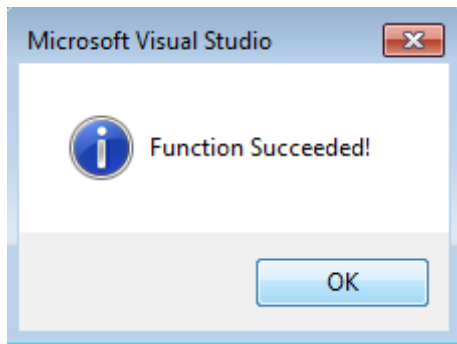


- EtherCAT-Master in PreOP schalten



- Slave in INIT schalten (A)
- Slave in BOOTSTRAP schalten

- Kontrolle des aktuellen Status (B, C)
- Download der neuen *efw-Datei, abwarten bis beendet. Ein Passwort wird in der Regel nicht benötigt.



- Nach Beendigung des Download in INIT schalten, dann in PreOP
- Slave kurz stromlos schalten (nicht unter Spannung ziehen!)
- Im CoE 0x100A kontrollieren ob der FW-Stand korrekt übernommen wurde.

8.3.4 FPGA-Firmware *.rbf

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

- Controller-Firmware für die Aufbereitung der E/A-Signale
- FPGA-Firmware für die EtherCAT-Kommunikation (nur für Klemmen mit FPGA)

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

Versionsbestimmung mit dem TwinCAT System-Manager

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

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

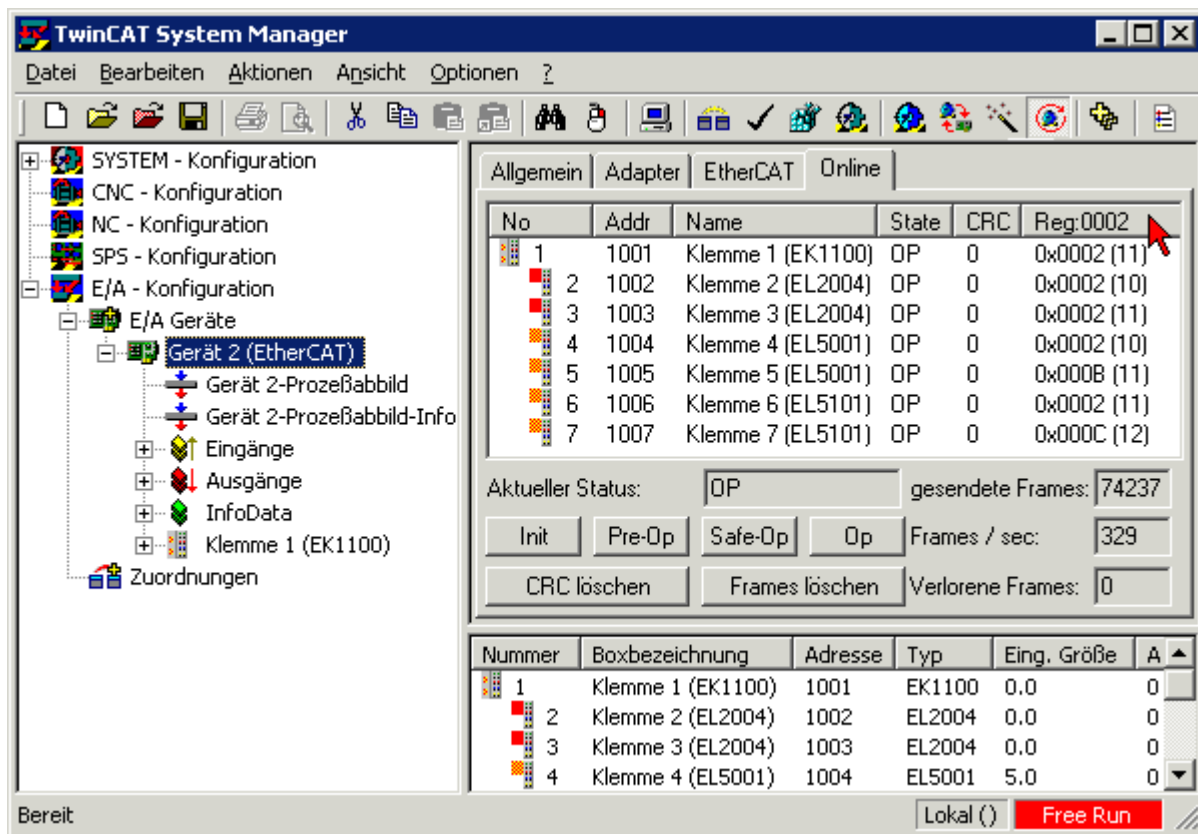
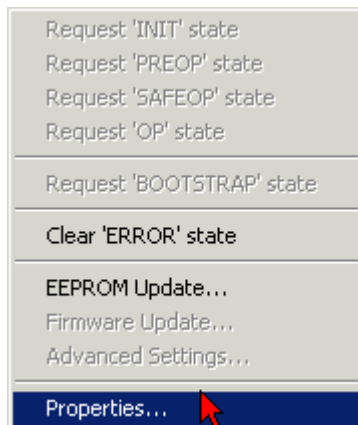
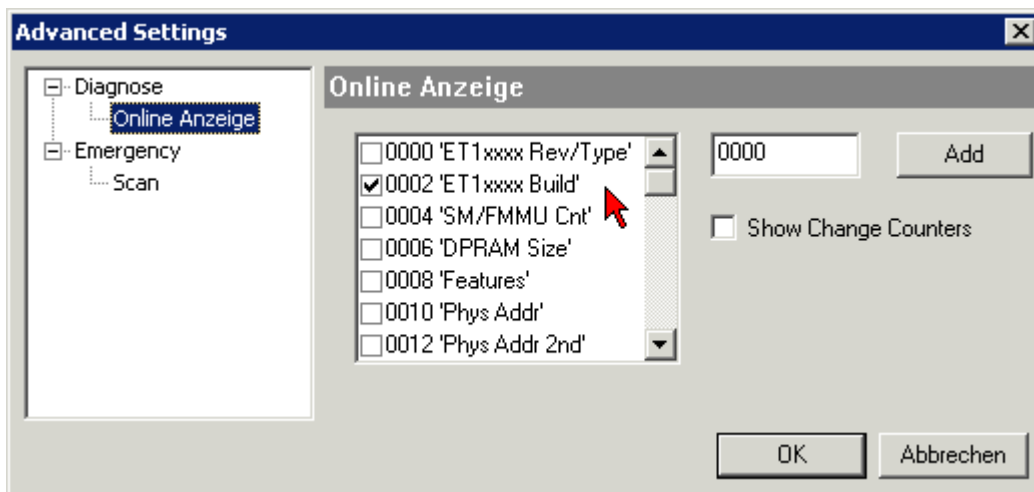


Abb. 84: Versionsbestimmung FPGA-Firmware

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

Abb. 85: Kontextmenu *Eigenschaften (Properties)*

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

Abb. 86: Dialog *Advanced settings*

Update

Für das Update der FPGA-Firmware

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

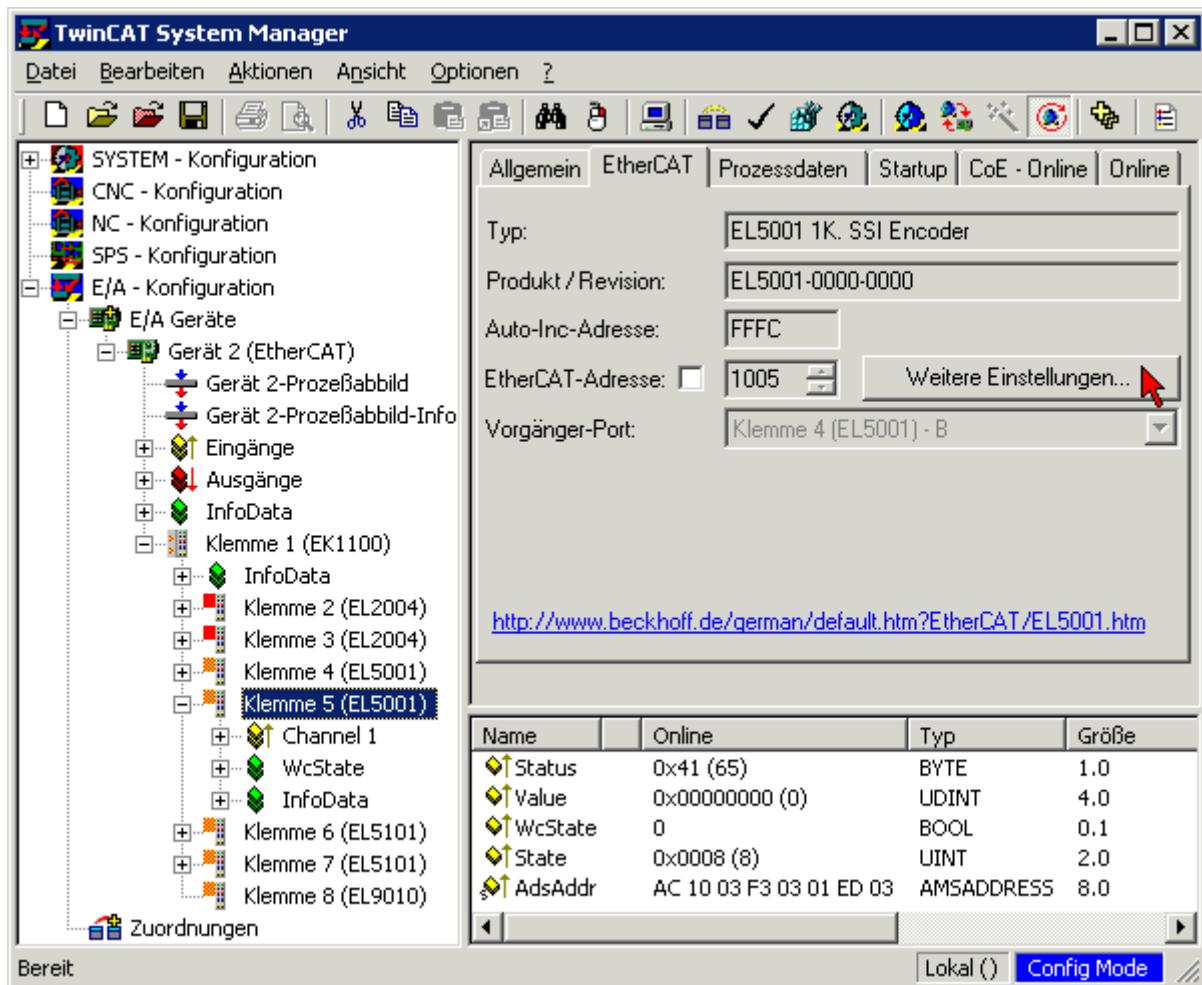
Ältere Firmware-Stände können nur vom Hersteller aktualisiert werden!

Update eines EtherCAT-Geräts

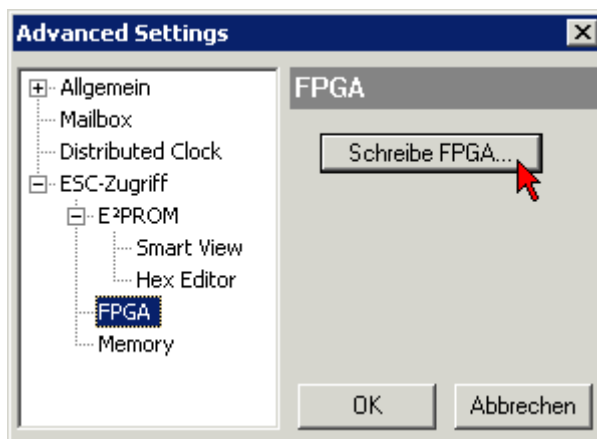
Es ist folgender Ablauf einzuhalten, wenn keine anderen Angaben z. B. durch den Beckhoff Support vorliegen:

- TwinCAT System in ConfigMode/FreeRun mit Zykluszeit ≥ 1 ms schalten (default sind im ConfigMode 4 ms). Ein FW-Update während Echtzeitbetrieb ist nicht zu empfehlen.

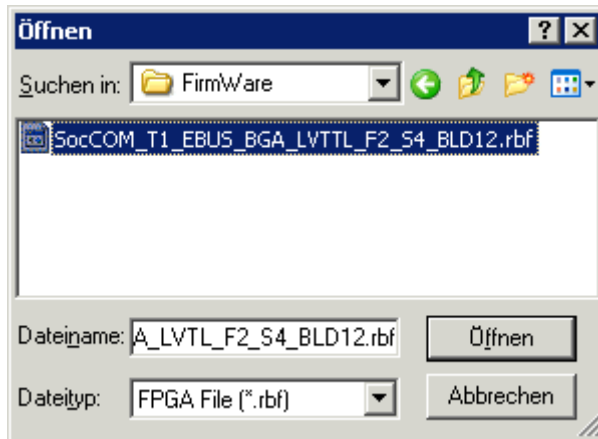
- Wählen Sie im TwinCAT System Manager die Klemme an, deren FPGA-Firmware Sie aktualisieren möchten (im Beispiel: Klemme 5: EL5001) und klicken Sie auf dem Karteireiter *EtherCAT* auf die Schaltfläche *Weitere Einstellungen*:



- Im folgenden Dialog *Advanced Settings* klicken Sie im Menüpunkt *ESC-Zugriff/E²PROM/FPGA* auf die Schaltfläche *Schreibe FPGA*:



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



- Abwarten bis zum Ende des Downloads
- Slave kurz stromlos schalten (nicht unter Spannung ziehen!). Um die neue FPGA-Firmware zu aktivieren ist ein Neustart (Aus- und Wiedereinschalten der Spannungsversorgung) des EtherCAT-Geräts erforderlich
- Kontrolle des neuen FPGA-Standes

HINWEIS

Beschädigung des Gerätes möglich!

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

8.3.5 Gleichzeitiges Update mehrerer EtherCAT-Geräte

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

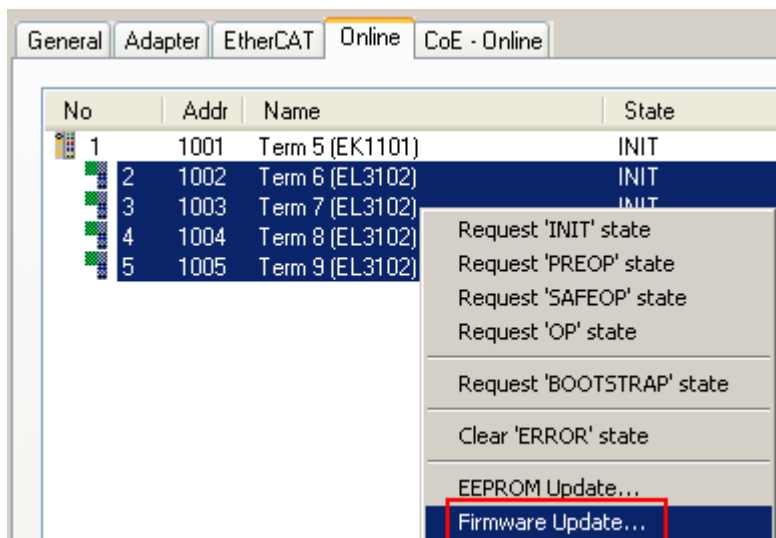


Abb. 87: Mehrfache Selektion und FW-Update

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

8.4 Abkürzungen

Abkürzung	Beschreibung
CAN	Controller Area Network. In ISO 11898 standardisiertes serielles Bussystem, das als Basistechnologie für CANopen dient
CiA	CAN in Automation e.V.. Internationaler Hersteller- und Nutzerverband mit Sitz in Erlangen/Deutschland.
CoB	Communication Object. CAN-Telegramm mit bis zu 8 Datenbytes.
CoB-ID	Communication Object Identifier. Telegrammadresse (nicht zu verwechseln mit Knotenadresse). CANopen verwendet die 11-Bit Identifier nach CAN 2.0A.
CoE	CANopen over EtherCAT
ESC	EtherCAT Slave Controller
FBM	Feldbus Master
MC	Motion Control
NMT	Network Management. Eines der Dienstelemente der CANopen-Spezifikation. Das Netzwerkmanagement dient zur Netzwerkinitialisierung und zur Knotenüberwachung.
OP	OPERATIONAL
PDO	Process Data Object oder Prozessdatenobjekt. CAN-Telegramm zur Übertragung von Prozessdaten (z. B. E/A-Daten).
PREOP	PRE OPERATIONAL
RxPDO	Empfangs-PDO. PDOs werden immer aus Sicht des jeweiligen Gerätes bezeichnet. So wird ein TxPDO mit Eingangsdaten einer E/A Baugruppe zum RxPDO aus Sicht der Steuerung.
SAFEOP	SAFE OPERATIONAL
SDO	Service Data Object oder Servicedatenobjekt. CAN-Telegramm mit Protokoll zur Kommunikation mit Daten des Objektverzeichnisses (typisch Parameterdaten).
SI	Subindex
SM	SyncManager
SoE	Servo Profile over EtherCAT
TxPDO	Sende-PDO (aus Sicht des CAN-Knotens bezeichnet).

8.5 Support und Service

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

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: www.beckhoff.com

Dort finden Sie auch weitere Dokumentationen zu Beckhoff Komponenten.

Support

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

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

Hotline: +49 5246 963 157
E-Mail: support@beckhoff.com
Internet: www.beckhoff.com/support

Service

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

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

Hotline: +49 5246 963 460
E-Mail: service@beckhoff.com
Internet: www.beckhoff.com/service

Unternehmenszentrale Deutschland

Beckhoff Automation GmbH & Co. KG

Hülshorstweg 20
33415 Verl
Deutschland

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

Trademark statements

Beckhoff®, ATRO®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, MX-System®, Safety over EtherCAT®, TC/BSD®, TwinCAT®, TwinCAT/BSD®, TwinSAFE®, XFC®, XPlanar® and XTS® are registered and licensed trademarks of Beckhoff Automation GmbH.

Third-party trademark statements

CANopen and CANopen FD are registered trademarks of CAN in AUTOMATION - International Users and Manufacturers Group e.V.

DeviceNet and EtherNet/IP are trademarks of ODVA, Inc.

PCI Express®, PCIe®, PCI™ and PCI HOT PLUG™ are trademarks or registered trademarks and/or service marks of PCI-SIG.

Mehr Informationen:
www.beckhoff.com/EL6752

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

