

Dokumentation | DE

Feldbus Box für CANopen





# Inhaltsverzeichnis

<b>1</b>	<b>Vorwort</b> .....	<b>5</b>
1.1	Hinweise zur Dokumentation .....	5
1.2	Sicherheitshinweise .....	6
1.3	Ausgabestände der Dokumentation .....	7
<b>2</b>	<b>Systemübersicht</b> .....	<b>9</b>
2.1	Das Feldbus Box System .....	9
2.2	Feldbus Box - Bezeichnungsübersicht .....	11
2.3	Firm- und Hardware-Stand .....	13
<b>3</b>	<b>CANopen</b> .....	<b>14</b>
3.1	CANopen Einführung.....	14
3.2	CANopen Verkabelung.....	16
3.3	Technische Daten.....	24
3.4	CANopen Protokoll .....	25
3.4.1	Netzwerkmanagement.....	25
3.4.2	Prozessdatenobjekte (PDO).....	29
3.4.3	PDO-Parametrierung.....	37
3.4.4	Servicedatenobjekte (SDO).....	39
3.4.5	Identifizier-Verteilung.....	42
3.5	CANopen Objektverzeichnis.....	45
3.5.1	Objektverzeichnis - Struktur.....	45
3.5.2	Objektverzeichnis - Übersicht .....	46
3.5.3	Objekte und Daten.....	49
<b>4</b>	<b>Parametrierung und Inbetriebnahme</b> .....	<b>93</b>
4.1	Anlaufverhalten der Feldbus Box .....	93
4.2	Adresse .....	94
4.3	Baud-Rate .....	95
4.4	Mapping der Feldbus Boxen.....	97
4.5	Konfiguration Feldbus.....	99
4.5.1	Konfigurationsdateien .....	99
4.5.2	Überblick.....	100
4.5.3	Konfiguration mit TwinCAT .....	102
4.5.4	Konfiguration mit Fremdsteuerungen .....	108
4.6	Konfiguration der komplexen E/A Module .....	109
4.6.1	Konfigurations-Software KS2000.....	109
4.6.2	Parametrierung durch Register.....	110
<b>5</b>	<b>Fehlerbehandlung und Diagnose</b> .....	<b>117</b>
5.1	LEDs .....	117
5.2	Diagnose-LEDs für lokale Fehler.....	121
5.3	Überprüfung der IP-Link-Verbindung.....	123
5.4	Emergency-Objekt.....	126
5.5	CANopen Trouble Shooting.....	131
<b>6</b>	<b>Anhang</b> .....	<b>134</b>
6.1	Schnelleinstieg für erfahrene Anwender.....	134

6.2	CAN Identifier-Liste .....	138
6.3	CANopen Baudrate und Bit Timing .....	161
6.4	Automatisches PDO-Mapping .....	162
6.5	Allgemeine Betriebsbedingungen .....	165
6.6	Zulassungen .....	167
6.7	Prüfnormen für die Geräteprüfung .....	168
6.8	Literaturverzeichnis .....	169
6.9	Abkürzungsverzeichnis .....	171
6.10	Support und Service .....	172

# 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, für jede Installation und Inbetriebnahme die zu dem betreffenden Zeitpunkt veröffentlichte Dokumentation zu verwenden.

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

### Disclaimer

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

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

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

### Marken

Beckhoff®, TwinCAT®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® und XPlanar® sind eingetragene und lizenzierte Marken der Beckhoff Automation GmbH. Die Verwendung anderer in dieser Dokumentation enthaltenen Marken oder Kennzeichen durch Dritte kann zu einer Verletzung von Rechten der Inhaber der entsprechenden Bezeichnungen führen.

### Patente

Die EtherCAT-Technologie ist patentrechtlich geschützt, insbesondere durch folgende Anmeldungen und Patente: EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702 mit den entsprechenden Anmeldungen und Eintragungen in verschiedenen anderen Ländern.



**EtherCAT®**

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

### Copyright

© Beckhoff Automation GmbH & Co. KG, Deutschland.

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

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

## 1.2 Sicherheitshinweise

### Sicherheitsbestimmungen

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

### Haftungsausschluss

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

### Qualifikation des Personals

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

### Erklärung der Hinweise

In der vorliegenden Dokumentation werden die folgenden Hinweise verwendet.  
Diese Hinweise sind aufmerksam zu lesen und unbedingt zu befolgen!

#### **GEFAHR**

##### **Akute Verletzungsgefahr!**

Wenn dieser Sicherheitshinweis nicht beachtet wird, besteht unmittelbare Gefahr für Leben und Gesundheit von Personen!

#### **WARNUNG**

##### **Verletzungsgefahr!**

Wenn dieser Sicherheitshinweis nicht beachtet wird, besteht Gefahr für Leben und Gesundheit von Personen!

#### **VORSICHT**

##### **Schädigung von Personen!**

Wenn dieser Sicherheitshinweis nicht beachtet wird, können Personen geschädigt werden!

#### **HINWEIS**

##### **Schädigung von Umwelt/Geräten oder Datenverlust**

Wenn dieser Hinweis nicht beachtet wird, können Umweltschäden, Gerätebeschädigungen oder Datenverlust entstehen.



##### **Tipp oder Fingerzeig**

Dieses Symbol kennzeichnet Informationen, die zum besseren Verständnis beitragen.

### 1.3 Ausgabestände der Dokumentation

Version	Änderungen
1.2.3	Kapitel Baud-Rate auf Firmware-Stand (C)6 aktualisiert
1.2.2	Systemübersicht aktualisiert
1.2.1	Beschreibung des Emergency-Objekts aktualisiert
1.2	Dokumentation entspricht Firmwarestand (C)5
1.1	Erweiterung der Spezifikation des IP-Links auf 15 Meter
1.0	<ul style="list-style-type: none"> <li>• Dokumentation vervollständigt</li> <li>• Dokumentation entspricht Firmwarestand (C)4</li> <li>• Signalvarianten und Anschlussbelegung der Signale in die Feldbus-neutrale Dokumentation <i>Signalvarianten (Feldbus Box E/A-Module)</i> ausgelagert, die Sie im Internet unter <a href="http://www.beckhoff.de">http://www.beckhoff.de</a> im Bereich <i>Download</i> finden.</li> </ul>
0.5	erste Doku-Version für IL23xx-B510 und IPxxxx-B510

Die Dokumentation bezieht sich auf einen Hardwarestand und Softwarestand zum Zeitpunkt der Erstellung der Dokumentation. Die Eigenschaften werden weiterentwickelt und verbessert. Module älteren Fertigungsstand können nicht die gleichen Eigenschaften haben wie Module neueren Standes. Bestehende Eigenschaften bleiben jedoch in der Regel erhalten und werden nicht geändert, so dass diese Module durch neue ersetzt werden können.

Den Software/Hardware Stand des Feldbus Box Moduls zum Zeitpunkt der Fertigung können Sie anhand der Datumsnummer "D." auf der Modulseite erkennen.

- D: kk jj xy zu  
 kk - Kalenderwoche  
 jj - Jahr  
 x - Softwarestand Busplatine  
 y - Hardwarestand Busplatine  
 z - Softwarestand I/O Platine  
 u - Softwarestand I/O Platine

**Beispiel:**

22011501  
 Kalenderwoche 22 des Jahres 2001, Software Busplatine Stand 1, Hardware Stand Busplatine 5, Softwarestand I/O 0 - keine Software für diese Platine notwendig, Hardware I/O Stand 1

Der aktuell vorhandene Firmwarestand lässt sich per Objekt 0x100A (Software Version) [► 52] auslesen.

**Übersicht Firmware-Versionen**

Bei Bedarf kann ein Firmware-Update über die serielle Schnittstelle (spezielles Kabel erforderlich) oder - ab Firmwarestand (C)1 - auch mit der Beckhoff CANopen Karte FC5101 über den Bus erfolgen. Firmware und Update-Tool finden Sie im Internet unter <http://www.beckhoff.de>.

Firmware	Änderung, Erweiterung	Bugfix
C(6)	<ul style="list-style-type: none"> <li>Unterstützung fester Baudraten</li> </ul>	
C(5)	<ul style="list-style-type: none"> <li>Durchsynchronisierung K-Bus Zyklus mit SYNC-Telegramm optimiert</li> <li>Wenn K-Bus Zyklus noch nicht abgeschlossen bevor nächstes SYNC Telegramm eintrifft, so wird ein Emergency Telegramm verschickt und die Tx-Overrun LED blinkt langsam. LED Signal und EMCY wird 10 Sekunden nach letztem Auftreten dieser Situation zurückgesetzt.</li> <li>KS2000 Online Mode wird unterstützt</li> </ul>	<ul style="list-style-type: none"> <li>Übergang zu Operational wird bei anstehendem K-Bus Fehler verweigert.</li> <li>Boot-Up Message kommt auch bei niedrigen Baudraten zuverlässig</li> <li>Default Mapping für Node-ID 64</li> </ul>
(C)4	<ul style="list-style-type: none"> <li>Neu: Objekt 0x6126 Interrupt Maske. Ermöglicht die Auswahl der Datenänderungen, die zum Versenden ereignisgesteuerter TxPDOs führen. Keine Änderung am Default Verhalten.</li> <li>SDO Antwortzeiten auf Objekte mit PDO Parametern (0x1400ff, 0x1800ff, 0x5500) drastisch verkürzt.</li> </ul>	<ul style="list-style-type: none"> <li>Life-Time-Factor 2 führt bei korrektem Guarding nicht mehr zum Guard-Fehler.</li> <li>RxPDOs mit Länge 0 führen nicht mehr zum Stop der Firmware</li> <li>Boot-Up Nachricht wird nun erst gesendet, wenn Koppler den Pre-Operational Zustand erreicht hat (nicht schon während des Statusübergangs)</li> </ul>
(C)2		<ul style="list-style-type: none"> <li>1 Waitstate für RAM Zugriff eingeführt - damit ist C2 auch auf Modulen alten Hardwarestands zuverlässig lauffähig.</li> </ul>
(C)1	<ul style="list-style-type: none"> <li>Firmwaredownload nun auch via CAN möglich (hierzu wurde Objekt 0x5FFF eingeführt). Erfordert Beckhoff CANopen PCI Karte FC510x.</li> </ul>	
(C)0	erstes Release	

Nicht aufgeführte Firmwareversionen wurden nur für interne Tests genutzt.

## 2 Systemübersicht

### 2.1 Das Feldbus Box System

Feldbus Box Module sind robuste Feldbusstationen für viele verschiedene Feldbus-Systeme. Sie verfügen über vielfältige E/A-Funktionalität. Alle relevanten Industriesignale werden unterstützt. Neben digitalen und analogen Ein- und Ausgängen samt Thermoelement- und RTD-Eingängen gibt es Inkrementalencoder-Interfaces für die Weg- und Winkelmessung sowie serielle Schnittstellen für die Lösung vielfältiger Kommunikationsaufgaben.

#### Signalanschluss in 3 Varianten

Die digitalen Ein-/Ausgänge können wahlweise über 8 mm Schnapp-Steckverbinder, M8 Schraub-Steckverbinder oder M12 Schraub-Steckverbinder angeschlossen werden. Für analoge Signale ist die M12-Variante vorgesehen.

#### Alle wichtigen Signalformen

Spezielle Ein-/Ausgangskanäle auf den Kombi-E/A-Module lassen sich wahlweise als Ein- oder Ausgang nutzen. Eine Konfiguration ist nicht erforderlich, da die Feldbusschnittstelle für jeden Kombikanal sowohl Eingangs- als auch Ausgangsdaten zur Verfügung stellt. Durch die Kombi-Module hat der Anwender alle Vorteile einer feinen Signal-Granularität.

Die Prozessor-Logik, die Eingangsbeschaltung sowie die Sensorversorgung werden aus der Steuerspannung gespeist. Die Lastspannung für die Ausgänge kann separat zugeführt werden. Bei Feldbus Boxen, in denen nur Eingänge zur Verfügung stehen, kann die Lastversorgung UP zur Weiterleitung optional angeschlossen werden.

Die Zustände der Feldbus Box, der Feldbusverbindung, der Spannungsversorgung sowie der Signale werden von LEDs angezeigt.

Die Beschriftungstreifen lassen sich extern maschinell beschriften und können dann eingeschoben werden.

#### Kombinierbare Feldbus Boxen für mehr Flexibilität

Die Feldbus Box Serie umfasst neben der Kompakt Box auch erweiterbare Geräte, die Koppler Box und die Erweiterungsbox sowie intelligente Geräte, die SPS Boxen.

#### Kompakt Box

Die Kompakt Box stellt dem Feldbus die E/A-Daten der angeschlossenen digitalen und analogen Sensoren und Aktuatoren zur Verfügung.

#### Koppler Box

Die Koppler Box sammelt zusätzlich über eine störsichere LWL-Verbindung (IP-Link) E/A-Daten von den Erweiterungsboxen ein. Sie können bis zu 120 Erweiterungsboxen an eine Koppler Box anschließen. Es ergibt sich so ein verteiltes IP67 E/A-Netzwerk mit nur einer Feldbusschnittstelle.

Die Koppler Box erkennt die angeschlossenen Erweiterungsmodule selbsttätig in der Aufstart-Phase und mappt die E/A-Daten automatisch in das Feldbus-Prozessabbild – eine Konfiguration ist nicht erforderlich. Aus Sicht des Feldbusses stellt sich die Koppler Box samt allen vernetzten Erweiterungsboxen als ein einziger Busteilnehmer mit entsprechend vielen E/A-Signalen dar.

Die Koppler Box entspricht dem Buskoppler aus dem BECKHOFF Busklemmen-System. BECKHOFF Feldbus-Geräte der Schutzart IP 20 (Busklemmen) und IP 67 (Feldbus Box) können problemlos kombiniert werden - das Daten-Handling ist jeweils gleich.

## **IP-Link**

IP-Link ist eine LWL-Verbindung mit 2 Megabit/s Übertragungsrate, die 1000 binäre E/A-Daten in ca. 1 ms schnell und sicher überträgt – kleinere Konfigurationen entsprechend schneller. Durch die hohe Nutzdatenrate ist gewährleistet, dass die Kopplung über IP-Link keine spürbare Einschränkung der Feldbus-Performance mit sich bringt.

Für die schnelle und einfache Konfektionierung der IP-Link-Kabel vor Ort wurden preisgünstige Steckverbinder in IP 67 Schutzart entwickelt. Der Anschluss erfordert kein Spezialwerkzeug und lässt sich schnell und einfach durchführen. Die IP-Link-Kabel sind alternativ auch mit vorkonfektionierten Steckern erhältlich.

Durch die getrennte Zuführung der Ausgangs-Spannungsversorgung lassen sich Ausgangsgruppen einzeln abschalten. Außerdem können problemlos unterschiedliche Potenziale innerhalb eines Erweiterungsringes aufgebaut werden, da IP-Link naturgemäß über eine optimale Potenzialtrennung verfügt.

## **Erweiterungsbox**

Die Erweiterungsboxen decken wie die Kompakt Boxen das gesamte Spektrum der E/A-Signale ab und dürfen bis zu 15 m Abstand voneinander haben. Sie bauen hierbei besonders klein und führen zu besonders preiswerten E/A-Lösungen in hoher Schutzart. Auch hier sind die digitalen Ein-/Ausgänge wahlweise über 8 mm Schnapp-Steckverbinder, M8 Schraub-Steckverbinder oder M12 Schraub-Steckverbinder anschließbar. Analoge Signaltypen werden mit der M12-Variante ausgerüstet. Die Schnapp-Steckverbinder verriegeln formschlüssig und ergeben eine rüttelfeste Verbindung, während sich die Schraub-Steckverbinder durch eine hohe Zugfestigkeit auszeichnen.

## **SPS Box**

Die SPS Box ist eine intelligente Feldbus Box mit SPS-Funktionalität für dezentrale Vorverarbeitung der E/A-Signale. Hiermit lassen sich Applikationsteile aus der zentralen Steuerung auslagern. Deren CPU und der Feldbus werden entlastet. Dezentral Zählen, Regeln oder Schalten sind typische Anwendungen für die SPS-Box. Die Reaktionszeiten sind unabhängig von der Buskommunikation und der übergeordneten Steuerung.

Bei Bus- oder Steuerungsausfall ist ein Funktionserhalt (z.B. geordnete Überführung des Prozesses in einen sicheren Zustand) möglich.

Die Programmierung erfolgt mit TwinCAT nach IEC 61131-3. Fünf verschiedene Programmiersprachen stehen zur Verfügung:

- Anweisungsliste (AWL)
- Funktionsplan (FUP)
- Kontaktplan (KOP)
- Ablaufsprache (AS)
- Strukturierter Text (ST).

Der Programm-Download erfolgt wahlweise über den Feldbus oder über die Programmierschnittstelle.

Es stehen umfangreiche Debug-Funktionalitäten (Breakpoint, Einzelschritt, Monitoring, etc.) zur Verfügung. SPS Box verfügt über einen leistungsfähigen 16 Bit Controller, 32/96 KByte Programmspeicher und 32/64 KByte Datenspeicher. Weiter stehen 512 Byte als nichtflüchtiger Speicher für remanente Merker zur Verfügung.

## **SPS Box mit IP-Link**

Nahezu unbeschränkte E/A-Möglichkeiten ergeben sich durch die programmierbare SPS Box mit IP-Link. Aus dem SPS-Programm heraus lassen sich bis zu 120 Erweiterungsmodule mit über 2000 E/As direkt ansprechen. Die SPS Box eignet sich damit auch als autarke Kleinststeuerung zur Steuerung von Anlagenteilen oder kleiner Maschinen.

## 2.2 Feldbus Box - Bezeichnungsübersicht

Die Bezeichnung der Feldbus Box Module ist wie folgt zu verstehen:  
IXxxxzyyyy

### IX beschreibt die Bauform:

"IP" steht für die Bauform Kompakt Box [► 12].

"IL" steht für die Bauform Koppler Box (mit IP-Link) [► 12].

"IE" steht für die Bauform Erweiterungsbox [► 12].

### xxxxy beschreibt die E/A-Beschaltung:

xxx bezeichnet die E/A-Eigenschaft:

"10x" - 8 x digitale Eingänge

"15x" - Zählermodul

"20x" - 8 x digitale Ausgänge

"23x" - 4 x digitale Eingänge und 4 x digitale Ausgänge

"24x" - 8 x digitale Eingänge und 8 x digitale Ausgänge

"25x" - PWM-Modul

"3xx" - 4 x analoge Eingänge

"4xx" - 4 x analoge Ausgänge

"5xx" - Inkremental-Encoder oder SSI-Geber

"6xx" - Gateway-Module RS232, RS422, RS485, TTY

y beschreibt den mechanischen Anschluss:

"0" steht für 8mm Schnappanschluss

"1" steht für M 8 Schraubanschluss

"2" steht für M 12 Schraubanschluss und

"9" steht für M23 Schraubanschluss

### zyyy bezeichnet die Programmierbarkeit und das Feldbus-System:

z unterscheidet ob es sich um einen Slave oder einen programmierbare Slave handelt:

"B" - nicht programmierbar

"C" - programmierbar (SPS Box [► 12])

yyy steht für das Feldbus-System und den Bus-Anschluss:

"110" - EtherCAT

"200" - Lightbus

"310" - PROFIBUS

"318" - PROFIBUS mit integriertem T-Stück

"400" - Interbus

"510" - CANopen

"518" - CANopen mit integriertem T-Stück

"520" - DeviceNet

"528" - DeviceNet mit integriertem T-Stück

"730" - Modbus

"800" - RS485

"810" - RS232

"900" - Ethernet TCP/IP mit RJ45 für den Bus-Anschluss

"901" - Ethernet TCP/IP mit M12 für den Bus-Anschluss

"903" - PROFINET

"905" - EtherNet/IP

**Kompakt Box**

Kompakt Box

Die Feldbus Boxe verfügen über vielfältige E/A-Funktionalität. Alle relevanten Industriesignale werden unterstützt. Die digitalen Ein-/Ausgänge können wahlweise über Ø 8 mm Schnapp-, M8 Schraub- oder über M12 Schraub-Steckverbinder angeschlossen werden. Für analoge Signale ist die M12 Variante verfügbar.

Je nach Modul kann der E/A-Teil und der Stromversorgungsteil unterschiedlich sein.

**Koppler Box**

Koppler Box

Die Koppler Box gibt es in drei Varianten als IL230x-Bxxx. Diese unterscheidet sich von der Kompakt Box dadurch, dass diese Module eine Schnittstelle zu den sogenannten Erweiterungsboxen bietet. Diese Schnittstelle ist ein Sub-Bussystem auf LWL Basis den sogenannten IP-Link. Dieses leistungsfähige Sub-Bussystem kann bis zu 120 Erweiterungsboxen an einer Koppler Box verarbeiten.

**Erweiterungsbox**

Erweiterungsbox

Feldbusunabhängige Erweiterungsmodule, die nur an einer Koppler Box über IP-Link betrieben werden können.

**SPS Box**

SPS Box

Eine SPS Box unterscheidet sich von einer Koppler Box dadurch, dass sie in IEC 61131-3 programmierbar ist. Dadurch kann dieser Slave auch ohne Master autonom arbeiten, zum Beispiel für Steuerungs- oder Regelungsaufgaben.

## 2.3 Firm- und Hardware-Stand

Diese Dokumentation bezieht sich auf den zum Zeitpunkt ihrer Erstellung gültigen Hard- und Firmware-Stand. Die Eigenschaften werden weiterentwickelt und verbessert. Module älteren Fertigungsstandes können nicht die gleichen Eigenschaften haben wie Module neue Standes. Bestehende Eigenschaften bleiben jedoch erhalten und werden nicht geändert, so dass diese Module immer durch neue ersetzt werden können.

Den Firm- und Hardware-Stand der Module können Sie anhand der mit einem *D* beginnenden Nummer auf der Seite des Moduls erkennen.

### Syntax:

D . ww yy x y z u

ww - Kalenderwoche

yy - Jahr

x - Firmware-Stand der Busplatine

y - Hardware-Stand der Busplatine

z - Firmware-Stand der E/A-Platine

u - Hardware-Stand der E/A-Platine

### Beispiel:

D.22081501

- Kalenderwoche 22

- des Jahres 2008

- Firmware-Stand Busplatine: 1

- Hardware Stand Busplatine: 5

- Firmware-Stand E/A-Platine: 0 (keine Firmware für diese Platine notwendig)

- Hardware-Stand E/A-Platine: 1

## 3 CANopen

### 3.1 CANopen Einführung



Abb. 1: CANopenLogo

CANopen ist eine weit verbreitete CAN-Anwendungsschicht, die im Verband CAN-in-Automation (CiA, <http://www.can-cia.org>) entwickelt und inzwischen zur internationalen Normung angenommen wurde.

#### Gerätemodell

CANopen besteht aus der Protokolldefinition (Kommunikationsprofil) so wie den Geräteprofilen, die den Dateninhalt für die jeweilige Gerätekategorie normieren. Zur schnellen Kommunikation der Ein- und Ausgangsdaten dienen die Prozessdatenobjekte (PDO) [► 29]. Die CANopen-Geräteparameter und Prozessdaten sind in einem Objektverzeichnis strukturiert. Der Zugriff auf beliebige Daten dieses Objektverzeichnisses erfolgt über die Servicedatenobjekte (SDO). Weiter gibt es einige Spezialobjekte (bzw. Telegrammarten) für Netzwerkmanagement (NMT), Synchronisation, Fehlermeldungen etc.

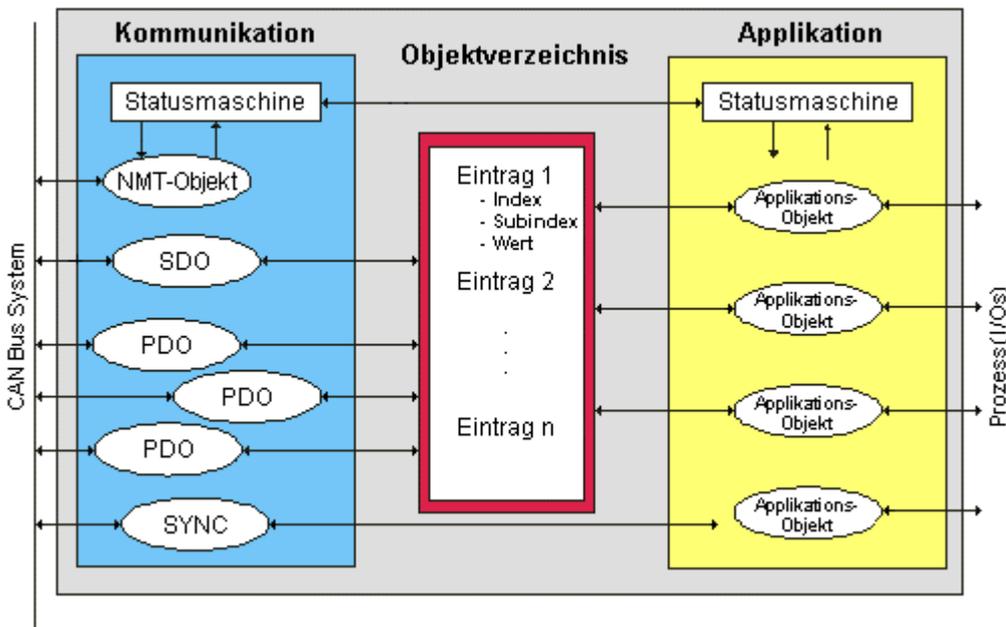


Abb. 2: CANopen-Gerätemodell

#### Kommunikationsarten

CANopen definiert mehrere Kommunikationsarten für die Ein- und Ausgangsdaten (Prozessdatenobjekte):

- Freignisgesteuert [► 32]: Telegramme werden versendet, sobald sich der Inhalt geändert hat. Hier wird nicht ständig das Prozessabbild, sondern nur die Änderung desselben übertragen.
- Zyklisch synchron [► 32]: Über ein SYNC Telegramm werden die Baugruppen veranlasst, die vorher empfangenen Ausgangsdaten zu übernehmen und neue Eingangsdaten zu senden.

- Angefordert (gepollt): Über ein CAN Datenanforderungstelegramm werden die Baugruppen veranlasst ihre Eingangsdaten zu senden.

Die gewünschte Kommunikationsart wird über den Parameter Transmission Type eingestellt.

### Geräteprofil

Die BECKHOFF CANopen-Geräte unterstützen alle E/A- Kommunikationsarten und entsprechen dem Geräteprofil für digitale und analoge Ein-/Ausgabebaugruppen (DS401 Version 1). Aus Gründen der Abwärtskompatibilität wurde das Default Mapping nicht der Profilversion DS401 V2 angepasst.

### Übertragungsraten

Übertragungsraten [► 161]

Neun Übertragungsraten von 10 kBaud bis 1 MBaud stehen für unterschiedliche Buslängen zur Verfügung. Durch die effektive Nutzung der Busbandbreite erreicht CANopen kurze Systemreaktionszeiten bei vergleichsweise niedrigen Datenraten.

### Topologie

Topologie [► 16]

CAN basiert auf einer linienförmigen Topologie. Die Anzahl der Teilnehmer pro Netz ist dabei von CANopen logisch auf 128 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 1 MBaud ist z.B. eine Netzausdehnung von 25 m, bei 50 kBaud eine Netzausdehnung von 1000 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 CANopen Parameter komfortabel eingestellt werden. Für die Parametrierung der Beckhoff CANopen-Geräte mit Konfigurationstools dritter Hersteller steht Ihnen auf der Beckhoff Website (<http://www.beckhoff.de>) ein eds-File (electronic data sheet) zur Verfügung.

### Zertifizierung

Die Beckhoff CANopen-Geräte verfügen über eine leistungsfähige Protokollimplementierung und sind vom Verband CAN-in-Automation (<http://www.can-cia.org>) zertifiziert.

## 3.2 CANopen Verkabelung

### Kapitelübersicht

[CAN-Topologie](#) [► 16]

Buslänge

[Stichleitungen](#) [► 17]

[Sternverteiler](#) [► 18]

[CAN-Kabel](#) [► 18]

[Schirmung](#) [► 19]

[Kabelfarben](#) [► 20]

BK5151, EL6751, FC51xx, CX805x, CX-B/M510: D-Sub 9polig

BK51x0/BX5100: 5-poliger Open Style Connector

LC5100 Busanschluss

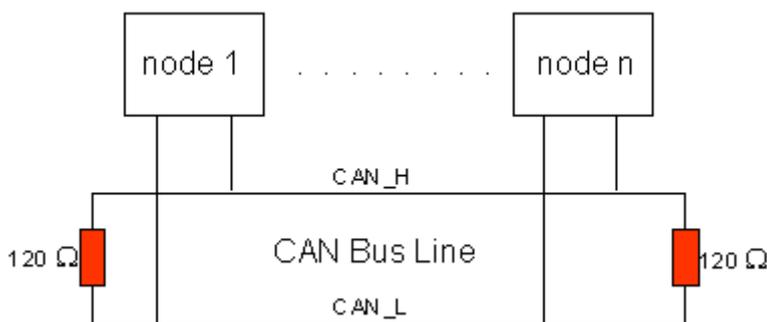
Feldbus Box: M 12 CAN Buchse

Hinweise für die Überprüfung der CAN-Verdrahtung finden sich im Kapitel [Fehlersuche / Trouble Shooting](#) [► 131].

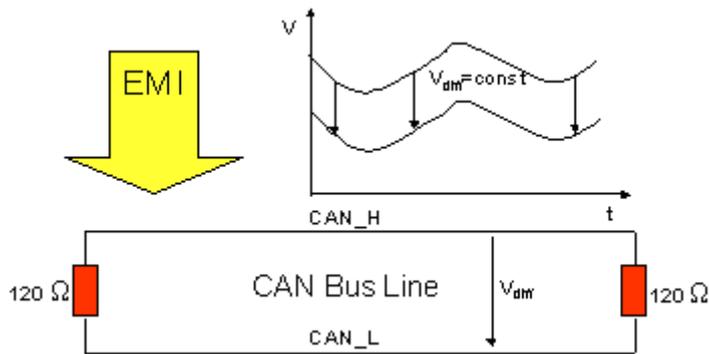
### CAN-Topologie

CAN-Topologie

CAN ist ein 2-Draht-Bussystem, an dem alle Teilnehmer parallel (d.h. mit kurzen Stichleitungen) angeschlossen werden. 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!



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



**Buslänge**

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
1 MBit/s	< 20 m*
500 kBit/s	< 100 m
250 kBit/s	< 250 m
125 kBit/s	< 500 m
50 kBit/s	< 1000 m
20 kBit/s	< 2500 m
10 kBit/s	< 5000 m

\*) Häufig findet man in der Literatur für CAN die Angabe 40 m bei 1 MBit/s. Dies gilt jedoch nicht für Netze mit optoentkoppelten CAN-Controllern. Die worst case Berechnung mit Optokopplern ergibt bei 1 MBit/s eine maximale Buslänge von 5m - erfahrungsgemäß sind jedoch 20 m problemlos erreichbar.

Bei Buslängen über 1000 m kann der Einsatz von Repeatern notwendig werden.

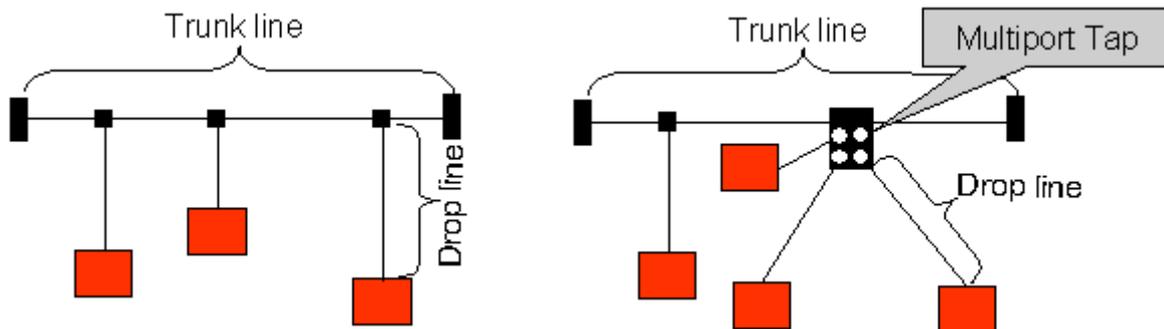
**Stichleitungen**

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 [▶ 161] kann dies angenommen werden, wenn folgende Stichleitungslängen nicht überschritten werden:

Baud-Rate	Länge Stichleitung	gesamte Länge aller Stichleitungen
1 MBit/s	< 1m	< 5 m
500 kBit/s	< 5 m	< 25 m
250 kBit/s	< 10m	< 50 m
125 kBit/s	< 20m	< 100 m
50 kBit/s	< 50m	< 250 m

Stichleitungen dürfen nicht mit Abschlusswiderständen versehen werden.



### Sternverteiler (Multiport Tap)

#### 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:

Baud-Rate	Länge Stichleitung bei Multiport Topologie	Länge Trunk Line (ohne Stichleitungen)
1 MBit/s	< 0,3 m	< 25 m
500 kBit/s	< 1,2 m	< 66 m
250 kBit/s	< 2,4 m	< 120 m
125 kBit/s	< 4,8 m	< 310 m

### CAN-Kabel

#### CAN-Kabel

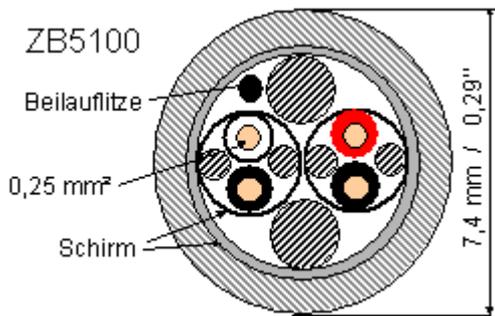
Für die CAN-Verdrahtung wird die Verwendung von paarig verdrehten, geschirmten Kabeln (2x2) mit einem Wellenwiderstand von 108...132 Ohm empfohlen. Wenn das Bezugspotential der CAN-Transceiver (CAN-Ground) nicht verbunden werden soll, so kann auf das zweite Adernpaar verzichtet werden (nur bei kleinen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

### ZB5100 CAN-Kabel

#### ZB5100

Beckhoff hat ein hochwertiges CAN-Kabel mit folgenden Eigenschaften im Programm:

- 2 x 2 x 0,25 mm<sup>2</sup> (AWG 24) paarig verseilt, Kabelfarben: rot/schwarz + weiß/schwarz
- doppelt geschirmt
- Schirmgeflecht mit Beilaufitze (kann direkt auf Pin3 der 5-pol Anschlussklemme aufgelegt werden),
- flexibel (Mindestbiegeradius 35mm bei einmaligem Biegen, 70mm bei mehrmaligem Biegen)
- Wellenwiderstand (60kHz): 120 Ohm
- Leiterwiderstand < 80 Ohm/km
- Mantel: PVC grau, Außendurchmesser 7,3 +/- 0,4 mm
- Gewicht: 64 kg/km.
- Bedruckt mit "Beckhoff ZB5100 CAN-BUS 2x2x0.25" und Metermarkierung (Längenangaben, alle 20cm)

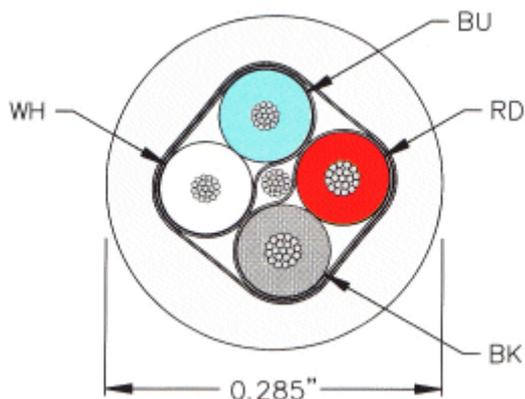


**ZB5200 CAN/DeviceNet-Kabel**

ZB5200

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 mm<sup>2</sup> (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



**Schirmung**

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.

Hinweise für die Überprüfung der CAN-Verdrahtung finden sich im Kapitel Fehlersuche / Trouble Shooting [► 131].

**Kabelfarben**

Kabelfarben

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

Pin BK51x0 PIN BX5100 (X510)	Pin BK5151 CX8050, CX8051, CXxxxx- B510/M510	Pin Feldbus Box	Pin FC51xx	Funktion	Kabelfarbe ZB5100	Kabelfarbe ZB5200
1	3	3	3	CAN Ground	<b>schwarz / (rot)</b>	<b>schwarz</b>
2	2	5	2	CAN Low	<b>schwarz</b>	<b>blau</b>
3	5	1	5	Schirm	Beilaufritze	Beilaufritze
4	7	4	7	CAN high	<b>weiß</b>	<b>weiß</b>
5	9	2	9	nicht benutzt	<b>(rot)</b>	<b>(rot)</b>

**BK5151, FC51xx, CX mit CAN Interface und EL6751: D-Sub 9polig**

BK5151, FC51xx, CX mit CAN Interface und EL6751: D-Sub 9polig

Die CAN Busleitung wird an die FC51x1, FC51x2 CANopen Karten und bei der EL6751 CANopen Master-/Slaveklemme über 9polige Sub-D-Buchsen mit folgender Steckerbelegung angeschlossen.

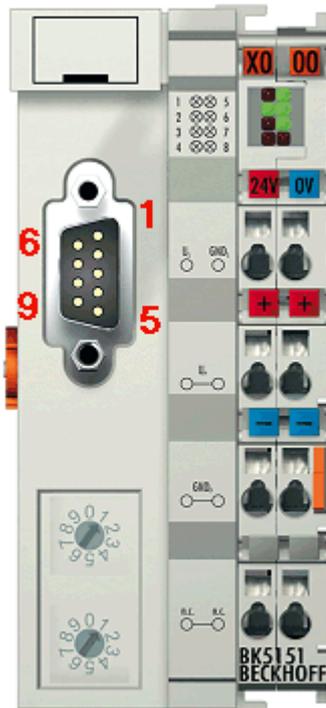
Pin	Belegung
2	CAN low (CAN-)
3	CAN Ground (intern verbunden mit Pin 6)
6	CAN Ground (intern verbunden mit Pin 3)
7	CAN high (CAN+)

Die nicht aufgeführten Pins sind nicht verbunden.

Die Hutschielenkontaktfeder und der Steckerschirm sind durchverbunden.

Hinweis: an Pin 9 darf eine Hilfsspannung bis 30 V<sub>DC</sub> angeschlossen sein (wird von manchen CAN Geräten zur Versorgung der Transceiver genutzt).

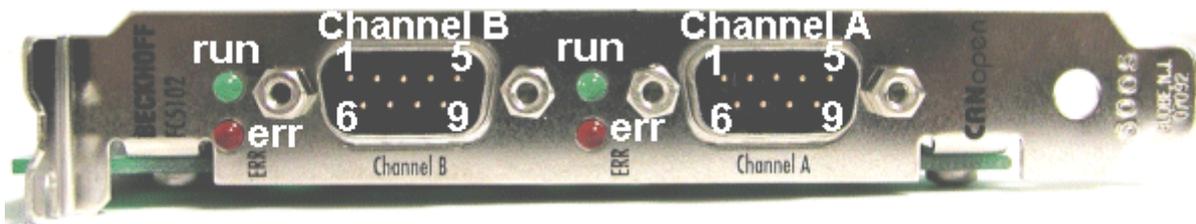
BK5151



EL6751



FC51x2:



**BK51x0/BX5100: 5poliger Open Style Connector**

BK51x0/BX5100: 5poliger Open Style Connector

Bei den BK51x0/BX5100 (X510) Buskopplern befindet sich auf der linken Seite eine abgesenkte Frontfläche mit einem 5poligen Stecker.

Hier kann die mitgelieferte CANopen- Verbindungsbuchse eingesteckt werden.

- 5: reserv.
- 4: CAN high
- 3: Shield
- 2: CAN low
- 1: CAN Ground



Das linke Bild zeigt die Buchse im Buskoppler BK51x0/BX5100. Pin 5 ist dabei der oberste Pin auf der Steckerleiste. Pin 5 ist nicht benutzt. Pin 4 ist die CAN-High-Leitung, Pin 2 die CAN-Low-Leitung und an Pin 3 wird der Schirm angelegt (ist über eine R/C-Schaltung mit der Tragschiene verbunden). Optional kann am Pin 1 CAN-GND angeschlossen werden. Wenn alle CAN-Ground Pins verbunden sind ergibt dies ein gemeinsames Bezugspotential für die CAN Transceiver im Netz. Es empfiehlt sich, CAN-GND an einer Stelle zu erden, damit das gemeinsame CAN Bezugspotential nahe beim Versorgungs-Potential liegt. Da die CANopen Buskoppler BK51X0/BX5100 über eine vollständige galvanische Trennung des Busanschlusses verfügen, kann u.U. auf die Verdrahtung von CAN-Ground verzichtet werden.

### Businterface Connector ZS1052-3000

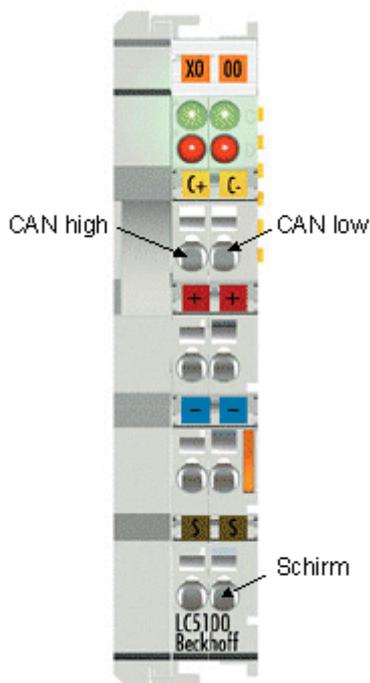
Businterface Connector ZS1052-3000

Alternativ zum mitgelieferten Stecker kann der CAN Interface Connector ZS1052-3000 eingesetzt werden. Dieser vereinfacht die Verdrahtung erheblich. Für die ankommenden und abgehenden Leitungen stehen separate Klemmen zur Verfügung, der Schirm wird durch die Zugentlastung großflächig angeschlossen. Der integrierte Abschlusswiderstand kann von außen zugeschaltet werden. Ist er eingeschaltet, so wird die abgehende Busleitung elektrisch abgetrennt - damit lassen sich Verdrahtungsfehler schnell lokalisieren, und es ist gewährleistet, dass nicht mehr als zwei Widerstände im Netz aktiv sind.

### LC5100: Busanschluss über Federkraftklemmen

LC5100: Busanschluss

Beim Low-Cost-Koppler LC5100 wird die CAN-Leitung direkt auf die Klemmstellen 1 (CAN-H, gekennzeichnet mit C+) bzw. 5 (CAN-L, gekennzeichnet mit C-) aufgelegt. Der Schirm kann optional auf die Klemmstellen 4 bzw. 8 aufgelegt werden, diese sind über eine R/C-Schaltung mit der Tragschiene verbunden.



## HINWEIS

### Achtung

Durch die nicht vorhandene galvanische Trennung kann durch unsachgemäße Verkabelung der CAN Treiber zerstört oder geschädigt werden. Verkabeln Sie immer im ausgeschalteten Zustand, verkabeln Sie erst die Spannungsversorgung und legen Sie erst dann den CAN auf. Überprüfen Sie die Verkabelung und schalten dann erst die Spannung ein.

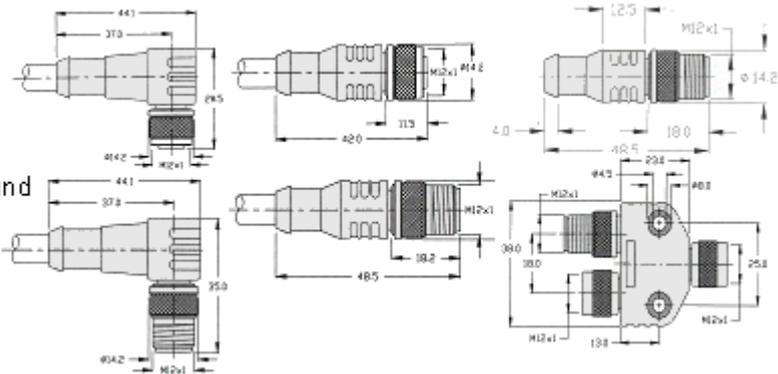
**Feldbus Box: M12 CAN Buchse**

Feldbus Box

Bei der Feldbus Box IPxxxx-B510, IL230x-B510 und IL230x-C510 wird der Busanschluss mit 5poligen M12 Steckverbindern ausgeführt.



- 1: Schirm
- 2: reserviert
- 3: CAN Ground
- 4: CAN high
- 5: CAN low



Für das Feldbus Box System bietet Beckhoff feldkonfektionierbare Stecker, Passivverteiler, Abschlusswiderstände sowie eine große Auswahl an vorkonfektionierten Kabeln an. Details finden sich im Katalog oder unter [www.beckhoff.de](http://www.beckhoff.de).

### 3.3 Technische Daten

Technische Daten	IPxxxx-B51x	IL230x-B510, (IL230x-C510)
Erweiterungsmodule	-	max. 120 mit insgesamt 128 Bytes Ein- und 128 Byte Ausgänge
Digitale Peripheriesignale	entsprechend E/A-Variante	max. 960 Ein- und Ausgänge
Analoge Peripheriesignale	entsprechend E/A-Variante	max. 60 Ein- und Ausgänge
Anzahl PDOs (CANopen)	5 RxPDOs und/oder 5 TxPDOs (entsprechend E/A-Variante)	16 RxPDOs und 16 TxPDOs
PDO-Kommunikationsarten	alle: ereignisgesteuert, zyklisch (event timer), synchron, gepoltt (per RTR)	
Weitere CANopen-Features	Life/Nodeguarding, Heartbeat, Emergency-Objekt, Variables Mapping, Store/Restore,	
Konfigurationsmöglichkeiten	über KS2000 oder die Steuerung (Service Daten Objekte)	
Baud-Raten	automatische Erkennung von 10, 20, 50, 100, 125, 250, 500, 800, 1000 kBaud	
Stromversorgung	Steuerspannung: 24V DC (-15%/+20%); Lastspannung: entsprechend E/A Variante	
Stromaufnahme Steuerspannung	entsprechend E/A- Variante + Stromaufnahme Sensorversorgung, max. 0,5A	
Stromaufnahme Lastspannung	entsprechend E/A- Variante	
Anschluss Stromversorgung	Einspeisung: 1 x M 8 Stecker 4-polig Weiterleitung: 1 x M 8 Buchse 4-polig (außer IP/IE204x)	
Anschluss Feldbus	1 x M12 Stecker 5-polig	
Potentialtrennung	Kanäle / Steuerspannung: nein zwischen den Kanälen: nein Steuerspannung / Feldbus: ja	
Betriebstemperatur	0°C ... +55°C	
Lagertemperatur	-25°C ... +85°C	
Vibrationsfestigkeit	gemäß IEC 68, Teil 2-6 / IEC 68, Teil 2-27	
EMV	gemäß EN 50082-2 / EN 50081-2	
Schutzart	IP 65/66/67 (gemäß EN 60529)	
Einbaulage	beliebig	
Zulassung	UL E172151	
Gewicht	ca. 210 g	

### 3.4 CANopen Protokoll

#### 3.4.1 Netzwerkmanagement

##### Einfacher Boot-Up

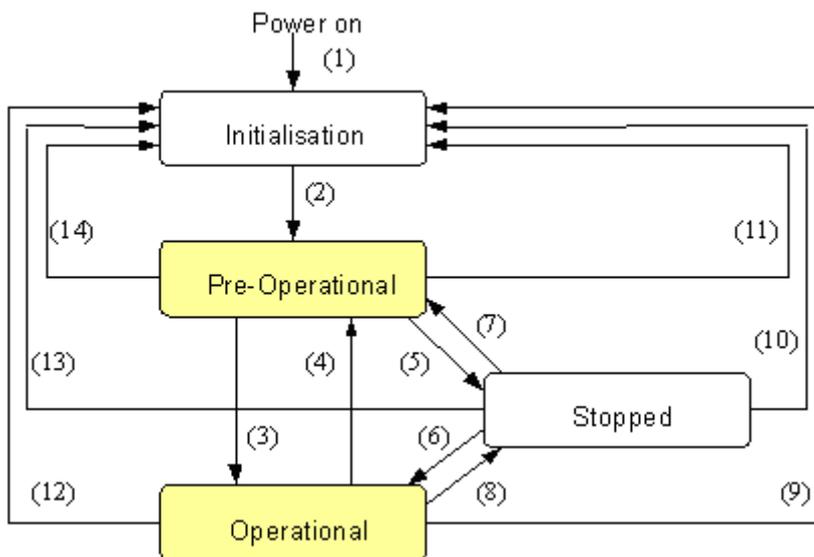
CANopen erlaubt einen sehr einfachen Boot-Up des verteilten Netzwerkes. Die Module befinden sich nach der Initialisierung automatisch im Zustand *Pre-Operational*. In diesem Zustand kann bereits über Service-Datenobjekte (SDOs) mit Default-Identifiern auf das Objektverzeichnis zugegriffen werden, die Module können also konfiguriert werden. Da für alle Einträge im Objektverzeichnis Default-Einstellungen vorhanden sind, kann in den meisten Fällen auf eine Konfiguration verzichtet werden.

Zum Starten der Module ist dann nur eine einzige CAN-Nachricht erforderlich: *Start\_Remote\_Node*: Identifier 0, zwei Datenbytes: 0x01, 0x00. Sie überführt die Knoten in den Zustand *Operational*.

##### Netzwerkstatus

Netzwerkstatus

Die Zustände im CANopen Boot-Up und die Zustandsübergänge sind aus dem Zustandsdiagramm ersichtlich:



##### Pre-Operational

Nach der Initialisierung geht der Buskoppler automatisch, d.h. ohne Befehl von außen, in den Zustand *Pre-Operational* über. In diesem Zustand kann er konfiguriert werden, denn die Servicedatenobjekte (SDOs) sind bereits aktiv. Die Prozessdatenobjekte sind hingegen noch gesperrt.

##### Operational

Im Zustand *Operational* sind auch die Prozessdatenobjekte aktiv.

Wenn der Buskoppler aufgrund äußerer Einflüsse (z.B. CAN-Störung, keine Ausgangs-Spannung) oder innerer Einflüsse (z.B. K-Bus-Fehler) nicht mehr in der Lage ist, Ausgänge zu setzen oder Eingänge zu lesen bzw. zu kommunizieren, so versucht er eine entsprechende Emergency-Nachricht zu senden, geht in den Fehlerzustand und fällt dabei in den Zustand *Pre-Operational* zurück. Damit kann auch die NMT-Statusmaschine des Netzwerkmasters fatale Fehler sofort erkennen.

## Stopped

Im Zustand *Stopped* (früher *Prepared*) ist keine Datenkommunikation mit dem Koppler möglich - lediglich NMT-Nachrichten werden empfangen. Die Ausgänge gehen in den Fehlerzustand.

## Statusübergänge

Statusübergänge

Die Netzwerkmanagement-Nachrichten haben einen sehr einfachen Aufbau: CAN-Identifizier 0 mit zwei Byte Dateninhalt. Das erste Datenbyte enthält den sogenannten Command-Specifier (cs), das zweite Datenbyte die Knotenadresse, wobei die Knotenadresse 0 alle Knoten anspricht (Broadcast).

11-bit Identifizier	2 Byte Nutzdaten						
0x00	cs	Node-ID					

Die folgende Tabelle gibt einen Überblick über alle CANopen Statusübergänge und die dazugehörigen Kommandos (Command Specifier im NMT Master-Telegramm):

Statusübergang	Command Specifier cs	Erläuterung
(1)		- Der Initialisierungs-Status wird beim Einschalten selbsttätig erreicht
(2)		- Nach der Initialisierung wird der Status Pre-Operational automatisch erreicht - dabei wird die Boot-Up-Nachricht abgeschickt.
(3), (6)	cs = 1 = 0x01	Start_Remote_Node. Startet Modul, gibt Ausgänge frei, Startet Übertragung von PDOs.
(4), (7)	cs = 128 = 0x80	Enter_Pre-Operational. Stoppt PDO-Übertragung, SDO weiter aktiv.
(5), (8)	cs = 2 = 0x02	Stop_Remote_Node. Ausgänge gehen in den Fehlerzustand, SDO und PDO abgeschaltet.
(9), (10), (11)	cs = 129 = 0x81	Reset_Node. Führt Reset durch. Alle Objekte werden auf Power-On Defaults zurückgesetzt.
(12), (13), (14)	cs = 130 = 0x82	Reset_Communication. Führt Reset der Kommunikationsfunktionen durch. Objekte 0x1000 - 0x1FFF werden auf Power-On Defaults zurückgesetzt

## Beispiel 1

Mit folgendem Telegramm werden netzwerkweit alle Baugruppen in den Fehlerzustand (Ausgänge sicherer Zustand) überführt:

11-bit Identifizier	2 Byte Nutzdaten						
0x00	0x02	0x00					

**Beispiel 2**

Mit folgendem Telegramm wird Knoten 17 zurückgesetzt (resetted):

11-bit Identifier	2 Byte Nutzdaten							
0x00	0x81	0x11						

**Boot-Up-Nachricht**

Boot-Up-Nachricht

Nach der Initialisierungsphase und dem Selbsttest sendet der Buskoppler die Boot-Up-Nachricht, eine CAN-Nachricht mit einem Datenbyte (0) auf dem Identifier der Guarding- bzw. Heartbeat-Nachricht: CAN-ID = 0x700 + Node-ID. Damit kann ein temporärer Ausfall einer Baugruppe während des Betriebs (z.B. durch einen Spannungseinbruch) oder eine nachträglich eingeschaltete Baugruppe zuverlässig auch ohne Node Guarding festgestellt werden. Der Sender kann über den Identifier der Nachricht (siehe Default-Identifier-Verteilung) bestimmt werden.

Außerdem ist es mit Hilfe der Boot-Up-Nachricht möglich, die beim Aufstarten am Netz befindlichen Knoten mit einem einfachen CAN-Monitor zu erkennen, ohne dass ein Schreibzugriff (z.B. Scannen des Netzwerks durch Auslesen von Parameter 0x1000) auf den Bus erforderlich ist.

Schließlich wird durch die Boot-Up-Nachricht das Ende der Initialisierungsphase kommuniziert; der Buskoppler signalisiert, dass er nun konfiguriert bzw. gestartet werden kann.



**Firmwarestand BA**

Bis Firmwarestand BA wurde für die Boot-Up-Nachricht der Emergency Identifier genutzt.

**Format Boot-Up Nachricht**

11-bit Identifier	1 Byte Nutzdaten							
0x700 (=1792) + Node-ID	0x00							

**Knotenüberwachung**

Knotenüberwachung

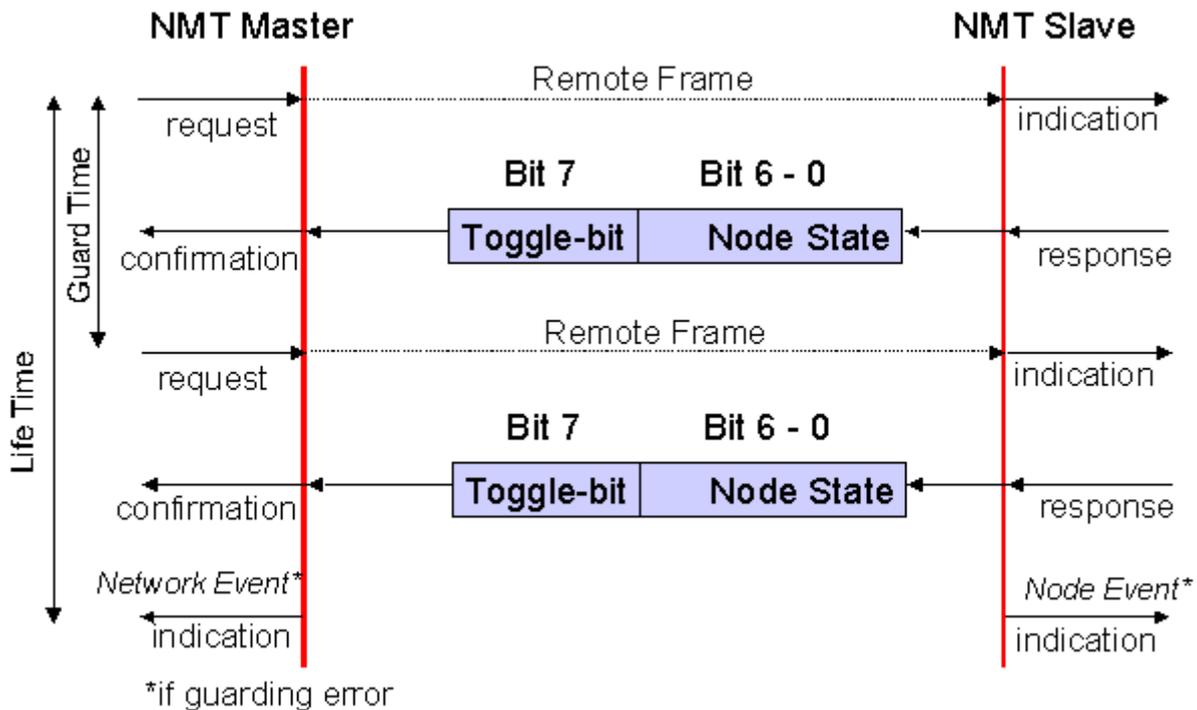
Für die Ausfallüberwachung des CANopen Netzwerkes stehen Heartbeat und Guarding-Mechanismen zur Verfügung. Diese sind bei CANopen besonders wichtig, da sich die Baugruppen in der ereignisgesteuerten Betriebsart nicht regelmäßig melden. Beim Guarding werden die Teilnehmer per Datenanforderungstelegramm (Remote Frame) zyklisch nach ihrem Status gefragt, beim Heartbeat senden die Knoten ihren Status von selbst.

**Guarding: Node Guarding und Life Guarding**

Guarding

Über Node Guarding werden die dezentralen Peripherie-Baugruppen überwacht, die ihrerseits über Life Guarding den Ausfall des Guarding-Masters erkennen können. Beim Guarding setzt der Master Remote Frames (remote transmit request, Nachrichten-Anforderungstelegramme) auf die Guarding Identifier der zu überwachenden Slaves ab. Diese antworten mit der Guarding-Nachricht. Diese enthält den Status-Code des Slaves sowie ein Toggle-Bit, das nach jeder Nachricht wechseln muss. Falls Status- oder Toggle-Bit nicht mit den vom NMT-Master erwarteten übereinstimmen oder falls keine Antwort erfolgt geht der Master von einem Slave-Fehler aus.

## Guarding-Verfahren



## Protokoll

Protokoll

Das im ersten Guarding-Telegramm übertragene Toggle-Bit (t) hat den Wert 0. Anschließend wechselt (toggelt) das Bit in jedem Guarding-Telegramm und signalisiert so, ob ein Telegramm verloren ging. In den restlichen sieben Bit gibt der Knoten seinen Netzwerk Status (s) an:

s	Status
4 = 0x04	Stopped (früher: Prepared)
5 = 0x05	Operational
127 = 0x7F	Pre-Operational

## Beispiel

Die Garding Nachricht des Knotens 27 (0x1B) muss mit einem Remote Frame mit Identifier 0x71B (1819<sub>dez</sub>) angefragt werden. Wenn der Knoten *Operational* ist, wechselt das erste Datenbyte der Antwort-Nachricht zwischen 0x05 und 0x85, im Zustand *Pre-Operational* wechselt es zwischen 0x7F und 0xFF.

## Guard Time und Life Time Factor

Wenn der Master die Guard-Nachrichten streng zyklisch anfordert, kann der Slave den Ausfall des Masters erkennen. Falls der Slave in diesem Fall innerhalb der eingestellten *Node Life Time* keine Nachrichtenforderung vom Master erhält (Guarding-Fehler), geht er von einem Masterausfall aus (Watchdog-Funktion). Dann setzt er seine Ausgänge in den Fehlerzustand, sendet ein Emergency-Telegramm und fällt in den Zustand *Pre-Operational* zurück. Nach einem Guarding Time-Out kann das Verfahren durch Übertragen eines erneuten Guarding-Telegramms wieder angeregt werden.

Die Node Life-Time berechnet sich aus den Parametern Guard-Time (Objekt 0x100C) und Life-Time-Factor (Objekt 0x100D):

Life-Time = Guard-Time x Life-Time-Factor

Falls einer der beiden Parameter "0" ist (Default-Einstellung), erfolgt keine Überwachung des Masters (kein Life Guarding).

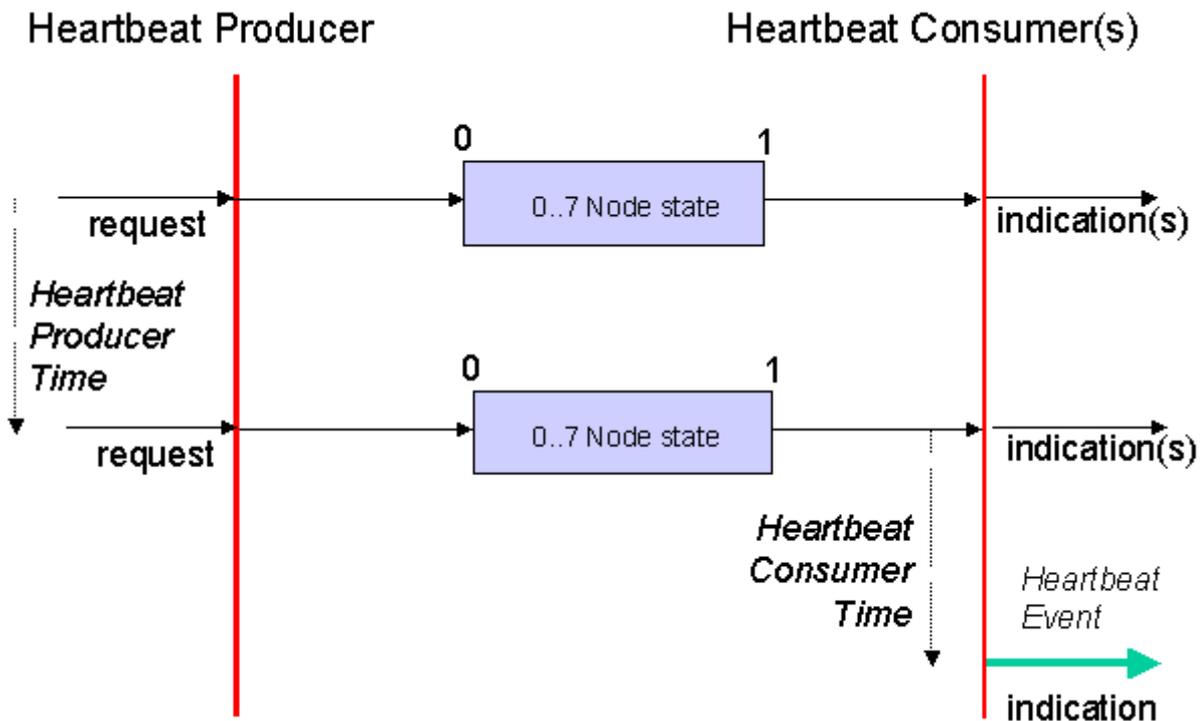
**Heartbeat: Knotenüberwachung ohne Remote Frame**

Heartbeat

Beim Heartbeat-Verfahren senden die Knoten ihre jeweilige Statusmeldung zyklisch selbsttätig. Es kann daher auf Remote Frames verzichtet werden und es wird weniger Buslast erzeugt als beim Guarding-Verfahren.

Der Master sendet sein Heartbeat-Telegramm ebenfalls zyklisch, die Slaves können somit den Ausfall des Masters ebenfalls erkennen.

**Heartbeat-Verfahren**



**Protokoll**

Beim Heartbeat-Verfahren wird auf das Toggle-Bit verzichtet, die Knoten senden zyklisch Ihren Status (s). Siehe [Guarding \[► 28\]](#).

**3.4.2 Prozessdatenobjekte (PDO)**

**Einführung**

Bei vielen Feldbus-Systemen wird ständig das gesamte Prozessabbild übertragen - meist mehr oder weniger zyklisch. CANopen ist nicht auf dieses Kommunikationsprinzip beschränkt, da CAN durch die Multi-Master Buszugriffsregelung auch andere Möglichkeiten bietet: die Prozessdaten werden bei CANopen nicht im Master/Slave-Verfahren übertragen, sondern folgen dem Produzenten/Konsumenten-Modell (Producer/Consumer). Hierbei sendet ein Busknoten seine Daten von sich aus (Producer), beispielsweise durch den Eintritt eines Ereignisses getriggert; alle anderen Knoten hören mit und entscheiden anhand des Identifiers, ob sie sich für dieses Telegramm interessieren und verarbeiten es entsprechend (Consumer).

Bei CANopen werden die Prozessdaten in Segmente zu maximal 8 Byte aufgeteilt. Diese Segmente heißen Prozessdatenobjekte (PDOs). Die PDOs entsprechen jeweils einem CAN-Telegramm und werden über dessen spezifischen CAN-Identifier zugeordnet und in ihrer Priorität bestimmt. Man unterscheidet Empfangs-PDOs (Receive-PDOs , RxPDOs) und Sende-PDOs (Transmit-PDOs , TxPDOs), wobei die Bezeichnung

jeweils aus Gerätesicht erfolgt: eine Ein-/Ausgabebaugruppe sendet ihre Eingangsdaten mit TxPDOs, und empfängt die Ausgangsdaten in den RxPDOs. **Diese Bezeichnung wird im TwinCAT-System-Manager beibehalten.**

## Kommunikationsparameter

### Kommunikationsparameter

Die PDOs können je nach Applikationsanforderung mit unterschiedlichen Kommunikationsparametern versehen werden. Wie alle CANopen-Parameter stehen auch diese im Objektverzeichnis des Gerätes, auf sie kann über die Servicedatenobjekte zugegriffen werden. Die Parameter für die Empfangs-PDOs stehen bei Index 0x1400 (RxPDO1) und folgende, bis zu 512 RxPDOs können vorhanden sein (Bereich bis Index 0x15FF). Entsprechend finden sich die Einträge für die Sende-PDOs bei Index 0x1800 (TxPDO1) bis 0x19FF (TxPDO512).

Für den Prozessdatenaustausch stehen auf den Beckhoff Buskopplern bzw. Feldbus Koppler Box Baugruppen jeweils 16 RxPDO und TxPDOs zur Verfügung (bei den Economy- und LowCost-Kopplern BK5110 und LC5100 sowie den Feldbus Boxen sind es jeweils 5 PDOs, da diese Geräte über weniger Prozessdaten verfügen). Die FC510x CANopen Master Karte unterstützt - beschränkt durch die DPRAM-Größe - je Kanal bis zu 192 Sende- und 192 Empfangs-PDOs. Die CANopen Klemme EL6751 organisiert die Prozessabbild dynamisch, d.h. die Prozessdaten werden hintereinander geschrieben, was eine höhere Datenübertragungsrate ermöglicht. Im Slave Mode können bis zu 64 TxPDOs und 64 RxPDOs verarbeitet werden.

Für jedes vorhandene Prozessdatenobjekt ist ein zugehöriges Kommunikationsparameter-Objekt vorhanden. Der TwinCAT-Systemmanager ordnet die eingestellten Parameter automatisch den jeweiligen Objektverzeichniseinträgen zu. Im Folgenden werden diese Einträge samt ihrer Bedeutung für das Kommunikationsverhalten der Prozessdaten erläutert.

## PDO-Identifizier

### PDO-Identifizier

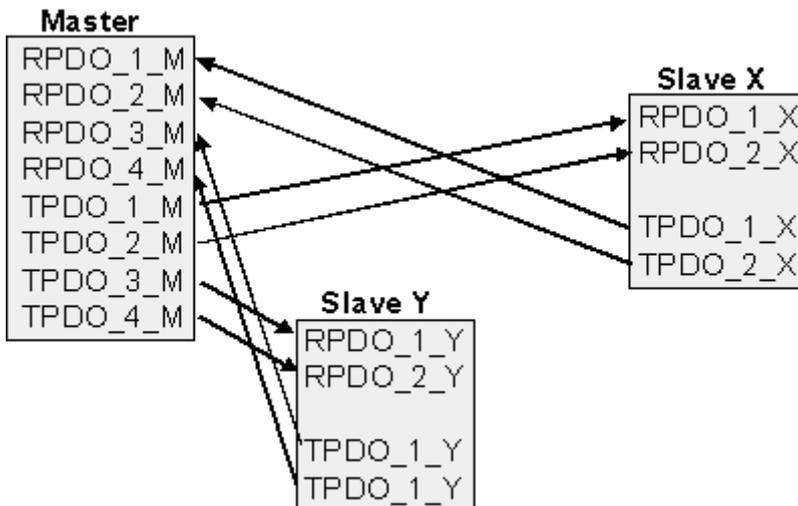
Der wichtigste Kommunikationsparameter eines PDOs ist der CAN-Identifizier (auch Communication Object Identifier, COB-ID genannt). Er dient zur Identifizierung der Daten und bestimmt deren Priorität beim Buszugriff. Für jedes CAN-Datentelegramm darf es nur einen Sendeknoten (Producer) geben; da CAN jedoch alle Nachrichten im Broadcast-Verfahren sendet kann ein Telegramm wie beschrieben von beliebig vielen Knoten empfangen werden (Consumer). Ein Knoten kann also seine Eingangsinformation mehreren Busteilnehmern gleichzeitig zur Verfügung stellen - auch ohne Weiterleitung durch einen logischen Busmaster. Der Identifizier steht in Subindex 1 des Kommunikationsparametersatzes. Er ist als 32-Bit Wert kodiert, wobei die niederwertigsten 11 Bits (Bit 0...10) den eigentlichen Identifizier enthalten. Die Datenbreite des Objektes von 32 Bit erlaubt auch den Eintrag von 29 Bit Identifiern nach CAN 2.0B, allerdings beziehen sich die Default-Identifizier [[▶ 42](#)] stets auf die üblichere 11Bit-Variante. Allgemein geht CANopen sparsam mit den zur Verfügung stehenden Identifiern um, sodass der Einsatz der 29Bit-Variante auf Sonderanwendungen beschränkt bleibt - und daher auch von den Beckhoff CANopen Geräten nicht unterstützt wird. Über das höchstwertige Bit (Bit 31) lässt sich das Prozessdatenobjekt aktivieren bzw. abschalten.

Im Anhang finden Sie eine komplette Identifizier-Liste [[▶ 138](#)].

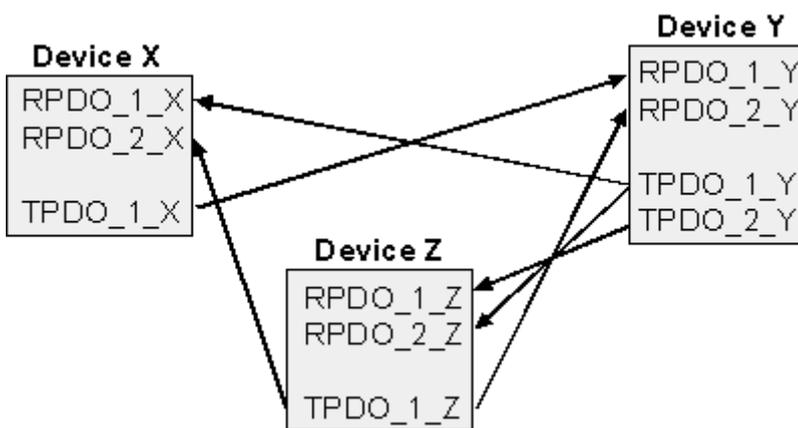
## PDO Linking

### PDO Linking

Im System der Default-Identifizier kommunizieren alle Knoten (hier: Slaves) mit einer Zentrale (Master), da kein Slave-Knoten per Default auf die Sende-Identifizier eines anderen Slave-Knotens hört).



Default Identifier-Verteilung: Master/Slave



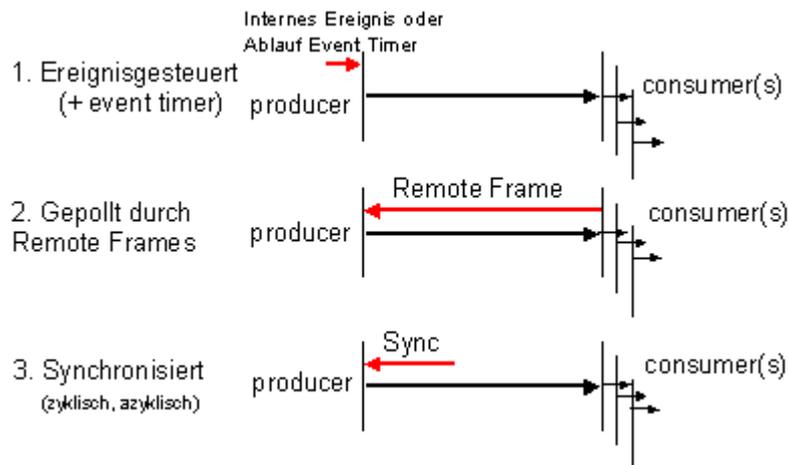
PDO Linking: Peer to Peer

Wenn das Consumer-Producer-Modell der CANopen PDOs zum direkten Datenaustausch zwischen Knoten (ohne Master) genutzt werden soll, so muss die Identifier-Verteilung entsprechend angepasst werden, damit der TxPDO-Identifier des Producers mit dem RxPDO-Identifier des Consumers übereinstimmt. Dieses Verfahren nennt man PDO Linking. Es ermöglicht beispielsweise den einfachen Aufbau von elektronischen Getrieben, bei denen mehrere Slave-Achsen gleichzeitig auf den Ist-Wert im TxPDO der Master-Achse hören.

**PDO-Kommunikationsarten: Überblick**

PDO-Kommunikationsarten: Überblick

CANopen bietet vielfältige Möglichkeiten, die Prozessdaten zu übertragen (siehe auch: [Hinweise zur PDO Parametrierung](#) [▶ 37])



## Ereignisgesteuert

### Ereignisgesteuert

Das "Ereignis" ist die Änderung eines Eingangswertes, die Daten werden sofort nach dieser Änderung verschickt. Durch die Ereignissteuerung wird die Busbandbreite optimal ausgenutzt, da nicht ständig das Prozessabbild, sondern nur die Änderung desselben übertragen wird. Gleichzeitig wird eine kurze Reaktionszeit erreicht, da bei Änderung eines Eingangswertes nicht erst auf die nächste Abfrage durch einen Master gewartet werden muss.

Ab CANopen Version 4 kann die ereignisgesteuerte Kommunikationsart mit einem zyklischen Update kombiniert werden. Auch wenn gerade kein Ereignis aufgetreten ist, werden ereignisgesteuerte TxPDO nach Ablauf des Event Timers verschickt. Beim Auftreten eines Ereignisses wird der Event Timer zurückgesetzt. Bei RxPDOs wird der Event Timer als Watchdog benutzt um das Eintreffen von ereignisgesteuerten PDOs zu überwachen. Sollte innerhalb der eingestellten Zeit kein PDO eingetroffen sein, so geht der Busknoten in den Fehlerzustand.

## Gepollt

### Gepollt

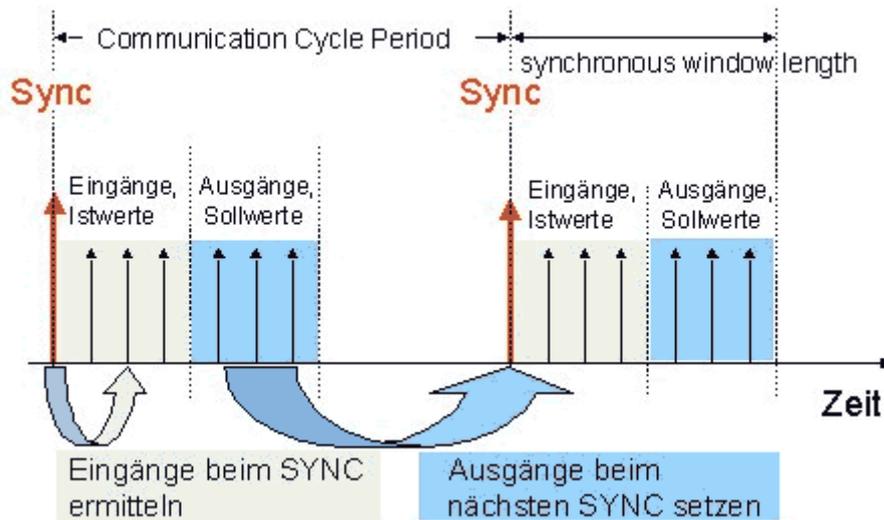
Die PDOs können auch durch Datenanforderungstelegramme (Remote Frames) gepollt werden. Auf diese Art kann etwa das Eingangsprozessabbild bei ereignisgesteuerten Eingängen auch ohne deren Änderung auf den Bus gebracht werden, beispielsweise bei einem zur Laufzeit ins Netz aufgenommenen Monitor- oder Diagnosegerät. Das zeitliche Verhalten von Remote Frame und Antworttelegramm hängt von den verwendeten CAN-Controllern ab (Bild8): Bausteine mit integrierter kompletter Nachrichtenfilterung ("FullCAN") beantworten ein Datenanforderungstelegramm in der Regel direkt und versenden sofort die im entsprechenden Sendebuffer stehenden Daten - dort muss die Applikation dafür Sorge tragen, dass die Daten ständig aktualisiert werden. CAN-Controller mit einfacher Nachrichtenfilterung (BasicCAN) reichen die Anforderung dagegen an die Applikation weiter, die nun das Telegramm mit den aktuellen Daten zusammenstellen kann. Das dauert länger, dafür sind die Daten aktuell. Beckhoff verwendet CAN Controller nach dem Basic CAN Prinzip.

Da dieses Geräteverhalten für den Anwender meist nicht transparent ist und zudem noch CAN-Controller in Verwendung sind, die Remote Frames überhaupt nicht unterstützen, kann die gepollte Kommunikationsart nur bedingt für den laufenden Betrieb empfohlen werden.

## Synchronisiert

### Synchronisiert

Nicht nur bei Antriebsanwendungen ist es sinnvoll, das Ermitteln der Eingangsinformation sowie das Setzen der Ausgänge zu synchronisieren. CANopen stellt hierzu das SYNC-Objekt zur Verfügung, ein CAN-Telegramm hoher Priorität ohne Nutzdaten, dessen Empfang von den synchronisierten Knoten als Trigger für das Lesen der Eingänge bzw. für das Setzen der Ausgänge verwendet wird.



**PDO-Übertragungsart: Parametrierung**

PDO-Übertragungsart: Parametrierung

Der Parameter PDO-Übertragungsart (Transmission Type) legt fest, wie das Versenden des PDOs ausgelöst wird bzw. wie empfangene PDOs behandelt werden:

Übertragungsart	Zyklisch	Azyklisch	Synchron	Asynchron	Nur RTR
0		X	X		
1-240	X		X		
241-251	- reserviert -				
252			X		X
253				X	X
254, 255				X	

Die Übertragungsart wird für RxPDOs in den Objekten 0x1400ff, Subindex 2, und für TxPDOs in den Objekten 0x1800ff, Subindex 2 parametrierung.

**Azyklisch Synchron**

PDOs der Übertragungsart 0 arbeiten synchron, aber nicht zyklisch. Ein RxPDO wird erst nach Empfang des nächsten SYNC-Telegramms ausgewertet. Damit lassen sich beispielsweise Achsgruppen nacheinander mit neuen Zielpositionen versehen, die alle beim nächsten SYNC gültig werden - ohne dass ständig Stützstellen ausgegeben werden müssen. Ein Gerät, dessen TxPDO auf Übertragungsart 0 konfiguriert ist, ermittelt seine Eingangsdaten beim Empfang des SYNC (synchrones Prozessabbild) und sendet sie anschließend, falls die Daten einem Ereignis entsprechen (beispielsweise eine Eingangsänderung) eingetreten ist. Die Übertragungsart 0 kombiniert also den Sendegrund "ereignisgesteuert" mit dem Sende- (und möglichst Sample-) bzw. Verarbeitungs-Zeitpunkt "SYNC-Empfang".

**Zyklisch Synchron**

Zyklisch Synchron

Bei Übertragungsart 1-240 wird das PDO zyklisch gesendet: nach jedem "n-ten" SYNC (n=1...240). Da die Übertragungsart nicht nur im Netz, sondern auch auf einem Gerät kombiniert werden dürfen, kann so z.B. ein schneller Zyklus für digitale Eingänge vereinbart werden (n=1), während die Daten der Analogeingänge in einem langsameren Zyklus übertragen werden (z.B. n=10). RxPDOs unterscheiden in der Regel nicht zwischen den Übertragungsarten 0...240: ein empfangenes PDO wird beim nächsten SYNC-Empfang gültig gesetzt. Die Zykluszeit (SYNC-Rate) kann überwacht werden (Objekt 0x1006), das Gerät reagiert bei SYNC-Ausfall dann entsprechend der Definition des Geräteprofils und schaltet z.B. seine Ausgänge in den Fehlerzustand.

Die FC510x Karte / EL6751Klemme unterstützen die synchrone Kommunikationsart vollständig: das Versenden des SYNC Telegramms ist mit der verknüpften Task gekoppelt, sodass zu jedem Taskbeginn neue Eingangsdaten zur Verfügung stehen. Das Ausbleiben eines synchronen PDOs wird erkannt und an die Applikation gemeldet.

### Nur RTR

Die Übertragungsarten 252 und 253 gelten für Prozessdatenobjekte, die ausschließlich auf Anforderung durch ein Remote Frame übertragen werden. 252 ist synchron: beim Empfang des SYNCs werden die Prozessdaten ermittelt, gesendet werden sie nur auf Anforderung. 253 ist asynchron, hier werden die Daten ständig ermittelt und auf Anforderung verschickt. Diese Übertragungsart ist generell nicht zu empfehlen, da das Abholen der Eingangsdaten von einigen CAN Controllern nur unvollständig unterstützt wird. Da die CAN Controller zudem teilweise selbsttätig auf Remote Frames antworten (ohne vorher aktuelle Eingangsdaten anzufordern), ist die Aktualität der gepollten Daten unter Umständen fragwürdig. Die Übertragungsart 252 und 253 wird aus diesen Gründen von den Beckhoff PC-Karten / Klemmen nicht unterstützt.

### Asynchron

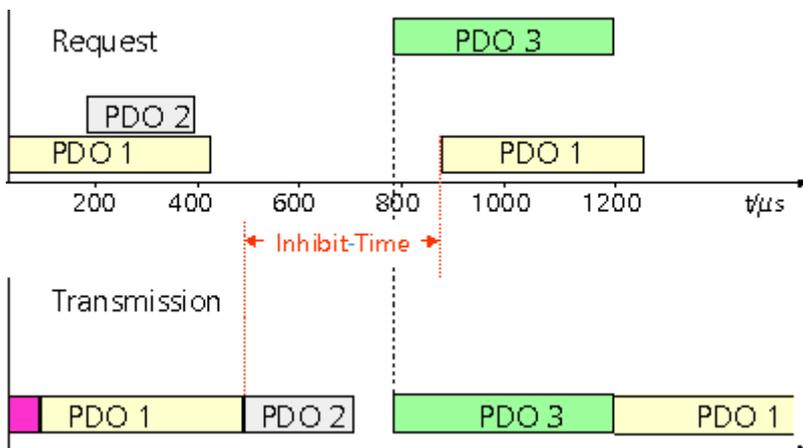
Asynchron

Die Übertragungsarten 254 + 255 sind asynchron oder auch ereignisgesteuert. Bei Übertragungsart 254 ist das Ereignis herstellerspezifisch, bei 255 im Geräteprofil definiert. Im einfachsten Fall ist das Ereignis die Veränderung eines Eingangswertes - es wird also jede Werteänderung übertragen. Die Asynchrone Übertragungsart kann mit dem Event Timer gekoppelt werden und liefert so auch dann Eingangsdaten, wenn aktuell kein Ereignis aufgetreten ist.

### Inhibit Zeit

Inhibit Zeit

Über den Parameter "Inhibit-Zeit" kann ein "Sende-Filter" aktiviert werden, der die Reaktionszeit bei der relativ ersten Eingangsänderung nicht verlängert, aber bei unmittelbar darauffolgenden Änderungen aktiv ist. Die Inhibit-Zeit (Sendeverzögerungszeit) beschreibt die Zeitspanne, die zwischen dem Versenden zweier gleicher Telegramme mindestens abgewartet werden muss. Wenn die Inhibit-Zeit genutzt wird, so kann die maximale Busbelastung und damit die Latenzzeit im "worst case"-Fall ermittelt werden.

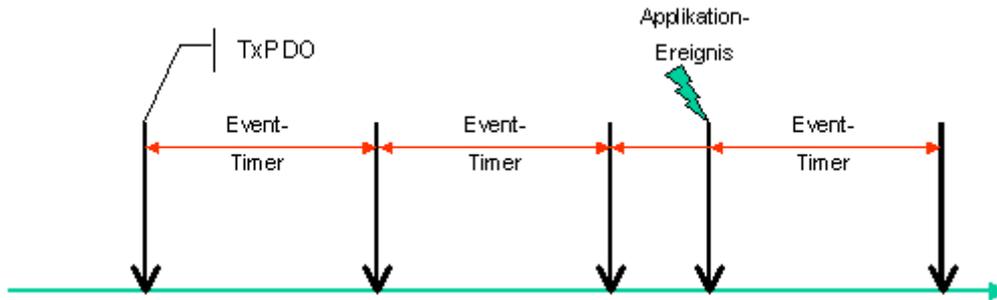


Die Beckhoff PC-Karten FC510x / EL6751 Klemme können zwar die Inhibit-Zeit auf Slave-Geräten parametrieren, unterstützen sie jedoch selbst nicht. Ein Spreizung der gesendeten PDOs (Sendeverzögerung) ergibt sich automatisch aus der gewählten Zyklus-Zeit der SPS - und es macht wenig Sinn, die SPS schneller laufen zu lassen als es die Busbandbreite zulässt. Zudem kann die Busbelastung wirkungsvoll über die synchrone Kommunikation beeinflusst werden.

### Event Timer

Event Timer

Über Subindex 5 der Kommunikationsparameter lässt sich ein Ereignis-Timer (Event Timer) für Sende-PDOs festlegen. Der Ablauf dieses Timers wird als zusätzlich eingetretenes Ereignis für das entsprechende PDO gewertet, das PDO wird also dann gesendet. Wenn das Applikationsereignis während einer Timer-Periode auftritt, so wird ebenfalls gesendet und der Timer wird zurückgesetzt .



Bei Empfangs-PDOs wird der Timer-Parameter dazu verwendet, die Überwachungszeit für dieses PDO anzugeben: Die Applikation wird benachrichtigt, wenn kein entsprechendes PDO innerhalb der eingestellten Zeit empfangen wurde. Auf diese Art kann die FC510x / EL6751 jedes einzelne PDO individuell überwachen.

[Hinweise zur PDO Parametrierung \[► 37\]](#)

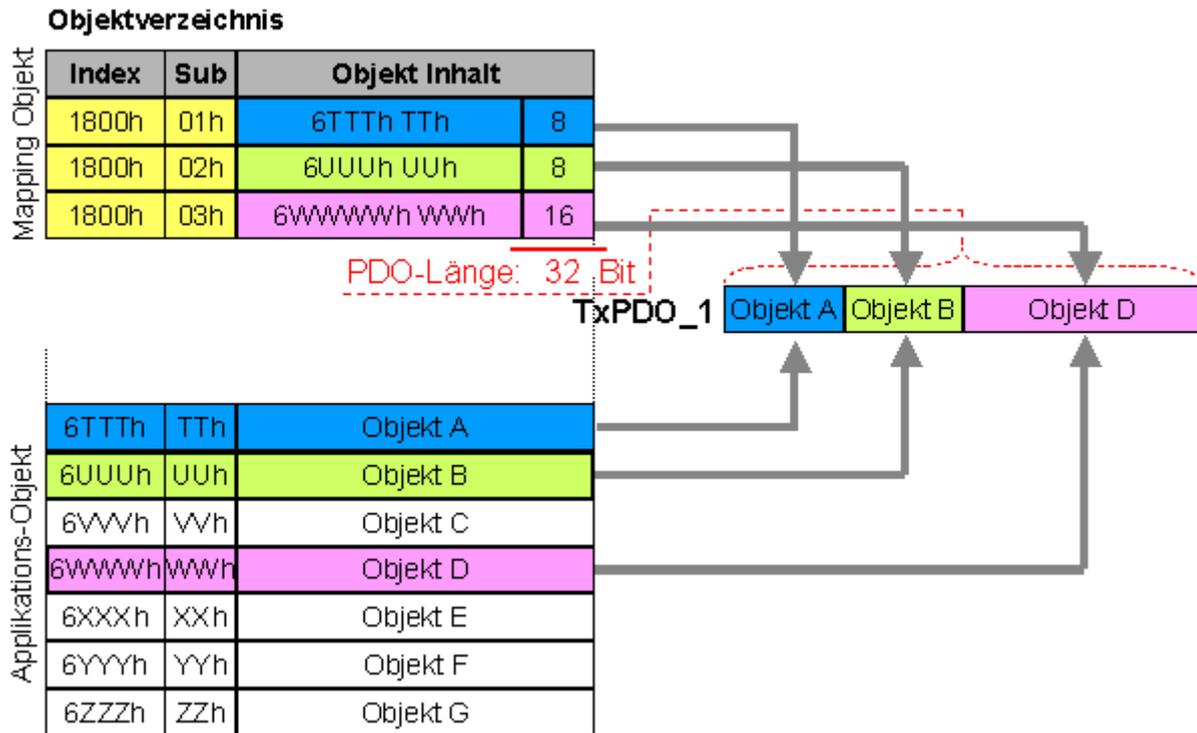
**PDO Mapping**

PDO Mapping

Unter PDO-Mapping versteht man die Abbildung der Applikationsobjekte (Echtzeitdaten) aus dem Objektverzeichnis in die Prozessdatenobjekte. Die CANopen-Geräteprofile sehen für jeden Gerätetyp ein Default Mapping vor, das für die meisten Anwendungen passend ist. So bildet das Default Mapping für digitale E/A einfach die Ein- bzw. Ausgänge ihrer physikalischen Reihenfolge gemäß in die Sende- bzw. Empfangs-Prozessdatenobjekte ab.

Die Default-PDOs für Antriebe enthalten jeweils 2 Byte Steuer- bzw. Statuswort und Soll- bzw. Istwert für die betreffende Achse.

Das aktuelle Mapping kann über entsprechende Einträge im Objektverzeichnis, die sogenannten Mapping-Tabellen, gelesen werden. An erster Stelle der Mapping Tabelle (Subindex 0) steht die Anzahl der gemappten Objekte, die im Anschluss aufgelistet sind. Die Tabellen befinden sich im Objektverzeichnis bei Index 0x1600 ff. für die RxPDOs bzw. 0x1A00ff für die TxPDOs.



### Digitale und analoge Ein-/Ausgabebaugruppen: E/A-Anzahl auslesen

Die aktuelle Anzahl der digitalen und analogen Ein-/Ausgänge lässt sich durch Auslesen der entsprechenden Applikationsobjekte im Objektverzeichnis ermitteln bzw. verifizieren:

Parameter	Adresse Objektverzeichnis
Anzahl digitale Eingangsbytes	Index 0x6000, Subindex 0
Anzahl digitale Ausgangsbytes	Index 0x6200, Subindex 0
Anzahl analoge Eingänge	Index 0x6401, Subindex 0
Anzahl analoge Ausgänge	Index 0x6411, Subindex 0

### Variables Mapping

In der Regel genügt die Default-Belegung der Prozessdatenobjekte (Default Mapping) bereits den Anforderungen. Für spezielle Anwendungsfälle kann die Belegung jedoch verändert werden: So unterstützen beispielsweise die Beckhoff CANopen Buskoppler das variable Mapping, bei dem die Applikationsobjekte (Ein- und Ausgangsdaten) frei den PDOs zugeordnet werden können. Hierzu müssen die Mapping-Tabellen konfiguriert werden: Ab CANopen Version 4 ist nur noch die folgende Vorgehensweise zulässig, die genau eingehalten werden muss:

1. Zunächst PDO löschen (0x1400ff, bzw. 0x1800ff, Subindex 1, Bit 31 auf "1" setzen)
2. Subindex 0 im Mapping Parameter (0x1600ff bzw. 0x1A00ff) auf "0" setzen
3. Mapping Einträge (0x1600ff bzw. 0x1A00ff, SI 1..8) verändern
4. Subindex 0 im Mapping Parameter auf gültigen Wert setzen. Das Gerät überprüft dann die Einträge auf Konsistenz.
5. PDO anlegen durch Eintragen d. Identifiers (0x1400ff bzw. 0x1800ff Subindex 1).

### Dummy-Mapping

Eine weiteres Feature von CANopen ist das Mappen von Platzhaltern (Dummy-Einträgen). Als Platzhalter dienen die im Objektverzeichnis hinterlegten Datentyp-Einträge, die ja selbst nicht mit Daten versehen sind. Sind solche Einträge in der Mapping-Tabelle enthalten, so werden die entsprechenden Daten vom Gerät nicht ausgewertet. Auf diese Art können beispielsweise mehrere Antriebe über ein einziges CAN-Telegramm mit neuen Sollwerten versorgt werden oder Ausgänge auf mehreren Knoten auch im ereignisgesteuerten Modus gleichzeitig gesetzt werden.

### 3.4.3 PDO-Parametrierung

Auch wenn die meisten CANopen-Netze in der Default-Einstellung und damit mit minimalem Konfigurationsaufwand zufrieden stellend arbeiten, so sollte zumindest überprüft werden, ob die vorhandene Buslast vertretbar ist. 80% Busauslastung mag für ein rein zyklisch synchron arbeitendes Netzwerk akzeptabel sein, für ein rein ereignisgesteuertes Netz ist dieser Wert in der Regel zu hoch, da kaum Bandbreite für zusätzliche Ereignisse zur Verfügung steht.

#### Applikationsanforderungen berücksichtigen

Die Prozessdatenkommunikation sollte hinsichtlich einiger sich teilweise widersprechender Applikationsanforderungen optimiert werden. Hierzu gehören

- Geringer Parametrierungsaufwand - optimal sind brauchbare Default-Werte
- Garantierte Reaktionszeit auf bestimmte Ereignisse
- Zykluszeit bei Regelvorgängen über den Bus
- Sicherheitsreserven für Busstörungen (genügend Bandbreite für Nachrichtenwiederholung)
- Maximale Baud-Rate - hängt von der maximalen Buslänge ab
- Gewünschte Kommunikationspfade - wer spricht mit wem

Der bestimmende Faktor ist meist die zur Verfügung stehende Busbandbreite (Buslast).

#### Baud-Rate

Baud-Rate

Allgemein wird man beginnen, die Baud-Rate so groß zu wählen, wie es die Buslänge erlaubt. Hierbei sollte man berücksichtigen, dass serielle Bussysteme grundsätzlich um so empfindlicher auf Störeinflüsse reagieren, je höher die Baud-Rate ist. Es gilt also die Regel: so schnell wie nötig. 1000 kBit/s sind meist nicht erforderlich und uneingeschränkt nur bei Netzwerken innerhalb eines Schaltschranks ohne galvanische Trennung der Busknoten empfehlenswert. Die Erfahrung zeigt auch, dass das Abschätzen der verlegten Buskabellänge häufig zu optimistisch erfolgt - die tatsächliche Kabellänge also größer ist.

#### Kommunikationsart bestimmen

Ist die Baud-Rate gewählt, so gilt es nun die PDO-Kommunikationsart(en) zu bestimmen. Diese haben unterschiedliche Vor- und Nachteile:

- Die zyklisch synchrone Kommunikation ergibt eine genau vorhersagbare Busbelastung und damit ein definiertes Zeitverhalten - man könnte auch sagen, der worst case ist Standard. Sie ist einfach zu konfigurieren: mit dem Parameter SYNC-Rate kann die Buslast global eingestellt werden. Die Prozessabbilder werden synchronisiert: Eingänge werden gleichzeitig gelesen, Ausgangsdaten gleichzeitig gültig gesetzt - die Qualität dieser Synchronisierung ist allerdings implementierungsabhängig. Die BECKHOFF PC-Karten FC510x / CANopen-Klemme EL6751 sind in der Lage, das CANopen Bussystems mit den Zyklen der Anwendungsprogramme (SPS bzw. NC) zu synchronisieren.

Die garantierte Reaktionszeit ist bei der zyklisch synchronen Kommunikation immer mindestens so groß wie die Zykluszeit, und die Busbandbreite wird nicht optimal genutzt, da auch alte, sich nicht ändernde Daten ständig übertragen werden. Es ist aber möglich, das Netz durch die Wahl unterschiedlicher SYNC-Vielfacher (Transmission Types 1...240) zu optimieren und sich langsam ändernde Daten seltener zu übertragen als z.B. zeitkritische Eingänge. Berücksichtigt werden sollte jedoch, dass Eingangszustände, die kürzer anstehen als die Zykluszeit, nicht unbedingt kommuniziert werden. Ist dies gefordert, so sollten die entsprechenden PDOs für asynchrone Kommunikation vorgesehen werden.

- Die ereignisgesteuerte, asynchrone Kommunikation ist optimal hinsichtlich Reaktionszeit und Verwendung der Busbandbreite - man könnte sie als "CAN pur" bezeichnen. Bei ihrer Wahl muss allerdings berücksichtigt werden, dass unter Umständen viele Ereignisse gleichzeitig auftreten und sich dann entsprechende Verzögerungszeiten einstellen können, bis ein relativ niederprioreres PDO verschickt werden kann - eine seriöse Netzwerkplanung erfordert demnach eine worst-case Betrachtung. Auch muss, z.B. durch Verwendung der Inhibit Zeit, verhindert werden, dass ein sich

ständig ändernder Eingang mit hoher PDO-Priorität den Bus blockiert (Fachbegriff: "babbling idiot"). Aus diesem Grund ist beispielsweise die Ereignissteuerung bei Analogeingängen im Geräteprofil per Default abgeschaltet und muss gezielt aktiviert werden. Über den Ablauf-Timer lassen sich Zeitfenster für die Sende-PDOs einstellen: Das Telegramm wird frühestens nach Ablauf der Inhibit-Zeit und spätestens nach Verstreichen des Ablauf-Timers erneut gesendet.

- Parametriert wird die Kommunikationsart über den Transmission Type.

Es ist auch möglich, beide PDO Kommunikationsprinzipien zu kombinieren. So kann es beispielsweise sinnvoll sein, die Soll- und Istwerte einer Achsregelung zyklisch synchron auszutauschen, während Endschalter oder die mit Grenzwerten versehene Motortemperatur mit ereignisgesteuerten PDOs überwacht werden. So kombiniert man die Vorteile beider Prinzipien: Synchronität der Achskommunikation und kurze Reaktionszeit für Endschalter. Durch die dezentrale Grenzwertüberwachung wird trotz Ereignissteuerung vermieden, dass der Temperatur-Analogwert ständig zur Buslast beiträgt.

Im genannten Beispiel kann es auch sinnvoll sein, die Identifier-Verteilung gezielt zu beeinflussen, um den Buszugriff durch die Prioritätsverteilung zu optimieren: die höchste Priorität bekommt das PDO mit den Endschalterdaten, die niedrigste das mit den Temperaturwerten.

In aller Regel ist es aber nicht erforderlich, die Identifier-Verteilung anzupassen, um die Latenzzeit beim Buszugriff zu optimieren. Dagegen müssen die Identifier verändert werden, um eine masterlose Kommunikation zu ermöglichen (PDO Linking). Im genannten Beispiel könnte je ein RxPDO der Achsen denselben Identifier wie das TxPDO des Endschalters zugewiesen bekommen und dadurch eine Veränderung des Eingangswertes verzögerungsfrei empfangen.

## Buslast bestimmen

### Buslast bestimmen

In jedem Fall ist es sinnvoll, die Buslast zu bestimmen. Doch welche Buslastwerte sind zulässig bzw. sinnvoll? Unterscheiden sollte man zunächst den kurzfristigen Burst von Telegrammen, bei dem eine Anzahl CAN-Nachrichten direkt aufeinander folgt - kurzzeitig 100% Buslast. Das ist nur dann problematisch, wenn die dadurch ausgelöste Folge von Empfangsinterrupts auf den CAN-Knoten nicht mehr abgearbeitet werden kann, es also zu einem Datenüberlauf (CAN-Queue-Overrun) kommt. Das kann bei sehr hohen Baud-Raten (> 500 kBit/s) bei Knoten mit Software-Telegrammfilterung und relativ langsamen oder stark ausgelasteten Mikro-Controllern vorkommen, wenn z.B. eine direkte Folge von Remote Frames (diese enthalten keine Datenbytes und haben daher minimale Länge) auf dem Bus ist (bei 1 Mbit/s kann so alle 40 µs ein Interrupt erzeugt werden; Beispiel: ein NMT-Master sendet alle Guarding-Anforderungen direkt hintereinander). Durch geschickte Implementierung lässt sich das vermeiden, der Anwender sollte davon ausgehen können, dass von den Geräteanbietern hierfür Sorge getragen wurde. Ein Burst-Zustand ist z.B. direkt nach dem SYNC Telegramm völlig normal: vom SYNC getriggert versuchen alle synchron arbeitenden Knoten quasi gleichzeitig Ihre Daten zu senden, es finden viele Arbitrierungsvorgänge statt, die Telegramme sortieren sich nacheinander in der Reihenfolge ihrer Priorität auf den Bus. Das ist in der Regel unkritisch, da es sich hier um Telegramme mit einigen Datenbytes handelt und die Telegrammfolge damit zwar eine schnelle, aber überschaubare Folge von Empfangsinterrupts auf den CAN-Knoten auslöst.

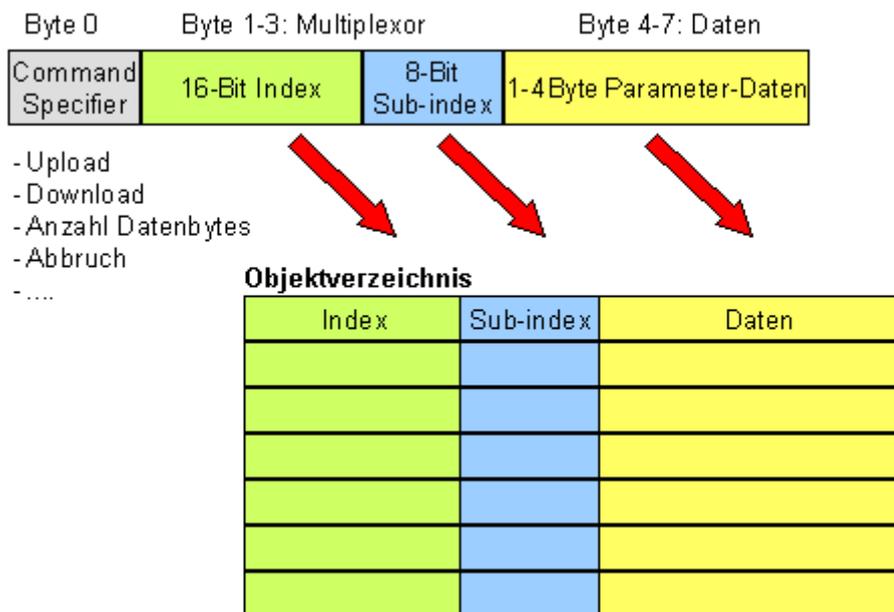
Unter Buslast versteht man meist den gemittelten Wert über mehrere Primärzyklen, also z.B. das Mittel über 100-500 ms. CAN, und damit CANopen, ist zwar in der Lage, nahe 100% Buslast auf Dauer zu bewältigen, aber dann steht keine Bandbreite für eventuelle Wiederholungen bei Störeinflüssen, asynchrone Fehlermeldungen, Parametrierung etc. zur Verfügung. Selbstverständlich hat die vorherrschende Art der Kommunikation einen großen Einfluss auf die sinnvolle Buslast: ein komplett zyklisch synchron arbeitendes Netz befindet sich ja bereits nahe am worst case Zustand und kann daher mit Werten von 70-80% betrieben werden. Für ein rein ereignisgesteuertes Netz ist diese Zahl nur schwer anzugeben: es muss hier abgeschätzt werden, wie viele zusätzliche Ereignisse im Vergleich zum derzeitigen Anlagenzustand auftreten können und für wie lange das zu einem Burst führt - also wie lange die relativ niederpriorste Nachricht dann verzögert würde. Ist dieser Wert von der Applikation her zulässig, so ist die aktuelle Buslast akzeptabel. Als Näherungswert kann meist angenommen werden, dass ein ereignisgesteuertes Netz mit 30-40% Grundlast genügend Reserven für worst-case-Szenarien hat - diese Annahme macht aber eine sorgfältige Analyse nicht überflüssig, wenn Verzögerungen zu kritischen Anlagenzuständen führen können.

Die BECKHOFF CANopen-Master-Karten FC510x / CANopen-Masterklemme EL6751 zeigen die Buslast über den System Manager ein. Diese Variable kann auch in der SPS verarbeitet oder in der Visualisierung zur Anzeige gebracht werden.

Neben den Kommunikationsparametern ist natürlich die Datenbelegung der Prozessdatenobjekte entscheidend: das [PDO Mapping](#). [▶ 35]

### 3.4.4 Servicedatenobjekte (SDO)

Die im Objektverzeichnis aufgeführten Parameter werden über Servicedatenobjekte gelesen und beschrieben. Diese SDOs sind *Multiplexed Domains*, also Datenstrukturen beliebiger Größe, die mit einem Multiplexor (Adresse) versehen sind. Der Multiplexor besteht aus 16-Bit-Index und 8-Bit-Subindex, die die entsprechenden Einträge im Objektverzeichnis adressieren.



SDO-Protokoll: Zugriff auf Objektverzeichnis

Die CANopen Buskoppler sind Server für das SDO, d.h. sie stellen auf Anforderung des Clients (z.B. des IPCs oder der SPS) Daten zur Verfügung (Upload) oder sie empfangen Daten vom Client (Download). Dabei findet ein Handshake zwischen Client und Server statt.

Wenn der zu übertragende Parameter bis zu 4 Bytes umfasst, genügt ein einziger Handshake (ein Telegrammpaar): Beim Download sendet der Client die Daten zusammen mit Index, Subindex und der Server bestätigt den Erhalt. Beim Upload fordert der Client die Daten an indem er Index und Subindex des gewünschten Parameters überträgt, und der Server sendet den Parameter (incl. Index und Subindex) in seinem Antworttelegramm.

Für Upload und Download wird das gleiche Identifier-Paar verwendet. In den stets 8 Byte großen Telegrammen sind im ersten Datenbyte die unterschiedlichen Dienste codiert. Bis auf die Objekte 1008h, 1009h und 100Ah (Gerätename, Hardware- bzw. Softwareversion) sind alle Parameter der Buskoppler nur bis zu 4 Byte groß, daher beschränkt sich diese Beschreibung auf die Übertragung dieser Daten im beschleunigten Transfer (Expedited Transfer).

#### Protokoll

Im Folgenden wird der Aufbau der SDO-Telegramme beschrieben.

**Client -> Server, Upload Request**

11-bit Identifier	8 Byte Nutzdaten							
0x600 (=1536dez) + Node-ID	0x40	Index0	Index1	SubIdx	0x00	0x00	0x00	0x00

Parameter	Erläuterung
Index0	Index Low-Byte (Unsigned16, LSB)
Index1	Index High-Byte (Unsigned16, MSB)
SubIdx	Subindex (Unsigned8)

**Client -> Server, Upload Response**

11-bit Identifier	8 Byte Nutzdaten							
0x580 (=1408dez) + Node-ID	0x4x	Index0	Index1	SubIdx	Data0	Data1	Data2	Data3

Parameter	Erläuterung
Index0	Index Low-Byte (Unsigned16, LSB)
Index1	Index High-Byte (Unsigned16, MSB)
SubIdx	Subindex (Unsigned8)
Data0	Daten Low-Low-Byte (LLSB)
Data3	Daten High-High-Byte (MMSB)

Parameter des Datentyps Unsigned8 werden im Byte D0 übertragen, Parameter des Typs Unsigned16 in D0 und D1.

Die Anzahl der gültigen Datenbytes ist im ersten CAN-Datenbyte (0x4x) wie folgt codiert:

Anzahl Parameter-Bytes	1	2	3	4
Erstes CAN-Datenbyte	0x4F	0x4B	0x47	0x43

**Client -> Server, Download Request**

11-bit Identifier	8 Byte Nutzdaten							
0x600 (=1536dez) + Node-ID	0x22	Index0	Index1	SubIdx	Data0	Data1	Data2	Data3

Parameter	Erläuterung
Index0	Index Low-Byte (Unsigned16, LSB)
Index1	Index High-Byte (Unsigned16, MSB)
SubIdx	Subindex (Unsigned8)
Data0	Daten Low-Low-Byte (LLSB)
Data3	Daten High-High-Byte (MMSB)

Optional ist es möglich, im ersten CAN-Datenbyte die Anzahl der gültigen Parameter-Datenbytes anzugeben

Anzahl Parameter-Bytes	1	2	3	4
Erstes CAN-Datenbyte	0x2F	0x2B	0x27	0x23

In der Regel ist das jedoch nicht erforderlich, da jeweils nur die niederwertigen Datenbytes bis zur Länge des zu beschreibenden Objektverzeichniseintrags ausgewertet werden. Ein Download von Daten bis zu 4 Byte Länge kann daher bei BECKHOFF Busknoten immer mit 22h im ersten CAN-Datenbyte erfolgen.

**Client -> Server, Download Response**

11-bit Identifier	8 Byte Nutzdaten							
0x580 (=1408dez) + Node-ID	0x60	Index0	Index1	SubIdx	0x00	0x00	0x00	0x00

Parameter	Erläuterung
Index0	Index Low-Byte (Unsigned16, LSB)
Index1	Index High-Byte (Unsigned16, MSB)
SubIdx	Subindex (Unsigned8)

**Abbruch Parameterkommunikation**

Im Falle einer fehlerhaften Parameterkommunikation wird diese abgebrochen. Client bzw. Server senden dazu ein SDO-Telegramm folgender Struktur:

11-bit Identifier	8 Byte Nutzdaten							
0x580 (Client) oder 0x600 (Server) + Node-ID	0x80	Index0	Index1	SubIdx	Error0	Error1	Error2	Error3

Parameter	Erläuterung
Index0	Index Low-Byte (Unsigned16, LSB)
Index1	Index High-Byte (Unsigned16, MSB)
SubIdx	Subindex (Unsigned8)
Error0	SDO Fehler-Code Low-Low-Byte (LLSB)
Error3	SDO Fehler-Code High-High-Byte (MMSB)

Liste der SDO-Fehler-Codes (Abbruch-Grund des SDO-Transfers):

SDO Fehler-Code	Erläuterung
0x05 03 00 00	Toggle Bit nicht geändert
0x05 04 00 01	SDO Command Specifier ungültig oder unbekannt
0x06 01 00 00	Zugriff auf dieses Objekt wird nicht unterstützt
0x06 01 00 02	Versuch, auf einen Read_Only Parameter zu schreiben
0x06 02 00 00	Objekt nicht im Objektverzeichnis vorhanden
0x06 04 00 41	Objekt kann nicht ins PDO gemappt werden
0x06 04 00 42	Anzahl und/oder Länge der gemappten Objekte würde PDO Länge überschreiten
0x06 04 00 43	Allgemeine Parameter Inkompatibilität
0x06 04 00 47	Allgemeiner interner Fehler im Gerät
0x06 06 00 00	Zugriff wegen Hardware-Fehler abgebrochen
0x06 07 00 10	Datentyp oder Parameterlänge stimmen nicht überein oder sind unbekannt
0x06 07 00 12	Datentyp stimmt nicht überein, Parameterlänge zu groß
0x06 07 00 13	Datentyp stimmt nicht überein, Parameterlänge zu klein
0x06 09 00 11	Subindex nicht vorhanden
0x06 09 00 30	allgemeiner Wertebereich-Fehler
0x06 09 00 31	Wertebereich-Fehler: Parameter wert zu groß
0x06 09 00 32	Wertebereich-Fehler: Parameter wert zu klein
0x06 0A 00 23	Resource nicht verfügbar
0x08 00 00 21	Zugriff wegen lokaler Applikation nicht möglich
0x08 00 00 22	Zugriff wegen aktuellem Gerätestatus nicht möglich

Für die Register-Kommunikation (Index 0x4500, 0x4501) wurden weitere, herstellerspezifische Fehler-Codes eingeführt:

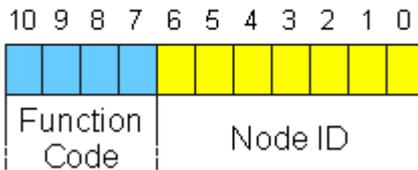
SDO Fehler-Code	Erläuterung
0x06 02 00 11	ungültige Tabelle: Tabelle oder Kanal nicht vorhanden
0x06 02 00 10	ungültiges Register: Tabelle nicht vorhanden
0x06 01 00 22	Schreibschutz noch gesetzt
0x06 07 00 43	fehlerhafte Anzahl Funktionsargumente
0x06 01 00 21	Funktion noch aktiv, später erneut versuchen
0x05 04 00 40	Allgemeiner Routing Fehler
0x06 06 00 21	Fehler Zugriff BC Tabelle
0x06 09 00 10	Allgemeiner Fehler bei Kommunikation mit Klemme
0x05 04 00 47	Time-out bei Kommunikation mit Klemme

### 3.4.5 Identifier-Verteilung

#### Default Identifier

CANopen sieht für die wichtigsten Kommunikationsobjekte Default Identifier vor, die aus der 7-Bit Knotenadresse (Node-ID) und einem 4-Bit Function-Code nach folgendem Schema abgeleitet werden:

11 Bit Identifier



Für die Broadcast-Objekte wird die Node-ID 0 eingesetzt. Damit ergeben sich folgende Default Identifier:

**Broadcast-Objekte**

Objekt	Funktion	Function Code	resultierende COB ID		Objekt für Comm. Parameter / Mapping
			hex	dez	
NMT	Boot-Up	0	0x00	0	- / -
SYNC	Synchronisation	1	0x80	128	0x1005

**Peer-to-Peer-Objekte**

Objekt	Funktion	Function Code	resultierende COB ID		Objekt für Comm. Parameter / Mapping
			hex	dez	
Emergency	Status / Fehler	1	0x81 - 0xFF	129 - 255	- / -
PDO1 (tx)	dig. Eingänge	11	0x181 - 0x1FF	385 - 511	0x1800
PDO1 (rx)	digitale Ausgänge	100	0x201 - 0x27F	513 - 639	0x1400
PDO2 (tx)	analoge Eingänge	101	0x281 - 0x2FF	641 - 767	0x1801
PDO2 (rx)	analoge Ausgänge	110	0x301 - 0x37F	769 - 895	0x1401
PDO3 (tx)	analoge Eingänge*	111	0x381 - 0x3FF	897 - 1023	0x1802
PDO3 (rx)	analoge Ausgänge*	1000	0x401 - 0x47F	1025 - 1151	0x1402
PDO4 (tx)	analoge Eingänge*	1001	0x481 - 0x4FF	1153 - 1279	0x1803
PDO4 (rx)	analoge Ausgänge*	1010	0x501 - 0x57F	1281 - 1407	0x1403
SDO (tx)	Parameter	1011	0x581 - 0x5FF	1409 - 1535	- / -
SDO (rx)	Parameter	1100	0x601 - 0x67F	1537 - 1663	- / -
Guarding	Life-/Node-guarding, Heartbeat, Boot-Up Nachricht	1110	0x701 - 0x77F	1793 - 1919	(0x100C)

\*) Für PDO3 und PDO4 gilt das Beckhoff Default Mapping [[► 162](#)]. In den meisten Konfigurationen enthalten PDO 3+4 Daten von analogen Ein/Ausgängen, es können jedoch auch "überzählige" Daten von digitalen E/As oder Daten von Sonderklemmen sein. Details finden Sie im Abschnitt [PDO Mapping](#) [[► 35](#)].

Bis zur CANopen-Spezifikation Version 3 waren jeweils 2 PDOs mit Default-Identifiern versehen. Die BECKHOFF Buskoppler bis Firmwarestand BA entsprechen diesem Stand der Spezifikation. Ab Firmwarestand C0 (CANopen Version 4) sind Default Identifier für bis zu 4 PDOs vorgesehen.

## Herstellerspezifische Default Identifier für zusätzliche PDOs

### Default Identifier für zusätzliche PDOs

Den zusätzlichen PDOs, die von den Beckhoff Buskopplern nach dem Standardschema befüllt werden, wird kein Identifier zugeordnet. Der Anwender muss in das Objektverzeichnis einen Identifier für diese PDOs eintragen. Einfacher ist es, die belegten PDOs über das Objekt 0x5500 aktivieren.

Über diesen Eintrag im Objektverzeichnis wird die Default-Identifier-Verteilung auf bis zu 11 PDOs ausgedehnt. Es ergeben sich folgende Identifier:

Objekt	Function Code	resultierende COB ID (hex)	resultierende COB ID (dez)
PDO5 (tx)	1101	0x681 - 0x6BF	1665 - 1727
PDO5 (rx)	1111	0x781 - 0x7BF	1921- 1983
PDO6 (tx)	111	0x1C1 - 0x1FF	449 - 511
PDO6 (rx)	1001	0x241 - 0x27F	577 - 639
PDO7 (tx)	1011	0x2C1 - 0x2FF	705 - 767
PDO7 (rx)	1101	0x341 - 0x37F	833 - 895
PDO8 (tx)	1111	0x3C1- 0x3FF	961 - 1023
PDO8 (rx)	10001	0x441 - 0x47F	1089 - 1151
PDO9 (tx)	10011	0x4C1 - 0x4FF	1217 - 1279
PDO9 (rx)	10101	0x541 - 0x57F	1345 - 1407
PDO10 (tx)	10111	0x5C1 - 0x5FF	1473 - 1535
PDO10 (rx)	11001	0x641 - 0x67F	1601- 1663
PDO11 (tx)	11011	0x6C1 - 0x6FF	1729 - 1791
PDO11 (rx)	11101	0x741 - 0x77F	1857 - 1919

### HINWEIS

#### Achtung

Der Index 0x5500 darf nicht genutzt werden, wenn Buskoppler mit mehr als 5 PDOs in Netzen mit Knotennummern über 64 vorhanden sind, da es sonst zu Identifier-Überschneidungen kommen kann.

## 3.5 CANopen Objektverzeichnis

### 3.5.1 Objektverzeichnis - Struktur

Im CANopen-Objektverzeichnis werden alle für den Buskoppler relevanten CANopen-Objekte eingetragen. Das Objektverzeichnis ist in drei verschiedene Bereiche aufgeteilt:

1. Kommunikationsspezifischer Profilbereich (Index 0x1000 - 0x1FFF).  
Enthält die Beschreibung aller spezifischen Parameter für die Kommunikation.
2. Herstellerspezifischer Profilbereich (Index 0x2000 - 0x5FFF).  
Enthält die Beschreibung herstellerspezifischen Einträge.
3. Standardisierter Geräteprofilbereich (0x6000 - 0x9FFF).  
Enthält die Objekte für das Geräteprofil nach DS-401.

Jeder Eintrag im Objektverzeichnis ist durch einen 16-Bit-Index gekennzeichnet. Falls ein Objekt aus mehreren Komponenten besteht (z.B. Objekttyp Array oder Record), sind die Komponenten über einen 8-Bit-Subindex gekennzeichnet. Der Objektname beschreibt die Funktion eines Objekts, das Datentyp-Attribut spezifiziert den Datentyp des Eintrags. Über das Zugriffsattribut ist spezifiziert, ob ein Eintrag nur gelesen werden kann, nur geschrieben werden oder gelesen und geschrieben werden darf.

#### Kommunikationsspezifischer Bereich

In diesem Bereich des Objektverzeichnisses stehen alle für die Kommunikation des CANopen-Buskopplers notwendigen Parameter und Objekte. Im Bereich 0x1000 - 0x1018 stehen verschiedene, allgemeine kommunikationsspezifische Parameter (z.B. der Gerätenamen).

Die Kommunikationsparameter (z.B. Identifier) der Receive-PDOs stehen im Bereich 0x1400 - 0x140F (plus Subindex). Die Mapping-Parameter der Receive-PDOs stehen im Bereich von 0x1600 - 0x160F (plus Subindex). Die Mappingparameter enthalten die Querverweise auf die Applikationsobjekte, die in die PDOs gemappt sind und die Datenbreite des entsprechenden Objektes (siehe auch Abschnitt PDO-Mapping).

Die Kommunikations- und Mapping-Parameter der Transmit-PDOs stehen in den Bereichen 0x1800 - 0x180F bzw. 0x1A00 - 0x1A0F.

#### Herstellerspezifischer Bereich

In diesem Bereich finden sich Einträge, die BECKHOFF spezifisch sind, z.B.:

- Datenobjekte für Sonderklemmen
- Objekte für die Register-Kommunikation, über die auf alle internen Register der Buskoppler und Busklemmen zugegriffen werden kann.
- Objekte für die vereinfachte Konfiguration der PDOs

#### Standardisierter Geräteprofilbereich

Im Standardisierten Geräteprofilbereich wird das CANopen-Geräteprofil DS-401 Version 1 unterstützt. Für Analogeingänge stehen dabei Funktionen zur Verfügung, um die Kommunikation in der ereignisgesteuerten Betriebsart an die Applikationsanforderungen anzupassen und die Buslast zu minimieren:

- Grenzwertüberwachung
- Deltafunktion
- Ereignissteuerung aktivieren / deaktivieren

## 3.5.2 Objektverzeichnis - Übersicht

---



### Hinweis

Die Objekte aus dem Objektverzeichnis sind per SDO-Zugriff, jedoch nicht generell über das KS2000 Konfigurationstool erreichbar. Dagegen sind alle Register, die per KS2000 konfiguriert werden können, auch per SDO-Zugriff auf das Objektverzeichnis (Objekte 0x4500 und 0x4501) erreichbar - wenn auch nicht mit dem gleichen Bedienungskomfort wie mit dem KS2000-Tool.

---

Parameter	Index	IL230x -B510	IP1xxx, IP2xxx -B510	IP3xxx -B510	IP4xxx -B510
Gerätetyp	0x1000	x	x	x	x
Fehlerregister	0x1001	x	x	x	x
<u>Fehlerspeicher</u> [▶ 50]	0x1003	x	x	x	x
Sync Identifier	0x1005	x	x	x	x
Sync Intervall	0x1006	x	x	x	x
Gerätename	0x1008	x	x	x	x
Hardware- Version [▶ 51]	0x1009	x	x	x	x
Software- Version [▶ 52]	0x100A	x	x	x	x
<u>Knotennummer</u> [▶ 52]	0x100B	x	x	x	x
Guard Time	0x100C	x	x	x	x
Life Time Factor	0x100D	x	x	x	x
Guarding Identifier	0x100E	x	x	x	x
Parameter speichern	0x1010	x	x	x	x
Default-Werte laden	0x1011	x	x	x	x
Emergency Identifier	0x1014	x	x	x	x
Consumer Heartbeat Time	0x1016	x	x	x	x
Producer Heartbeat Time	0x1017	x	x	x	x
Geräteerkennung (Identity Object)	0x1018	x	x	x	x
Server SDO Parameter	0x1200	x	x	x	x
Komm.-Parameter 1.-5. RxPDO	0x1400 - 0x1404	x	x	x	x
Komm.-Parameter 6.-16. RxPDO	0x1405 - 0x140F	x			
Mapping 1.-5. RxPDO	0x1600 - 0x1604	x	x	x	x
Mapping 6. -16. RxPDO	0x1605 - 0x160F	x			
Komm.-Parameter 1.-5. TxPDO	0x1800 - 0x1804	x	x	x	x
Komm.-Parameter 6.-16. TxPDO	0x1805 - 0x180F	x			
Mapping 1.-5. TxPDO	0x1A00 - 0x1A04	x	x	x	x
Mapping 6. -16. TxPDO	0x1A05 - 0x1A0F	x			

Parameter	Index	IL230x -B510	IP1xxx, IP2xxx -B510	IP3xxx -B510	IP4xxx -B510
3-Byte Sonderklemmen, Eingangsdaten	0x2600	x			
3-Byte Sonderklemmen, Ausgangsdaten	0x2700	x			
4-Byte Sonderklemmen, Eingangsdaten	0x2800	x			
4-Byte Sonderklemmen, Ausgangsdaten	0x2900	x			
5-Byte Sonderklemmen, Eingangsdaten	0x2A00	x			
5-Byte Sonderklemmen, Ausgangsdaten	0x2B00	x			
6-Byte Sonderklemmen, Eingangsdaten	0x2C00	x			
6-Byte Sonderklemmen, Ausgangsdaten	0x2D00	x			
8-Byte Sonderklemmen, Eingangsdaten	0x3000	x			
8-Byte Sonderklemmen, Ausgangsdaten	0x3100	x			
Register-Kommunikation Busknoten	0x4500	x	x	x	x
Register-Kommunikation Busklemme/Erweiterungsbox	0x4501	x		x	x
PDOs aktivieren	0x5500	x	x	x	x
Digitale Eingänge	0x6000	x	x	x	x
Interrupt Maske	0x6126	x	x	x	x
Digitale Ausgänge	0x6200	x	x	x	x
Analoge Eingänge	0x6401	x		x	

Parameter	Index	IL230x -B510	IP1xxx, IP2xxx -B510	IP3xxx -B510	IP4xxx -B510
Analoge Ausgänge	0x6411	x			x
Ereignissteuerung Analoge Eingänge	0x6423	x		x	
Oberer Grenzwert Analoge Eingänge	0x6424	x		x	
Unterer Grenzwert Analoge Eingänge	0x6425	x		x	
Deltafunktion Analoge Eingänge	0x6426	x		x	

### 3.5.3 Objekte und Daten

#### Gerätetyp

Gerätetyp

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x1000</b>	0	Device Type	Unsigned32	ro	N	0x00000000	Angabe des Gerätetyps

Der 32Bit-Wert ist in zwei 16Bit-Felder unterteilt:

MSB	LSB
Additional Information	Geräteprofil-Nummer
0000 0000 0000 wxyz	0x191 (401 <sub>dez</sub> )

Die *Additional Information* enthält Angaben über die Signalarten des E/A-Gerätes:

z=1 bedeutet digitale Eingänge,

y=1 bedeutet digitale Ausgänge,

x=1 bedeutet analoge Eingänge,

w=1 bedeutet analoge Ausgänge.

Ein BK5120 mit digitalen und analogen Eingängen, aber ohne Ausgänge, liefert also 0x00 05 01 91 zurück.

Sonderklemmen (z.B. serielle Schnittstellen, PWM-Ausgänge, Inkrementalencoder-Eingänge) werden nicht berücksichtigt. Ein Koppler, der z.B. nur serielle Schnittstellenklemmen KL6001 bestückt hat, liefert also 0x00 00 01 91 zurück.

Der Gerätetyp liefert nur eine grobe Klassifizierung des Gerätes. Für die detaillierte Identifizierung des Buskopplers und der angesteckten Klemmen kann das Klemmenbezeichnungs-Register des Buskopplers gelesen werden (Details siehe Register-Kommunikation Index 0x4500).

#### Fehlerregister

Fehlerregister

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x1001</b>	0	Error Register	Unsigned8	ro	N	0x00	Fehlerregister

Der 8Bit-Wert ist wie folgt kodiert:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ManSpec.	reserviert	reserviert	Comm.	reserviert	reserviert	reserviert	Generic

ManSpec. Herstellerspezifischer Fehler, wird in Objekt 1003 genauer spezifiziert.

Comm. Kommunikationsfehler (Overrun CAN)

Generic Ein nicht näher spezifizierter Fehler ist aufgetreten (Flag ist bei jeder Fehlermeldung gesetzt)

## Fehlerspeicher

Fehlerspeicher

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1003	0x00	Predefined error field (Fehlerspeicher)	Unsigned8	rw	N	0x00	Objekt 1003h enthält eine Beschreibung der im Gerät aufgetretenen Fehler - Subindex 0 die Anzahl der gespeicherten Fehlerzustände.
	1	Actual error	Unsigned32	ro	N	Keiner	Letzter aufgetretener Fehlerzustand
	...	...	...	--	...	...	...
	10	Standard error field	Unsigned32	ro	N	Keiner	Es werden maximal 10 Fehlerzustände gespeichert.

Der 32Bit-Wert im Fehlerspeicher ist in zwei 16Bit-Felder unterteilt:

MSB	LSB
Additional Code	Error Code

Der Additional Code enthält den Error Trigger (siehe [Emergency-Objekt \[▶ 126\]](#)) und damit eine detaillierte Fehlerbeschreibung.

Neue Fehler werden jeweils an Subindex 1 gespeichert, alle anderen Sub-indices werden entsprechend inkrementiert. Durch Schreiben einer 0 auf Subindex 0 wird der gesamte Fehlerspeicher gelöscht.

Wenn kein Fehler seit dem Power-On aufgetreten ist, dann besteht Objekt 0x1003 nur aus Subindex 0 mit eingetragener 0. Durch einen Reset oder Power Cycle wird der Fehlerspeicher gelöscht.

Wie bei CANopen üblich wird das LSB zuerst und das MSB zuletzt übertragen.

**Sync Identifier**

Sync Identifier

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1005	0	COB-ID Sync Message	Unsigned32	rw	N	0x80000080	Identifier der SYNC-Nachricht

Die unteren 11 Bit des 32-Bit Wertes enthalten den Identifier (0x80=128dez). Bit 30 gibt Auskunft, ob das Gerät das SYNC-Telegramm sendet (1) oder nicht (0). Die CANopen E/A Geräte empfangen das SYNC Telegramm, dementsprechend ist Bit 30=0. Bit 31 ist aus Gründen der Abwärtskompatibilität ohne Bedeutung.

**Sync Intervall**

Sync Intervall

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1006	0	Communication cycle period	Unsigned32	rw	N	0x00000000	Länge des SYNC-Intervalls in µs.

Wenn hier ein Wert ungleich Null eingetragen wird, so geht der Busknoten in den Fehlerzustand, wenn beim synchronen PDO-Betrieb innerhalb der Watchdog-Zeit kein SYNC-Telegramm empfangen wurde. Die Watchdog- Zeit entspricht hierbei dem 1,5-fachen der eingestellten communication cycle period - es kann also der vorgesehene SYNC-Abstand eingetragen werden.

Das E/A Update wird bei den Beckhoff CANopen Busknoten direkt nach Empfang des SYNC Telegramms durchgeführt, wenn folgende Voraussetzungen gegeben sind:

- Firmwarestand ab C0 (ab CANopen Version 4.01).
- alle PDOs, die über Daten verfügen, auf die synchrone Kommunikationsart eingestellt (0..240).
- Sync Intervall in Objekt 0x1006 eingetragen und (Sync Intervall x kleinste PDO Übertragungsart) kleiner als 90ms.

Die Baugruppen sind dann durchsynchronisiert.

**Gerätename**

Gerätename

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1008	0	Manufacturer Device Name	Visible String	ro	N	BK51x0, LC5100, IPxxxx-B510 od. ILxxxx-B510	Gerätename des Busknotens

Da der zurück gelieferte Wert größer als 4 Bytes ist, wird das segmentierte SDO-Protokoll zur Übertragung verwendet.

**Hardware-Version**

Hardware-Version

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1009	0	Manufacturer Hardware-Version	Visible String	ro	N	-	Hardwareversionsnummer des Busknotens

Da der zurück gelieferte Wert größer als 4 Bytes ist, wird das segmentierte SDO-Protokoll zur Übertragung verwendet.

### Software-Version

Software-Version

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x100A	0	Manufacturer Software-Version	Visible String	ro	N	-	Softwareversionsnummer des Busknotens

Da der zurück gelieferte Wert größer als 4 Bytes ist, wird das segmentierte SDO-Protokoll zur Übertragung verwendet.

### Knotennummer

Knotennummer

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x100B	0	Node-ID	Unsigned32	ro	N	keiner	eingestellte Knotennummer

Die Knotennummer wird aus Kompatibilitätsgründen unterstützt.

### Guard Time

Guard Time

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x100C	0	Guard Time [ms]	Unsigned16	rw	N	0	Abstand zwischen zwei Guard Telegrammen. wird durch NMT-Master oder Konfigurationsstool eingestellt.

### Life Time Factor

Life Time Factor

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x100D</b>	0	Life Time Factor	Unsigned8	rw	N	0	Life Time Factor x Guard Time = Life Time (Watchdog für Life Guarding)

Wenn innerhalb der Life Time kein Guarding-Telegramm empfangen wurde, geht der Knoten in den Fehlerzustand. Wenn Life Time Factor und/oder Guard Time = 0 sind, so führt der Knoten kein Lifeguarding durch, kann aber dennoch vom Master überwacht werden (Node Guarding).

**Guarding Identifier**

Guarding Identifier

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x100 E</b>	0	COB-ID guarding protocol	Unsigned32	ro	N	0x000007xy, xy = NodeID	Identifier des Guarding Protokolls

Der Guarding Identifier wird aus Kompatibilitätsgründen unterstützt. Seit CANopen Version 4 darf der Guarding Identifier nicht mehr verändert werden.

**Parameter speichern**

Parameter speichern

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x1010</b>	0	Store Parameter	Unsigned8	ro	N	1	Anzahl der Speicheroptionen
	1	store all parameters	Unsigned32	rw	N	1	Speichert alle (speicherbaren) Parameter

Durch Schreiben der Signatur *save* im ASCII-Code (hexadezimal 0x65766173) auf Subindex 1 werden die aktuellen Parameter nichtflüchtig gespeichert. (Bytefolge auf dem Bus incl. SDO Protokoll: 0x23 0x10 0x10 0x01 0x73 0x61 0x76 0x65).

Der Speichervorgang dauert ca. 3 Sec., bei Erfolg wird anschließend durch das entsprechende TxSDO (0x60 im ersten Byte) bestätigt. Da der Buskoppler während des Speichervorgangs keine CAN-Telegramme senden und empfangen kann, kann nur gespeichert werden, wenn der Knoten im Zustand Pre-Operational ist. Es wird empfohlen, vor dem Abspeichern das gesamte Netz in den Zustand Pre-Operational zu versetzen. Dadurch wird ein Puffer-Überlauf vermieden.

Gespeichert werden:

- Die aktuelle Klemmenbestückung (Anzahl jeder Klemmenkategorie)
- Alle PDO Parameter (Identifier, Transmission Type, Inhibit Zeit, Mapping).

## **i** Hinweis

Anschließend gelten die gespeicherten Identifier, nicht mehr die aus der Knotenadresse abgeleiteten Default-Identifier. Änderungen der DIP-Schalter-Stellung beeinflussen die PDOs dann nicht mehr!

- Alle SYNC Parameter
- Alle Guarding Parameter
- Grenzwerte, Deltawerte und Interrupt Enable für Analogeingänge

Die in den Klemmen über Register-Kommunikation direkt gespeicherten Parameter werden dort sofort nichtflüchtig gespeichert.

### Default-Werte laden

Default-Werte laden

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1011	0	Restore Parameter	Unsigned8	ro	N	4	Anzahl der Rücksetze-Optionen
	1	Restore all parameters	Unsigned32	rw	N	1	Setzt alle Parameter auf Default-Werte zurück
	4	Set manufacturer Defaults	Unsigned32	rw	N	1	Setzt alle Koppler-Parameter auf Hersteller-Einstellungen zurück (auch Register)

Durch Schreiben der Signatur *load* im ASCII-Code (hexadezimal 0x6461666C) auf Subindex 1 werden alle Parameter **beim nächsten Booten (Reset)** auf Default-Werte (Auslieferungszustand) zurückgesetzt.

(Bytefolge auf dem Bus incl. SDO Protokoll: 0x23 0x11 0x10 0x01 0x6C 0x6F 0x61 0x64).

Hierdurch werden die Default-Identifier für die PDOs wieder aktiv.

### Emergency Identifier

Emergency Identifier

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1014	0	COB-ID Emergency	Unsigned32	rw	N	0x0000008 0, + NodeID	Identifier des Emergency-Telegramms

Die unteren 11 Bit des 32-Bit Wertes enthalten den Identifier (0x80=128dez). Über das MSBit lässt sich einstellen ob das Gerät das Emergency-Telegramm sendet (1) oder nicht (0).

Alternativ lässt sich die Diagnose-Funktion der Busknoten auch durch das Bit *Gerätediagnose* in der K-Buskonfiguration (siehe Objekt 0x4500) abschalten.

**Consumer Heartbeat Time**

Consumer Heartbeat Time

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1016	0	Anzahl Elemente	Unsigned8	ro	N	2	Die Consumer Heartbeat Time beschreibt die erwartete Heartbeat-Zykluszeit sowie die Node-ID des überwachten Knotens
	1	Consumer Heartbeat Time	Unsigned32	rw	N	0	Watchdog Zeit in ms und Node-ID des überwachten Knotens

Der 32Bit-Wert wird wie folgt verwendet:

MSB		LSB
Bit 31...24	Bit 23...16	Bit 15...0
reserviert (0)	Node-ID (Unsigned8)	heartbeat time in ms (Unsigned16)

Aus der Node-ID ergibt sich der überwachte Identifier durch die Default-Identifier-Verteilung: Guard-ID = 0x700 + Node-ID.

Wie bei CANopen üblich wird das LSB zuerst und das MSB zuletzt übertragen.

**Producer Heartbeat Time**

Producer Heartbeat Time

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1017	0	Producer Heartbeat Time	Unsigned16	rw	N	0	Zeitspanne in ms zwischen zwei gesendeten Heartbeat-Telegrammen

**Geräteerkennung (Identity Object)**

Geräteerkennung (Identity Object)

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1018	0	Identity Object: Anzahl Elemente	Unsigned8	ro	N	4	Das Identity Objekt enthält allgemeine Angaben zu Art und Ausgabestand des Gerätes.
	1	Vendor ID	Unsigned32	ro	N	0x00000002	Herstellerkennung. Beckhoff hat die Vendor-ID 2
	2	Product Code	Unsigned32	ro	N	abhängig vom Produkt	Geräteerkennung
	3	Revision Number	Unsigned32	ro	N	-	Versionsnummer
	4	Serial Number	Unsigned32	ro	N	-	Produktionsdatum Low-Word, High-Byte: Kalenderwoche (dez), Low-Word, Low-Byte: Kalenderjahr

Produkt	Product Code
BK5120	0x11400
BK5110	0x113F6
LC5100	0x113EC
IPwxyz-B510	0x2wxyz
IL2301-B510	0x2008FD

### Server SDO Parameter

Server SDO Parameter

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1200	0	Anzahl Elemente	Unsigned8	ro	N	2	Kommunikationsparameter des Server SDOs. Subindex 0: Anzahl der folgenden Parameter
	1	COB-ID Client ->Server	Unsigned32	ro	N	0x000006xy, xy=Node-ID	COB-ID RxSDO (Client -> Server)
	2	COB-ID Server ->Client	Unsigned32	ro	N	0x00000580 + Node-ID	COB-ID TxSDO (Client -> Server)

Aus Gründen der Abwärtskompatibilität im Objektverzeichnis enthalten.

**Kommunikationsparameter1. RxPDO**

1. RxPDOKommunikationsparameter

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1400	0	Anzahl Elemente	Unsigned8	ro	N	5	Kommunikationsparameter des ersten Receive-PDOs. Subindex 0: Anzahl der folgenden Parameter
	1	COB-ID	Unsigned32	rw	N	0x000002xy, xy=Node-ID	COB-ID (Communication Object Identifier) RxPDO1
	2	Transmission Type	Unsigned8	rw	N	255	Übertragungstyp des PDOs
	3	Inhibit Time	Unsigned16	rw	N	0	Aus Gründen der Abwärtskompatibilität vorhanden, im RxPDO nicht genutzt.
	4	CMS Priority Group	Unsigned8	rw	N	-	Aus Gründen der Abwärtskompatibilität vorhanden, nicht genutzt.
	5	Event Timer	Unsigned16	rw	N	0	Event-Timer. Definiert Watchdog Zeit für Empfangsüberwachung des PDOs.

Subindex 1 (COB-ID): Die unteren 11 Bit des 32-Bit Wertes (Bits 0-10) enthalten den CAN-Identifizierer, das MSBit (Bit 31) gibt Auskunft, ob das PDO aktuell existiert (0) oder nicht (1), Bit 30 teilt mit, ob ein RTR-Zugriff auf dieses PDO zulässig ist (0) oder nicht (1). Es ist nicht erlaubt, den Identifizierer (Bit 0-10) zu ändern, während das Objekt existiert (Bit 31=0). Der Subindex 2 enthält die Übertragungstyp (siehe Einführung PDOs).

## Kommunikationsparameter2. RxPDO

### 2. RxPDO Kommunikationsparameter

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1401	0	Anzahl Elemente	Unsigned8	ro	N	5	Kommunikationsparameter des zweiten Receive-PDOs.
	1	COB-ID	Unsigned32	rw	N	0x000003xy, xy=Node-ID	COB-ID (Communication Object Identifier) RxPDO2
	2	Transmission Type	Unsigned8	rw	N	255	Übertragungstyp des PDOs
	3	Inhibit Time	Unsigned16	rw	N	0	Aus Gründen der Abwärtskompatibilität vorhanden, im RxPDO nicht genutzt.
	4	CMS Priority Group	Unsigned8	rw	N	-	Aus Gründen der Abwärtskompatibilität vorhanden, nicht genutzt.
	5	Event Timer	Unsigned16	rw	N	0	Event-Timer. Definiert Watchdog Zeit für Empfangsüberwachung des PDOs.

**Kommunikationsparameter3. RxPDO**

3. RxPDO Kommunikationsparameter

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1402	0	Anzahl Elemente	Unsigned8	ro	N	5	Kommunikationsparameter des dritten Receive-PDOs.
	1	COB-ID	Unsigned32	rw	N	0x000004xy, xy=Node-ID	COB-ID (Communication Object Identifier) RxPDO3
	2	Transmission Type	Unsigned8	rw	N	255	Übertragungstyp des PDOs
	3	Inhibit Time	Unsigned16	rw	N	0	Aus Gründen der Abwärtskompatibilität vorhanden, im RxPDO nicht genutzt.
	4	CMS Priority Group	Unsigned8	rw	N	-	Aus Gründen der Abwärtskompatibilität vorhanden, nicht genutzt.
	5	Event Timer	Unsigned16	rw	N	0	Event-Timer. Definiert Watchdog Zeit für Empfangsüberwachung des PDOs.

#### Kommunikationsparameter4. RxPDO

##### 4. RxPDO Kommunikationsparameter

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x1403</b>	0	Anzahl Elemente	Unsigned8	ro	N	5	Kommunikationsparameter des vierten Receive-PDOs.
	1	COB-ID	Unsigned32	rw	N	0x000005xy, xy=Node-ID	COB-ID (Communication Object Identifier) RxPDO4
	2	Transmission Type	Unsigned8	rw	N	255	Übertragungstyp des PDOs
	3	Inhibit Time	Unsigned16	rw	N	0	Aus Gründen der Abwärtskompatibilität vorhanden, im RxPDO nicht genutzt.
	4	CMS Priority Group	Unsigned8	rw	N	-	Aus Gründen der Abwärtskompatibilität vorhanden, nicht genutzt.
	5	Event Timer	Unsigned16	rw	N	0	Event-Timer. Definiert Watchdog Zeit für Empfangsüberwachung des PDOs.

**Kommunikationsparameter 5.-16. RxPDO**

5.-16. RxPDO Kommunikationsparameter

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x1404 - 0x140F (je nach Geräte Typ)</b>	0	Anzahl Elemente	Unsigned8	ro	N	5	Kommunikationsparameter des 5. bis 16. Receive-PDOs.
	1	COB-ID	Unsigned32	rw	N	0x8000000	COB-ID (Communication Object Identifier) RxPDO5...16
	2	Transmission Type	Unsigned8	rw	N	255	Übertragungstyp des PDOs
	3	Inhibit Time	Unsigned16	rw	N	0	Aus Gründen der Abwärtskompatibilität vorhanden, im RxPDO nicht genutzt.
	4	CMS Priority Group	Unsigned8	rw	N	-	Aus Gründen der Abwärtskompatibilität vorhanden, nicht genutzt.
	5	Event Timer	Unsigned16	rw	N	0	Event-Timer. Definiert Watchdog Zeit für Empfangsüberwachung des PDOs.

Die Anzahl der RxPDOs je Busknoten-Typ kann den technischen Daten entnommen werden.

### Mapping-Parameter1. RxPDO

#### 1. RxPDOMapping-Parameter

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1600	0	Anzahl Elemente	Unsigned8	rw	N	abhängig von Typ und Bestückung	Mapping-Parameter des ersten Receive-PDOs; Subindex 0: Anzahl der gemappten Objekte.
	1	1. gemapptes Objekt	Unsigned32	rw	N	0x62000108	1. gemapptes Applikation subjekt (2 Byte Index, 1 Byte Subindex, 1 Byte Bitbreite)
	2	2. gemapptes Objekt	Unsigned32	rw	N	0x62000208	2. gemapptes Applikation subjekt (2 Byte Index, 1 Byte Subindex, 1 Byte Bitbreite)
	...	...	...	...	...	...	...
	8	8. gemapptes Objekt	Unsigned32	rw	N	0x62000808	8. gemapptes Applikation subjekt (2 Byte Index, 1 Byte Subindex, 1 Byte Bitbreite)

Das erste Empfangs-PDO (RxPDO1) ist per Default für digitale Ausgangsdaten vorgesehen. Je nach Anzahl der bestückten Ausgänge wird automatisch die erforderliche Länge des PDOs bestimmt und die entsprechenden Objekte gemappt. Da die digitalen Ausgänge byteweise organisiert sind, kann die Länge des PDOs in Bytes direkt dem Subindex 0 entnommen werden.

**Mapping-Änderungen**

Um das Mapping zu verändern muss folgende Reihenfolge eingehalten werden (ab CANopen Version 4 vorgeschrieben):

1. PDO löschen (Bit 31 im Identifier-Eintrag (Subindex1) des Kommunikations-Parameters auf 1 setzen)
2. Mapping deaktivieren (Subindex 0 des Mapping Eintrages auf 0 setzen)
3. Mapping Einträge ändern (Subindices 1...8)
4. Mapping aktivieren (Subindex 0 des Mapping Eintrages auf die korrekte Anzahl der gemappten Objekte setzen)
5. PDO anlegen (Bit 31 im Identifier-Eintrag (Subindex 1) des Kommunikations-Parameters auf 0 setzen)

**Mapping-Parameter2. RxPDO**

## 2. RxPDOMapping-Parameter

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1601	0	Anzahl Elemente	Unsigned8	rw	N	abhängig von Typ und Bestückung	Mapping-Parameter des zweiten Receive-PDOs; Subindex 0: Anzahl der gemappten Objekte.
	1	1. gemapptes Objekt	Unsigned32	rw	N	0x64110110	1. gemapptes Applikation subjekt (2 Byte Index, 1 Byte Subindex, 1 Byte Bitbreite)
	2	2. gemapptes Objekt	Unsigned32	rw	N	0x64110210	2. gemapptes Applikation subjekt (2 Byte Index, 1 Byte Subindex, 1 Byte Bitbreite)
	...	...	...	...	...	...	...
	8	8. gemapptes Objekt	Unsigned32	rw	N	0x00000000	8. gemapptes Applikation subjekt (2 Byte Index, 1 Byte Subindex, 1 Byte Bitbreite)

Das zweite Empfangs-PDO (RxPDO2) ist per Default für analoge Ausgänge vorgesehen. Je nach Anzahl der bestückten Ausgänge wird automatisch die erforderliche Länge des PDOs bestimmt und die entsprechenden Objekte gemappt. Da die analogen Ausgänge wortweise organisiert sind, kann die Länge des PDOs in Bytes direkt dem Subindex 0 entnommen werden.

Um das Mapping zu verändern muss eine bestimmte Reihenfolge eingehalten werden (siehe Objekt Index 0x1600).

**Mapping-Parameter3.-16. RxPDO**

## 3.-16. RxPDOMapping-Parameter

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x1602-0x160F (je nach Geräte Typ)</b>	0	Anzahl Elemente	Unsigned8	rw	N	abhängig von Typ und Bestückung	Mapping-Parameter des 3.-16. Receive-PDOs; Subindex 0: Anzahl der gemappten Objekte.
	1	1. gemapptes Objekt	Unsigned32	rw	N	0x00000000 (Siehe Text)	1. gemapptes Applikation subjekt (2 Byte Index, 1 Byte Subindex, 1 Byte Bitbreite)
	2	2. gemapptes Objekt	Unsigned32	rw	N	0x00000000 (Siehe Text)	2. gemapptes Applikation subjekt (2 Byte Index, 1 Byte Subindex, 1 Byte Bitbreite)
	...	...	...	...	...	...	...
	8	8. gemapptes Objekt	Unsigned32	rw	N	0x00000000 (Siehe Text)	8. gemapptes Applikation subjekt (2 Byte Index, 1 Byte Subindex, 1 Byte Bitbreite)

Das 3. bis 16. Empfangs-PDO (RxPDO3ff) wird vom Busknoten je nach Klemmen-Bestückung (bzw. je nach Erweiterungs-Modulen) automatisch mit einem Default Mapping versehen. Die Vorgehensweise ist im Kapitel [PDO-Mapping \[▶ 162\]](#) beschrieben.

Um das Mapping zu verändern muss eine bestimmte Reihenfolge eingehalten werden (siehe Objekt Index 0x1600).

**i Hinweis**

DS401 V2 schreibt für die PDOs 3+4 als Default Mapping analoge Ein- bzw. Ausgangsdaten vor. Das entspricht dem Beckhoff Default Mapping dann, wenn weniger als 65 digitale Ein- bzw. Ausgänge vorhanden sind. Um die Abwärtskompatibilität zu gewährleisten wird das Beckhoff Default Mapping beibehalten - die Geräte entsprechen damit in ihrem Mapping-Verhalten DS401 V1, in allen anderen Belangen DS401 V2.

**Kommunikationsparameter1. TxPDO**

1. TxPDOKommunikationsparameter

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1800	0	Anzahl Elemente	Unsigned8	ro	N	5	Kommunikationsparameter des ersten SendepDOs. Subindex 0: Anzahl der folgenden Parameter
	1	COB-ID	Unsigned32	rw	N	0x00000180 + Node-ID	COB-ID (Communication Object Identifier) TxPDO1
	2	Transmission Type	Unsigned8	rw	N	255	Übertragungstyp des PDOs
	3	Inhibit Time	Unsigned16	rw	N	0	Wiederholungsverzögerung [Wert x 100 µs]
	4	CMS Priority Group	Unsigned8	rw	N	-	Aus Gründen der Abwärtskompatibilität vorhanden, nicht genutzt.
5	Event Timer	Unsigned16	rw	N	0	Event-Timer	

Subindex 1 (COB-ID): Die unteren 11 Bit des 32-Bit Wertes (Bits 0-10) enthalten den CAN-Identifizierer, das MSBit (Bit 31) gibt Auskunft, ob das PDO aktuell existiert (0) oder nicht (1), Bit 30 teilt mit, ob ein RTR-Zugriff auf dieses PDO zulässig ist (0) oder nicht (1). Es ist nicht erlaubt, den Identifizierer (Bit 0-10) zu ändern, während das Objekt existiert (Bit 31=0). Der Subindex 2 enthält die Übertragungstyp, Subindex 3 die Wiederholungsverzögerung zwischen zwei gleichen PDOs, Subindex 5 enthält den Event Timer. Subindex 4 ist aus Kompatibilitätsgründen vorhanden, wird aber nicht genutzt. (siehe auch Einführung PDOs).

## Kommunikationsparameter2. TxPDO

### 2. TxPDO Kommunikationsparameter

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1801	0	Anzahl Elemente	Unsigned8	ro	N	5	Kommunikationsparameter des zweiten Sende-PDOs. Subindex 0: Anzahl der folgenden Parameter
	1	COB-ID	Unsigned32	rw	N	0x00000280 + Node-ID	COB-ID (Communication Object Identifier) TxPDO1
	2	Transmission Type	Unsigned8	rw	N	255	Übertragungstyp des PDOs
	3	Inhibit Time	Unsigned16	rw	N	0	Wiederholungsverzögerung [Wert x 100 µs]
	4	CMS Priority Group	Unsigned8	rw	N	-	Aus Gründen der Abwärtskompatibilität vorhanden, nicht genutzt.
5	Event Timer	Unsigned16	rw	N	0	Event-Timer	

Das zweite Sende-PDO ist per Default für analoge Eingänge vorgesehen und für ereignisgesteuerte Übertragung konfiguriert (Transmission Type 255). Die Ereignissteuerung muss zunächst aktiviert werden (siehe Objekt 0x6423), ansonsten können die Eingänge nur per Remote Transmission Request (RTR) abgefragt (gepollt) werden.

**Kommunikationsparameter3. TxPDO**

3. TxPDOKommunikationsparameter

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1802	0	Anzahl Elemente	Unsigned8	ro	N	5	Kommunikationsparameter des dritten SendepDOs. Subindex 0: Anzahl der folgenden Parameter
	1	COB-ID	Unsigned32	rw	N	0x00000380 + Node-ID	COB-ID (Communication Object Identifier) TxPDO1
	2	Transmission Type	Unsigned8	rw	N	255	Übertragungstyp des PDOs
	3	Inhibit Time	Unsigned16	rw	N	0	Wiederholungsverzögerung [Wert x 100 µs]
	4	CMS Priority Group	Unsigned8	rw	N	-	Aus Gründen der Abwärtskompatibilität vorhanden, nicht genutzt.
5	Event Timer	Unsigned16	rw	N	0	Event-Timer	

Das dritte SendepDO wird in der Regel analoge Eingangsdaten enthalten (siehe Mapping [► 162]). Es ist für ereignisgesteuerte Übertragung konfiguriert (Transmission Type 255). Die Ereignissteuerung muss zunächst aktiviert werden (siehe Objekt 0x6423), ansonsten können die Eingänge nur per Remote Transmission Request (RTR) abgefragt (gepollt) werden.

#### Kommunikationsparameter4. TxPDO

##### 4. TxPDOKommunikationsparameter

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x1803</b>	0	Anzahl Elemente	Unsigned8	ro	N	5	Kommunikationsparameter des vierten SendepDOs. Subindex 0: Anzahl der folgenden Parameter
	1	COB-ID	Unsigned32	rw	N	0x00000480 + Node-ID	COB-ID (Communication Object Identifier) TxPDO1
	2	Transmission Type	Unsigned8	rw	N	255	Übertragungstyp des PDOs
	3	Inhibit Time	Unsigned16	rw	N	0	Wiederholungsverzögerung [Wert x 100 µs]
	4	CMS Priority Group	Unsigned8	rw	N	-	Aus Gründen der Abwärtskompatibilität vorhanden, nicht genutzt.
5	Event Timer	Unsigned16	rw	N	0	Event-Timer	

Das vierte SendepDO wird in der Regel analoge Eingangsdaten enthalten (siehe Mapping [▶ 1621](#)). Es ist für ereignisgesteuerte Übertragung konfiguriert (Transmission Type 255). Die Ereignissteuerung muss zunächst aktiviert werden (siehe Objekt 0x6423), ansonsten können die Eingänge nur per Remote Transmission Request (RTR) abgefragt (gepollt) werden.

**Kommunikationsparameter 5.-16. TxPDO**

5.-16. TxPDO Kommunikationsparameter

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x1804-0x180F (je nach Gerätetyp)</b>	0	Anzahl Elemente	Unsigned8	ro	N	5	Kommunikationsparameter des 5.-16. SendepDOs. Subindex 0: Anzahl der folgenden Parameter
	1	COB-ID	Unsigned32	rw	N	0x0000000	COB-ID (Communication Object Identifier) TxPDO1
	2	Transmission Type	Unsigned8	rw	N	255	Übertragungstyp des PDOs
	3	Inhibit Time	Unsigned16	rw	N	0	Wiederholungsverzögerung [Wert x 100 µs]
	4	CMS Priority Group	Unsigned8	rw	N	-	Aus Gründen der Abwärtskompatibilität vorhanden, nicht genutzt.
5	Event Timer	Unsigned16	rw	N	0	Event-Timer	

### Mapping 1. TxPDO

Mapping 1. TxPDO

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x1A00</b>	0	Anzahl Elemente	Unsigned8	rw	N	abhängig von Typ und Bestückung	Mapping-Parameter des ersten Transmit PDOs; Subindex 0: Anzahl der gemappten Objekte.
	1	1. gemapptes Objekt	Unsigned32	rw	N	0x60000108	1. gemapptes Applikation subjekt (2 Byte Index, 1 Byte Subindex, 1 Byte Bitbreite)
	2	2. gemapptes Objekt	Unsigned32	rw	N	0x60000208	2. gemapptes Applikation subjekt (2 Byte Index, 1 Byte Subindex, 1 Byte Bitbreite)
	...	...	...	...	...	...	...
	8	8. gemapptes Objekt	Unsigned32	rw	N	0x60000808	8. gemapptes Applikation subjekt (2 Byte Index, 1 Byte Subindex, 1 Byte Bitbreite)

Das erste Sende-PDO (TxPDO1) ist per Default für digitale Eingangsdaten vorgesehen. Je nach Anzahl der bestückten Eingänge wird automatisch die erforderliche Länge des PDOs bestimmt und die entsprechenden Objekte gemappt. Da die digitalen Eingänge byteweise organisiert sind, kann die Länge des PDOs in Bytes direkt dem Subindex 0 entnommen werden.

Um das Mapping zu verändern muss eine bestimmte Reihenfolge eingehalten werden (siehe Objekt Index 0x1600).

**Mapping 2. TxPDO**

Mapping 2. TxPDO

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1A01	0	Anzahl Elemente	Unsigned8	rw	N	abhängig von Typ und Bestückung	Mapping-Parameter des zweiten Transmit PDOs; Subindex 0: Anzahl der gemappten Objekte.
	1	1. gemapptes Objekt	Unsigned32	rw	N	0x64010110	1. gemapptes Applikation subjekt (2 Byte Index, 1 Byte Subindex, 1 Byte Bitbreite)
	2	2. gemapptes Objekt	Unsigned32	rw	N	0x64010210	2. gemapptes Applikation subjekt (2 Byte Index, 1 Byte Subindex, 1 Byte Bitbreite)
	...	...	...	...	...	...	...
	8	8. gemapptes Objekt	Unsigned32	rw	N		8. gemapptes Applikation subjekt (2 Byte Index, 1 Byte Subindex, 1 Byte Bitbreite)

Das zweite Sende-PDO (TxPDO2) ist per Default für analoge Eingangsdaten vorgesehen. Je nach Anzahl der bestückten Eingänge wird automatisch die erforderliche Länge des PDOs bestimmt und die entsprechenden Objekte gemappt. Da die analogen Eingänge wortweise organisiert sind, kann die Länge des PDOs in Bytes direkt dem Subindex 0 entnommen werden.

Um das Mapping zu verändern muss eine bestimmte Reihenfolge eingehalten werden (siehe Objekt Index 0x1600).

### Mapping 3.-16. TxPDO

Mapping 3.-16. TxPDO

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x1A02-0x1A0F (je nach Geräte Typ)	0	Anzahl Elemente	Unsigned8	rw	N	abhängig von Typ und Bestückung	Mapping-Parameter des 3.-16. Transmit PDOs; Subindex 0: Anzahl der gemappten Objekte.
	1	1. gemapptes Objekt	Unsigned32	rw	N	0x00000000 (Siehe Text)	1. gemapptes Applikation subjekt (2 Byte Index, 1 Byte Subindex, 1 Byte Bitbreite)
	2	2. gemapptes Objekt	Unsigned32	rw	N	0x00000000 (Siehe Text)	2. gemapptes Applikation subjekt (2 Byte Index, 1 Byte Subindex, 1 Byte Bitbreite)
	...	...	...	...	...	...	...
	8	8. gemapptes Objekt	Unsigned32	rw	N	0x00000000 (Siehe Text)	8. gemapptes Applikation subjekt (2 Byte Index, 1 Byte Subindex, 1 Byte Bitbreite)

Das 3. bis 16. Sende-PDO (TxPDO3ff) wird vom Busknoten je nach Klemmen-Bestückung (bzw. je nach Erweiterungs-Modulen) automatisch mit einem Default Mapping versehen. Die Vorgehensweise ist im Kapitel [PDO-Mapping](#) [► 162] beschrieben.

Um das Mapping zu verändern muss eine bestimmte Reihenfolge eingehalten werden (siehe Objekt Index 0x1600).

**i Hinweis**

DS401 V2 schreibt für die PDOs 3+4 als Default Mapping analoge Ein- bzw. Ausgangsdaten vor. Das entspricht dem Beckhoff Default Mapping dann, wenn weniger als 65 digitale Ein- bzw. Ausgänge vorhanden sind. Um die Abwärtskompatibilität zu gewährleisten wird das Beckhoff Default Mapping beibehalten - die Geräte entsprechen damit in ihrem Mapping-Verhalten DS401 V1, in allen anderen Belangen DS401 V2.

Im Objektverzeichnis (und damit auch im eds File) sind der Vollständigkeit halber zusätzlich folgende Objekteinträge vorhanden:

Index	Bedeutung
0x2000	Digitale Eingänge (Funktion identisch mit Objekt 0x6000)
0x2100	Digitale Ausgänge (Funktion identisch mit Objekt 0x6200)
0x2200	1-Byte Sonderklemmen, Eingänge (derzeit keine entsprechenden Klemmen im Produktprogramm vorhanden)
0x2300	1-Byte Sonderklemmen, Ausgänge (derzeit keine entsprechenden Klemmen im Produktprogramm vorhanden)
0x2400	2-Byte Sonderklemmen, Eingänge (derzeit keine entsprechenden Klemmen im Produktprogramm vorhanden)
0x2500	2-Byte Sonderklemmen, Ausgänge (derzeit keine entsprechenden Klemmen im Produktprogramm vorhanden)
0x2E00	7-Byte Sonderklemmen, Eingänge (derzeit keine entsprechenden Klemmen im Produktprogramm vorhanden)
0x2F00	7-Byte Sonderklemmen, Ausgänge (derzeit keine entsprechenden Klemmen im Produktprogramm vorhanden)

### 3-Byte Sonderklemmen, Eingangsdaten

3-Byte Sonderklemmen, Eingangsdaten

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x2600	0	Anzahl Elemente	Unsigned8	ro	N	abhängig von Typ und Bestückung	Anzahl verfügbarer 3-Byte Sonderkanäle, Eingänge
	1	1st input block	Unsigned24	ro	Y	0x000000	1. Eingangskanal
	...	...	...	...	...	...	...
	0x80	128. input block	Unsigned24	ro	Y	0x000000	128. Eingangskanal

Beispiel für Sonderklemmen mit 3-Byte Eingangsdaten (in Default-Einstellung): KL2502 (PWM Ausgänge, 2 x 3 Bytes)

### 3-Byte Sonderklemmen, Ausgangsdaten

3-Byte Sonderklemmen, Ausgangsdaten

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x2700</b>	0	Anzahl Elemente	Unsigned8	ro	N	abhängig von Typ und Bestückung	Anzahl verfügbarer 3-Byte Sonderkanäle, Ausgänge
	1	1st output block	Unsigned24	rww	Y	0x000000	1. Ausgangskanal
	...	...	...	...	...	...	...
	0X80	128. output block	Unsigned24	rww	Y	0x000000	128. Ausgangskanal

Beispiel für Sonderklemmen mit 3-Byte Ausgangsdaten (in der Default-Einstellung): KL2502 (PWM Ausgänge, 2 x 3 Bytes)

**4-Byte Sonderklemmen, Eingangsdaten**

4-Byte Sonderklemmen, Eingangsdaten

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x2800</b>	0	Anzahl Elemente	Unsigned8	ro	N	abhängig von Typ und Bestückung	Anzahl verfügbarer 4-Byte Sonderkanäle, Eingänge
	1	1st input block	Unsigned32	ro	Y	0x00000000	1. Eingangskanal
	...	...	...	...	...	...	...
	0X80	128. input block	Unsigned32	ro	Y	0x00000000	128. Eingangskanal

Beispiele für Sonderklemmen mit 4-Byte Eingangsdaten (in der Default-Einstellung): KL5001, KL6001, KL6021, KL6051

**4-Byte Sonderklemmen, Ausgangsdaten**

4-Byte Sonderklemmen, Ausgangsdaten

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x2900</b>	0	Anzahl Elemente	Unsigned8	ro	N	abhängig von Typ und Bestückung	Anzahl verfügbarer 4-Byte Sonderkanäle, Ausgänge
	1	1st output block	Unsigned32	rww	Y	0x00000000	1. Ausgangskanal
	...	...	...	...	...	...	...
	0X80	128. output block	Unsigned32	rww	Y	0x00000000	128. Ausgangskanal

Beispiele für Sonderklemmen mit 4-Byte Ausgangsdaten (in der Default-Einstellung): KL5001, KL6001, KL6021, KL6051

### 5-Byte Sonderklemmen, Eingangsdaten

5-Byte Sonderklemmen, Eingangsdaten

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x2A00</b>	0	Anzahl Elemente	Unsigned8	ro	N	abhängig von Typ und Bestückung	Anzahl verfügbarer 5-Byte Sonderkanäle, Eingänge
	1	1st input block	Unsigned40	ro	Y	0x00000000	1. Eingangskanal
	...	...	...	...	...	...	...
	0X40	64. input block	Unsigned40	ro	Y	0x00000000	64. Eingangskanal

Beispiel für Sonderklemmen mit 5-Byte Eingangsdaten (in der Default-Einstellung): KL1501

### 5-Byte Sonderklemmen, Ausgangsdaten

5-Byte Sonderklemmen, Ausgangsdaten

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x2B00</b>	0	Anzahl Elemente	Unsigned8	ro	N	abhängig von Typ und Bestückung	Anzahl verfügbarer 5-Byte Sonderkanäle, Ausgänge
	1	1st output block	Unsigned40	rww	Y	0x00000000	1. Ausgangskanal
	...	...	...	...	...	...	...
	0X40	64. output block	Unsigned40	rww	Y	0x00000000	64. Ausgangskanal

Beispiel für Sonderklemmen mit 5-Byte Ausgangsdaten (in der Default-Einstellung): KL1501

**6-Byte Sonderklemmen, Eingangsdaten**

6-Byte Sonderklemmen, Eingangsdaten

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x2C00</b>	0	Anzahl Elemente	Unsigned8	ro	N	abhängig von Typ und Bestückung	Anzahl verfügbarer 6-Byte Sonderkanäle, Eingänge
	1	1st input block	Unsigned48	ro	Y	0x00000000	1. Eingangskanal
	...	...	...	...	...	...	...
	0X40	64. input block	Unsigned48	ro	Y	0x00000000	64. Eingangskanal

Beispiel für Sonderklemmen mit 6-Byte Eingangsdaten (in der Default-Einstellung): KL5051, KL5101, KL5111

**6-Byte Sonderklemmen, Ausgangsdaten**

6-Byte Sonderklemmen, Ausgangsdaten

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x2D00</b>	0	Anzahl Elemente	Unsigned8	ro	N	abhängig von Typ und Bestückung	Anzahl verfügbarer 6-Byte Sonderkanäle, Ausgänge
	1	1st output block	Unsigned48	rww	Y	0x00000000	1. Ausgangskanal
	...	...	...	...	...	...	...
	0x40	64. output block	Unsigned48	rww	Y	0x00000000	64. Ausgangskanal

Beispiel für Sonderklemmen mit 6-Byte Ausgangsdaten (in der Default-Einstellung): KL5051, KL5101, KL5111

### 8-Byte Sonderklemmen, Eingangsdaten

8-Byte Sonderklemmen, Eingangsdaten

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x3000</b>	0	Anzahl Elemente	Unsigned8	ro	N	abhängig von Typ und Bestückung	Anzahl verfügbarer 6-Byte Sonderkanäle, Eingänge
	1	1st input block	Unsigned64	ro	Y	0x00000000	1. Eingangskanal
	...	...	...	...	...	...	...
	0x40	64. input block	Unsigned64	ro	Y	0x00000000	64. Eingangskanal

Beispiel für Sonderklemmen mit 8-Byte Eingangsdaten: KL5101 (mit Word-Alignment, nicht in der Default-Einstellung)

### 8-Byte Sonderklemmen, Ausgangsdaten

8-Byte Sonderklemmen, Ausgangsdaten

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x3100	0	Anzahl Elemente	Unsigned8	ro	N	abhängig von Typ und Bestückung	Anzahl verfügbarer 6-Byte Sonderkanäle, Ausgänge
	1	1st output block	Unsigned64	rww	Y	0x00000000	1. Ausgangskanal
	...	...	...	...	...	...	...
	0x40	64. output block	Unsigned64	rww	Y	0x00000000	64. Ausgangskanal

Beispiel für Sonderklemmen mit 8-Byte Ausgangsdaten: KL5101 (mit Word-Alignment, nicht in der Default-Einstellung)

**Register-Kommunikation Busknoten**

Register-Kommunikation Busknoten

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x4500	0	Register Access	Unsigned32	rw	N	keiner	Zugriff interne Register Busknoten

Der 32Bit-Wert ist wie folgt aufgebaut:

MSB		LSB	
Zugriff (Bit7) + Tabellenummer (Bit 6..0)	Registernummer	High-Byte Registerwert	Low-Byte Registerwert
[0..1] + [0...0x7F]	[0...0xFF]	[0...0xFF]	[0...0xFF]

Wie bei CANopen üblich wird das LSB zuerst und das MSB zuletzt übertragen.

Durch Zugriff auf Index 0x4500 können beliebige Register der Busstation beschrieben oder gelesen werden. Die Kanalnummer und Register werden hierbei im 32Bit-Datenwert adressiert.

**Registerwert lesen**

Zunächst muss dem Koppler mitgeteilt werden, welches Register gelesen werden soll. Hierzu muss ein SDO-Schreibzugriff auf die entsprechende Index/Subindex-Kombination erfolgen mit:

- Tabellenummer (Zugriffs-Bit=0) in Byte 3
- Registeradresse in Byte 2 des 32 Bit Datenwertes.

Bytes 1 und 0 werden nicht ausgewertet, wenn das Zugriffs-Bit (MSB in Byte 3) = 0 ist. Anschließend kann der Registerwert auf derselben Index/Subindex-Kombination gelesen werden.

Der Koppler setzt das Zugriffs-Bit nach dem Schreiben der auszulesenden Registeradresse so lange auf 1, bis der korrekte Wert zur Verfügung steht. Beim SDO-Lesezugriff ist also zu überprüfen, dass die Tabellenummer im Wertebereich 0...0x7F liegt.

Ein Zugriffsfehler bei der Register-Kommunikation wird durch entsprechende Rückgabewerte des SDO-Protokolls angezeigt (siehe Kapitel SDO, Abbruch Parameterkommunikation).

### Beispiel Registerwert lesen

Es soll festgestellt werden, welcher Baud-Ratenindex der Schalterstellung 1,1 (DIP 7,8) zugeordnet ist (siehe Kapitel *Netzwerkadresse und Baud-Raten*). Hierzu muss der Wert in Tabelle 100, Register 3 gelesen werden. Es müssen also folgende SDO Telegramme gesendet werden:

Schreibzugriff (Download Request) auf Index 4500, Subindex 0 mit 32 Bit Datenwert 0x64 03 00 00.

Id=0x600+Node-ID DLC=8; Data=23 00 45 00 00 00 03 64

Anschließend Lesezugriff (Upload Request) auf den gleichen Index/Subindex, hierbei ist der Datenwert beliebig (hier 00).

Id=0x600+Node-ID DLC=8; Data=40 00 45 00 00 00 00 00

Der Koppler antwortet mit dem Upload Response Telegramm:

Id=0x580+Node-ID DLC=8; Data=43 00 45 00 04 00 03 64

Es steht hier also der Wert 4 in diesem Register, dieser Baud-Ratenindex entspricht 125 kBit/s (Default-Wert).

### Registerwert Schreiben

SDO-Schreibzugriff auf die entsprechende Index/Subindex-Kombination mit:

- Tabellenummer + 0x80 (Zugriffs-Bit=1) in Byte 3
- Registeradresse in Byte 2
- High-Byte Registerwert in Byte 1
- Low-Byte Registerwert in Byte 0 des 32 Bit Datenwertes

### Koppler-Schreibschutz aufheben

Bevor die Register des Buskopplers beschrieben werden können muss zunächst der Schreibschutz aufgehoben werden. Hierzu müssen die folgenden Werte in der angegebenen Reihenfolge auf die entsprechenden Register geschrieben werden:

Arbeitsschritt	Tabelle	Register	Wert	entsprechender SDO Download-Wert (0x4500/0)
1.	99	2	45054 (0xAFFE)	0xE3 02 AF FE (0xE3=0x63(=99)+0x80)
2.	99	1	1 (0x0001)	0xE3 01 00 01
3.	99	0	257 (0x0101)	0xE3 00 01 01

### Koppler-Schreibschutz aufheben (CAN Darstellung)

Um den Koppler-Schreibschutz aufzuheben müssen also folgende SDO-Telegramme (Download Requests) an den Koppler geschickt werden:

Id=0x600+Node-ID DLC=8; Data=23 00 45 00 FE AF 02 E3

Id=0x600+Node-ID DLC=8; Data=23 00 45 00 01 00 01 E3

Id=0x600+Node-ID DLC=8; Data=23 00 45 00 01 01 00 E3

### Beispiel Registerwert Schreiben

Nachdem der Schreibschutz aufgehoben wurde, soll nun der Baud-Ratenindex für die DIP-Schalterstellung 1,1 auf den Wert 7 gesetzt werden. Damit wird dieser Schalterstellung die Baud-Rate 20 kBaud zugeordnet.

Hierzu muss Tabelle 100, Register 3 mit dem Wert 7 beschrieben werden, das erfolgt durch SDO-Schreibzugriff (Download Request) auf Index 0x4500, Subindex 0 mit dem 32 Bit-Wert E4 03 00 07 (0xE4 = 0x64+0x80):

Id=0x600+Node-ID DLC=8; Data=23 00 45 00 07 00 03 E4

**Klemmen identifizieren**

Über die Tabelle 9 des Buskopplers kann die Kennung des Kopplers (bzw. der Busstation) und der angesteckten Busklemmen gelesen werden. Dabei enthält Register 0 die Kennung des Buskopplers selbst, Register 1 die Kennung der ersten Klemme und Register n die Kennung der n-ten Klemme:

Tabellennummer	Registernummer	Beschreibung	Wertebereich
9	0	Busstation-Kennung	0 - 65535
9	1-255	Kennung Erweiterungsmodul/ Busklemme	0 - 65535

Die Buskopplerbeschreibung in Registernummer 0 enthält 5120 = 0x1400 beim BK5120, 5110 = 0x13F6 beim BK5110 und 5100 = 0x13EC beim LC5100. Bei den Feldbus Box Baugruppen steht in Register 0 die Kennung 510dez =0x1FE bzw. 518dez = 0x206.

Die Kennung der Erweiterungsmodule bzw. Klemmenbeschreibung enthält bei analogen und Sonderklemmen die Klemmenbezeichnung (Dez);  
 Beispiel: ist als dritte Klemme eine KL3042 gesteckt, so enthält Register 3 den Wert 3042<sub>dez</sub> (0x0BE2).

Bei digitalen Klemmen wird folgende Bit-Kennung verwendet:

MSB								LSB							
1	s6	s5	s4	s3	s2	s1	s0	0	0	0	0	0	0	a	e

s6...s1: Datenbreite in Bit; a=1: Ausgangsklemme; e=1: Eingangsklemme

Diese Kennung führt zu den unten aufgeführten Klemmenbeschreibungen bei den Klemmen:

Kennung Klemmen	Bedeutung
0x8201	2 Bit digitale Eingangsklemme,z.B. KL1002, KL1052, KI9110, KL9260
0x8202	2 Bit digitale Ausgangsklemme, z.B. KL2034, KL2612, KL2702
0x8401	4 Bit digitale Eingangsklemme, z.B. KL1104, KL1124, KL1194
0x8402	4 Bit digitale Ausgangsklemme,z.B. KL2124, KL2134, KL2184
0x8403	4 Bit digitale Ein/Ausgangsklemme,z.B. KL2212

und folgende Kennung bei den Erweiterungs Box Module:

Kennung Erweiterungs Box Module	Bedeutung
0x000A	4 Bit Eingangs- und 4 Bit Ausgangsmodul
0x0011	8 Bit Eingangs- und 8 Bit Ausgangsmodul
0x0014	8 Bit digitales Eingangsmodul
0x0015	8 Bit digitales Ausgangsmodul

**Allgemeine Koppler-Konfiguration (Tabelle 0)**

Die Tabelle 0 des Buskopplers enthält die Daten für die allgemeine Kopplerkonfiguration. In der Regel muss diese nicht verändert werden; für besondere Anwendungsfälle können die Einstellungen jedoch über die KS2000 Konfigurations-Software oder den direkten Zugriff über die Register-Kommunikation verändert werden. Hierzu muss zunächst der Schreibschutz aufgehoben werden (siehe oben).

Im Folgenden werden die relevanten Registerinträge beschrieben:

## K-Buskonfiguration

Tabelle 0, Register 2 enthält die K-Buskonfiguration und ist wie folgt codiert (Default-Wert: 0x0006):

MSB								LSB							
0	0	0	0	0	0	0	0	0	0	0	0	0	D	G	A

### A: Autoreset

Bei K-Bus-Fehler wird zyklisch versucht, den K-Bus durch Reset wieder zu aufzustarten. Wenn Emergencies und Guarding nicht ausgewertet werden, so kann es bei aktiviertem Autoreset vorkommen, dass Aus- und Eingangsinformation unerkant verloren geht.

0: kein Autoreset (Default)

1: Autoreset aktiv

### G: Gerätediagnose

Meldung (über Emergency), z.B. dass  
 - Drahtbruch bei Stromeingängen (mit Diagnose)  
 - 10 V überschritten bei 1-10V Eingangsklemme

0: Gerätediagnose abgeschaltet

1: Gerätediagnose aktiv (Default)

### D: Diagnosedaten

digitaler Klemmen ins Prozessabbild einblenden (z.B. KL2212). Diese Flag wird nur ausgewertet, wenn die Gerätediagnose aktiv ist (siehe oben).

0: Nicht einblenden

1: Einblenden (Default)

## Prozessabbildbeschreibung

Tabelle 0, Register 3 enthält die Prozessabbildbeschreibung und ist wie folgt codiert (Default-Wert: 0x0903):

MSB								LSB							
0	0	0	0	k1	k0	f1	f0	0	0	a	0	d	k	1	1

### k0...k1: Reaktion auf K-Bus-Fehler

0,2: Eingänge bleiben unverändert (Default=2);

1: Eingänge auf 0 setzen (TxPDO mit Nullen wird verschickt)

### f0...f1: Reaktion auf Feldbusfehler

0: Stoppen der K-Bus Zyklen, Watchdog auf Klemmen spricht an, Fehlerausgangswerte werden aktiv. Beim Neustart werden zunächst die alten Ausgangswerte gesetzt.

1: Ausgänge auf 0 setzen, Stoppen der K-Bus Zyklen (Default). 2: Ausgänge bleiben unverändert.

### a: Word-Alignment von Analog- und Sonderklemmen

0: kein Alignment (Default)

1: Daten auf Wortgrenzen mappen (Prozessdatum beginnt stets auf gerader Adresse im PDO)

**d: Datenformat komplexe Klemmen (Analog- und Sonderklemmen)**

0: Intel-Format (Default)

1: Motorola-Format

**k: Auswertung komplexe Klemmen (Analog- und Sonderklemmen)**

0: nur Nutzdaten (Default)

1: komplette Auswertung (Achtung: Analogkanäle benötigen dann statt z.B. 2 Eingangsbytes je 3 Eingangs- und 3 Ausgangsbytes; statt 4 Kanäle je PDO werden für 2 Kanäle je ein Rx- und ein TxPDO benötigt)

**Register-Kommunikation Busklemme/Erweiterungsbox**

Register-Kommunikation Busklemme/Erweiterungsbox

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x4501	0	Access Terminal Register	Unsigned8	ro	N	keiner	Index 0x4501 ermöglicht den Zugriff auf alle Register der Busklemmen bzw. Erweiterungsmodule. Subindex 0 enthält die Anzahl der gesteckten Busklemmen.
	1	Access Reg. Terminal 1	Unsigned32	rw	N	keiner	Zugriff Register Busklemme bzw. E-Modul 1
	...	...	...	...	...	...	...
	0XFE	Access Reg. Terminal 254	Unsigned32	rw	N	keiner	Zugriff Register Busklemme bzw. E-Modul 254

Der 32Bit-Wert ist wie folgt aufgebaut:

MSB		LSB	
Zugriff (Bit7) + Kanalnummer (Bit 6...0)	Registernummer	High-Byte Registerwert	Low-Byte Registerwert
[0..1] + [0...0x7F]	[0...0xFF]	[0...0xFF]	[0...0xFF]

Wie bei CANopen üblich wird das LSB zuerst und das MSB zuletzt übertragen.

Durch Zugriff auf Index 0x4501 können die Anwenderregister der Busklemmen bzw. Erweiterungsmodule beschrieben oder gelesen werden. Die Baugruppen verfügen über einen Registersatz je Ein- bzw. Ausgangskanal. Die Adressierung der Baugruppen erfolgt über den Subindex, die Kanalnummer und Register werden im 32Bit-Datenwert adressiert. Hierbei entspricht die Kanalnummer 0 dem ersten Kanal, 1 dem zweiten Kanal etc.

## Registerwert lesen

Zunächst muss dem Koppler mitgeteilt werden, welches Register gelesen werden soll. Hierzu muss ein SDO-Schreibzugriff auf die entsprechende Index/Subindex-Kombination erfolgen mit:

- Kanalnummer (Zugriffs-Bit=0) in Byte 3
- Registeradresse in Byte 2 des 32 Bit Datenwertes.

Bytes 1 und 0 werden nicht ausgewertet, wenn das Zugriffs-Bit (MSB in Byte 3) = 0 ist. Anschließend kann der Registerwert auf derselben Index/Subindex-Kombination gelesen werden.

Der Koppler setzt das Zugriffs-Bit nach dem Schreiben der auszulesenden Registeradresse so lange auf 1, bis der korrekte Wert zur Verfügung steht. Beim SDO-Lesezugriff ist also zu überprüfen, dass die Tabellennummer im Wertebereich 0...0x7F liegt.

Ein Zugriffsfehler bei der Register-Kommunikation wird durch entsprechende Rückgabewerte des SDO-Protokolls angezeigt (siehe Kapitel SDO, Abbruch Parameterkommunikation).

## Beispiel Registerwert lesen

Bei einer Thermoelement-Eingangsklemme KL3202 soll festgestellt werden, auf welchen Thermoelement-Typ der zweite Eingangs-Kanal eingestellt ist. Hierzu muss das Feature-Register 32 gelesen werden. Die Klemme befindet sich am fünften Steckplatz neben dem Buskoppler. Es müssen also folgende SDO-Telegramme gesendet werden:

Schreibzugriff (Download Request) auf Index 4501, Subindex 5 mit 32 Bit Datenwert 01 20 00 00 (0x01 = 2. Kanal, 0x20 = Register 32)  
Id=0x600+Node-ID DLC=8; Data=23 01 45 05 00 00 20 01

Anschließend Lesezugriff (Upload Request) auf den gleichen Index/Subindex, hierbei ist der Datenwert beliebig (hier: 0x00).  
Id=0x600+Node-ID DLC=8; Data=40 01 45 05 00 00 00 00

Der Koppler antwortet mit dem Upload Response Telegramm:  
Id=0x580+Node-ID DLC=8; Data=43 01 45 05 06 31 20 01

Es steht hier also der Wert 31 06 im Feature-Register. Die obersten 4 Bit kennzeichnen den Thermoelement-Typ. Sie sind hier 3, demnach ist der eingestellte Typ für diesen Kanal PT500 (siehe Dokumentation KL3202).

## Registerwert Schreiben

SDO-Schreibzugriff auf die entsprechende Index/Subindex-Kombination mit:

- Kanalnummer + 0x80 (Zugriffs-Bit=1) in Byte 3
- Registeradresse in Byte 2
- High-Byte Registerwert in Byte 1
- Low-Byte Registerwert in Byte 0 des 32 Bit Datenwertes

### HINWEIS

#### Achtung

Wenn der Schreibschutz nicht aufgehoben wurde (z.B. fehlerhaftes Codewort), so wird ein Schreibzugriff auf die Klemmenregister zwar bestätigt (SDO Download Response), der Wert jedoch nicht in das Register übernommen. Es wird deshalb empfohlen, den geschriebenen Wert anschließend auszulesen und zu vergleichen.

## Klemmen-Schreibschutz aufheben

Bevor die Anwender-Register der Busklemmen (Register 32-xx, je nach Klemmentyp bzw. Erweiterungsmodul) beschrieben werden können muss zunächst der Schreibschutz aufgehoben werden. Hierzu wird das folgende Codewort in das Register 31 des entsprechenden Kanals geschrieben:

Schreibschutz	Kanal	Register	Wert	entsprechender SDO Download-Wert (0x4500/0)
	1,2, 3 oder 4	31 (0x1F)	4661 (0x1235)	8y 1F 12 35 (y=Kanalnummer)

**Klemmen-Schreibschutz aufheben (CAN Darstellung)**

Um den Klemmen-Schreibschutz aufzuheben muss also das folgende SDO-Telegramm an den Koppler geschickt werden:

Id=600 + Node-ID DLC=8; Data=23 01 45 xx 35 12 1F 8y

wobei xx den Steckplatz der Klemme und y den Kanal kennzeichnen.

**Beispiel Schreibschutz aufheben**

Steckt also beispielsweise an einem BK5120 mit der Knotenadresse 3 eine Thermoelement-Eingangsklemme KL3202 an Steckplatz 5, so ist der Schreibschutz für den ersten Kanal wie folgt aufzuheben:

Id=0x603 DLC=8; Data=23 01 45 05 35 12 1F 80

Für den zweiten Kanal ist folgendes Telegramm zu senden:

Id=0x603 DLC=8; Data=23 01 45 05 35 12 1F 81

**Beispiel Registerwert Schreiben**

Der Thermoelement-Typ des zweiten Kanals der KL3202 Klemme an Steckplatz 5 soll nun auf PT1000 umgestellt werden. Hierzu müssen die obersten 4 Bits (oberstes Nibble) im Feature-Register mit dem Wert 2 beschrieben werden. Es wird davon ausgegangen, dass für alle anderen Bits des Feature-Registers die Default-Werte übernommen werden sollen. Nachdem der Schreibschutz aufgehoben wurde, ist per SDO Schreibzugriff (Download Request) der folgende 32Bit-Wert auf Index 0x4501, Subindex 05 zu schreiben: 81 20 21 06 (0x81=01+0x80; 0x20=32;0x2106 = Registerwert).

Das entsprechende Telegramm sieht auf dem Bus wie folgt aus:

Id=0x600+Node-ID DLC=8; Data=23 01 45 05 06 21 20 81

**PDOs aktivieren**

PDOs aktivieren

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x5500	0	Activate PDO Defaults	Unsigned32	rw	N	0x00000000	setzt PDO Communication Parameter für PDO 2...11

CANopen definiert Default-Identifizier für jeweils 4 Sende (Tx) und Empfangs (Rx) PDOs, alle anderen PDOs sind nach dem Aufstarten der Knoten zunächst deaktiviert. Über den Index 0x5500 lassen sich alle PDOs aktivieren, die gemäß Klemmenbestückung mit Prozessdaten vorbelegt sind (herstellerspezifisches Default Mapping). Dabei wird für PDO5...11 eine herstellereigene Default-Identifizier-Verteilung vorgenommen sowie für PDO 2...11 der Transmission Type und eine einheitliche Inhibit Zeit eingestellt. Nicht mit Prozessdaten versehene (also in der aktuellen Konfiguration überzählige) PDOs werden nicht aktiviert.



**Hinweis**

Dieses Objekt kann nur im Pre-Operational Zustand beschrieben werden!

Der 32Bit-Wert wird wie folgt verwendet:

MSB		LSB	
Transmission Type RxPDOs	Transmission Type TxPDOs	High-Byte Inhibit Zeit	Low-Byte Inhibit Zeit

Wie bei CANopen üblich wird das LSB zuerst und das MSB zuletzt übertragen.

### Beispiel:

PDOs aktivieren für Busknoten Nummer 1, Inhibit Zeit auf 10ms (=100 x 100µs) setzen, Transmission Type TxPDOs auf 255 setzen, Transmission Type RxPDOs auf 1 setzen. Folgendes Telegramm ist zu senden:  
Id=0x601 DLC=8; Data=23 00 55 00 64 00 FF 01

Der Knoten antwortet mit folgendem Telegramm:  
Id=0x601 DLC=8; Data=60 00 55 00 00 00 00 00

### Verwendete Identifier

Die Default-Identifier-Verteilung für die zusätzlichen PDOs läßt die vordefinierten Bereiche für Guarding, SDOs etc. frei, geht ab PDO6 von maximal 64 Knoten im Netz aus und erfolgt nach folgendem Schema:

Objekt	Function Code	resultierende COB-ID (hex)	resultierende COB-ID (dez)
TxPDO5	1101	0x681 - 0x6BF	1665 - 1727
RxPDO5	1111	0x781 - 0x7BF	1921 - 1983
TxPDO6	00111	0x1C1 - 0x1FF	449 - 511
RxPDO6	01001	0x241 - 0x27F	577 - 639
TxDPO7	01011	0x2C1 - 0x2FF	705 - 767
RxPDO7	01101	0x341 - 0x37F	833 - 895
TxPDO8	01111	0x3C1 - 0x3FF	961 - 1023
RxPDO8	10001	0x441 - 0x47F	1089 - 1151
TxPDO9	10011	0x4C1 - 0x4FF	1217 - 1279
RxPDO9	10101	0x541 - 0x57F	1345 - 1407
TxDPO10	10111	0x5C1 - 0x5FF	1473 - 1535
RxPDO10	11001	0x641 - 0x67F	1601 - 1663
TxPDO11	11011	0x6C1 - 0x6FF	1729 - 1791
RxPDO11	11101	0x741 - 0x77F	1857 - 1919

### HINWEIS

#### Achtung

Es ist darauf zu achten, dass der Index 0x5500 nicht genutzt wird, wenn Buskoppler mit mehr als 5 PDOs in Netzen mit Knoten-Adressen >64 vorhanden sind, da es sonst zu Identifier-Überschneidungen kommen kann. In diesem Fall müssen die PDO Identifier individuell eingestellt werden.

Der Übersichtlichkeit halber sind die nach CANopen definierten Default-Identifier hier ebenfalls aufgeführt:

Objekt	Function Code	resultierende COB-ID (hex)	resultierende COB-ID (dez)
<b>Emergency</b>	0001	0x81 - 0xBF [0xFF]	129 - 191 [255]
<b>TxPDO1</b>	0011	0x181 - 0x1BF [0x1FF]	385 - 447 [511]
<b>RxPDO1</b>	0100	0x201 - 0x23F [0x27F]	513 - 575 [639]
<b>TxPDO2</b>	0101	0x281 - 0x2BF [0x2FF]	641 - 676 [767]
<b>RxPDO2</b>	0110	0x301 - 0x33F [0x37F]	769 - 831 [895]
<b>TxDPO3</b>	0111	0x381 - 0x3BF [0x3FF]	897 - 959 [1023]
<b>RxPDO3</b>	1000	0x401 - 0x43F [0x47F]	1025 - 1087 [1151]
<b>TxPDO4</b>	1001	0x481 - 0x4BF [0x4FF]	1153 - 1215 [1279]
<b>RxPDO4</b>	1010	0x501 - 0x53F [0x57F]	1281- 1343 [1407]
<b>SDO (Tx)</b>	1011	0x581 - 0x5BF [0x5FF]	1409 - 1471 [1535]
<b>SDO (Rx)</b>	1100	0x601 - 0x63F [0x67F]	1537 - 1599 [1663]
<b>Guarding / Heartbeat/ Bootup</b>	1110	0x701 - 0x73F [0x77F]	1793 - 1855 [1919]

Angegeben sind die Identifier, die sich aus den DIP-Schalter-Einstellungen am Koppler ergeben, sowie in eckigen Klammern der Identifier-Bereich für die Knotenadressen 64...127 (am Buskoppler BK5110, BK5120 und LC5100 nicht einstellbar). Bei den Feldbus Box-Modulen und dem Buskoppler BK515x lassen sich die Adressen 1...99 einstellen.

Eine tabellarische Übersicht über alle Identifier findet sich im [Anhang \[► 138\]](#).

**Digitale Eingänge**

Digitale Eingänge

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x6000</b>	0	Anzahl Elemente	Unsigned8	ro	N	abhängig von Typ und Bestückung	Anzahl verfügbarer digitaler 8-Bit Eingangsdatenblöcke
	1	1st input block	Unsigned8	ro	Y	0x00	1. Eingangskanal
	...	...	...	...	...	...	...
	0XFE	254. input block	Unsigned8	ro	Y	0x00	254. Eingangskanal

**Interrupt Maske**

Interrupt Maske

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x6126</b>	0	Anzahl Elemente	Unsigned8	ro	N	abhängig von Typ	Anzahl der 32-Bit Interrupt Masken = 2 x Anzahl TxPDOs
	1	IR-Mask0 TxPDO1	Unsigned32	rw	N	0xFFFFFFFF	IR-Maske Bytes 0...3 TxPDO1
	2	IR-Mask1 TxPDO1	Unsigned32	rw	N	0xFFFFFFFF	IR-Maske Bytes 4...7 TxPDO1
	3	IR-Mask0 TxPDO2	Unsigned32	rw	N	0xFFFFFFFF	IR-Maske Bytes 0...3 TxPDO2
	...	...	...	...	...	...	...
	0x20	IR-Mask1 TxPDO16	Unsigned32	rw	N	0xFFFFFFFF	IR-Maske Bytes 4...7 TxPDO16

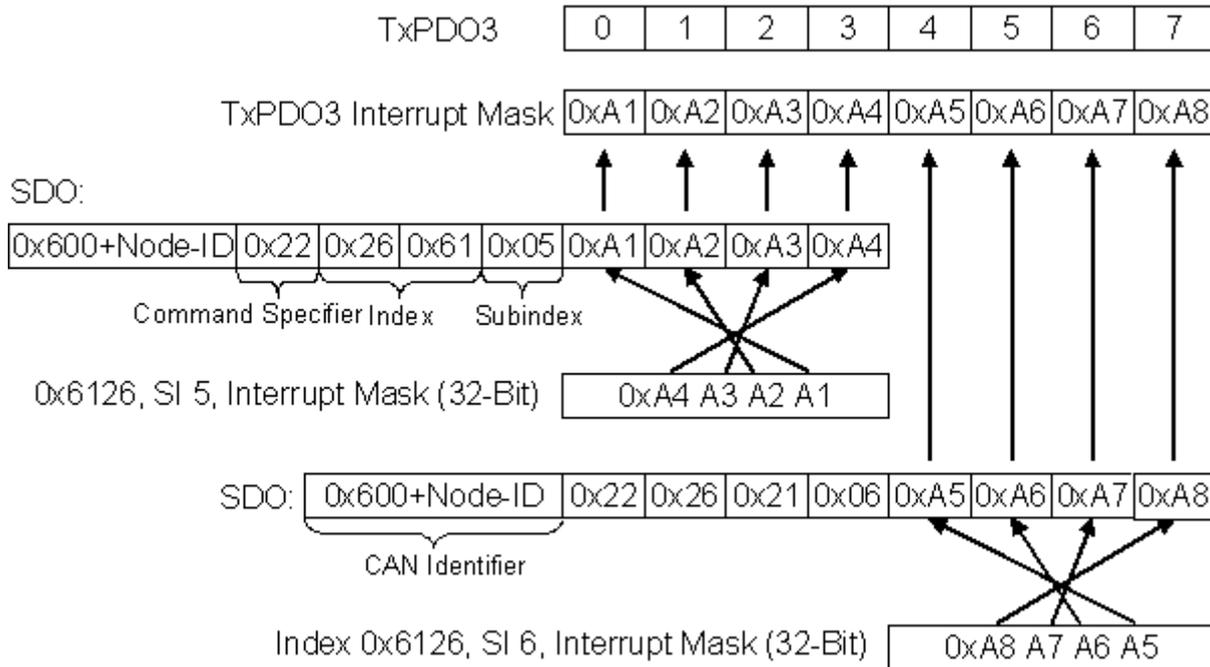
Per Default führt jede Änderung eines Wertes im ereignisgesteuerten PDO zum Versenden des Telegramms. Mit der Interrupt Maske kann bestimmt werden, welche Daten-Änderungen hierfür ausgewertet werden. Durch Nullen der entsprechenden Bereiche innerhalb der PDOs werden diese bei der Ereignissteuerung ("Interrupt-Steuerung") ausmaskiert. Die Interrupt Maske umfasst nicht nur die PDOs mit digitalen Eingängen, sondern alle vorhandenen TxPDOs. Falls die TxPDOs kürzer als 8 Bytes sind, wird der überzählige Teil der IR-Maske nicht ausgewertet.

Die Interrupt Maske beeinflusst nur TxPDOs mit Transmission Type 254 und 255. Sie wird nicht auf dem Gerät gespeichert (auch nicht durch das Objekt 0x1010). Änderungen der Maske zur Laufzeit (im Operational Status) sind möglich und werden bei der nächsten Eingangsdaten-Änderung ausgewertet.

Die Interrupt Maske wird für TxPDOs mit analogen Eingangsdaten nicht ausgewertet, wenn für die Eingänge Grenzwerte (0x6424, 0x6425) oder die Delta Funktion (0x6426) aktiviert wurden.

Dieser Eintrag ist ab Firmware Stand C3 implementiert.

**Beispiel zur Zuordnung der Daten**



**Anwendungsbeispiel**

Der Zählerwert eines schnellen Zählereingangs soll nur übertragen werden, sobald sich Bits im Statuswort (z.B. der Latch-Eingang) geändert haben. Hierzu muss der 32-Bit Zählerwert in der Interrupt Maske ausmaskiert (=genullt) werden. Der Status befindet sich im Byte 0, der Zählerwert liegt per Default in den Bytes 1..4 des entsprechenden PDOs (im Beispiel TxPDO3, da <65 digitale und <5 analoge Eingänge vorhanden sind).

Also muss in Index 0x6126, Subindex5 der Wert 0x0000 00FF und in Subindex6 der Wert 0xFFFF FF00 eingetragen werden.

Die entsprechenden SDOs sehen demnach wie folgt aus:

11-bit Identifier	8 Byte Nutzdaten							
0x600+ Node-ID	0x22	0x26	0x61	0x05	0xFF	0x00	0x00	0x00

11-bit Identifier	8 Byte Nutzdaten							
0x600+ Node-ID	0x22	0x26	0x61	0x06	0x00	0xFF	0xFF	0xFF

**Digitale Ausgänge**

Digitale Ausgänge

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x6200</b>	0	Anzahl Elemente	Unsigned8	ro	N	abhängig von Typ und Bestückung	Anzahl verfügbarer digitaler 8-Bit Ausgangsdatenblöcke
	1	1st input block	Unsigned8	rw	Y	0x00	1. Ausgangskanal
	...	...	...	...	...	...	...
	0XFE	254. input block	Unsigned8	rw	Y	0x00	254. Ausgangskanal

### Analoge Eingänge

Analoge Eingänge

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x6401</b>	0	Anzahl Elemente	Unsigned8	ro	N	abhängig von Typ und Bestückung	Anzahl verfügbarer analoger Eingangskanäle
	1	1st input	Unsigned16	ro	Y	0x0000	1. Eingangskanal
	...	...	...	...	...	...	...
	0XFE	254. input	Unsigned16	ro	Y	0x0000	254. Eingangskanal

Die analogen Signale werden linksbündig dargestellt. Damit wird die Darstellung im Prozessabbild unabhängig von der tatsächlichen Auflösung. Details zum Datenformat finden sich beim jeweiligen Signaltyp.

### Analoge Ausgänge

Analoge Ausgänge

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x6411</b>	0	Anzahl Elemente	Unsigned8	ro	N	abhängig von Typ und Bestückung	Anzahl verfügbarer analoger Ausgangskanäle
	1	1st input block	Unsigned16	rw	Y	0x0000	1. Ausgangskanal
	...	...	...	...	...	...	...
	0XFE	254. input block	Unsigned16	rw	Y	0x0000	254. Ausgangskanal

Die analogen Signale werden linksbündig dargestellt. Damit wird die Darstellung im Prozessabbild unabhängig von der tatsächlichen Auflösung. Details zum Datenformat finden sich beim jeweiligen Signaltyp.

**Ereignissteuerung Analoge Eingänge**

Ereignissteuerung Analoge Eingänge

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x6423	0	Global Interrupt Enable	Boolean	rw	N	FALSE (0)	Aktiviert das ereignisgesteuerte Senden von PDOs mit Analogeingängen.

Nach CANopen sind die Analogeingänge in TxPDO2..4 zwar per Default auf den Transmission Type 255 (ereignisgesteuert) eingestellt, jedoch ist das Ereignis (die Änderung eines Eingangswertes) über die Ereignissteuerung im Objekt 0x6423 deaktiviert, um ein Überfluten des Busses mit Analogsignalen zu verhindern. Es empfiehlt sich, das Datenaufkommen der Analog-PDOs entweder durch synchrone Kommunikation oder durch Verwendung des Event Timers zu kontrollieren. Im ereignisgesteuerten Betrieb kann das Sendeverhalten der Analog-PDOs vor dem Aktivieren durch Einstellen von Inhibit-Zeit (Objekt 0x1800ff, Subindex 3) und/oder Grenzwertüberwachung (Objekt 0x6424 + 0x6425) und/oder Deltafunktion (Objekt 0x6426) parametrisiert werden.

**Oberer Grenzwert Analoge Eingänge**

Oberer Grenzwert Analoge Eingänge

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
0x6424	0	Anzahl Elemente	Unsigned8	ro	N	abhängig von Typ und Bestückung	Anzahl verfügbarer analoger Eingangskanäle
	1	upper limit 1st input	Unsigned16	rw	Y	0x0000	Oberer Grenzwert 1. Eingangskanal
	...	...	...	...	...	...	...
	0XFE	upper limit 254. input	Unsigned16	rw	Y	0x0000	Oberer Grenzwert 254. Eingangskanal

Werte ungleich 0 aktivieren den oberen Grenzwert für diesen Kanal. Ein PDO wird dann abgesetzt wenn dieser Grenzwert überschritten wird. Zusätzlich muss die Ereignissteuerung aktiviert sein (Objekt 0x6423). Das Datenformat entspricht dem der Analogeingänge.

**Unterer Grenzwert Analoge Eingänge**

Unterer Grenzwert Analoge Eingänge

Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x6425</b>	0	Anzahl Elemente	Unsigned8	ro	N	abhängig von Typ und Bestückung	Anzahl verfügbarer analoger Eingangskanäle
	1	lower limit 1st input	Unsigned16	rw	Y	0x0000	Unterer Grenzwert 1. Eingangskanal
	...	...	...	...	...	...	...
	0XFE	lower limit 254. input	Unsigned16	rw	Y	0x0000	Unterer Grenzwert 254. Eingangskanal

Werte ungleich 0 aktivieren den unteren Grenzwert für diesen Kanal. Ein PDO wird dann abgesetzt wenn dieser Grenzwert unterschritten wird. Zusätzlich muss die Ereignissteuerung aktiviert sein (Objekt 0x6423). Das Datenformat entspricht dem der Analogeingänge.

### Deltafunktion Analoge Eingänge

Deltafunktion Analoge Eingänge

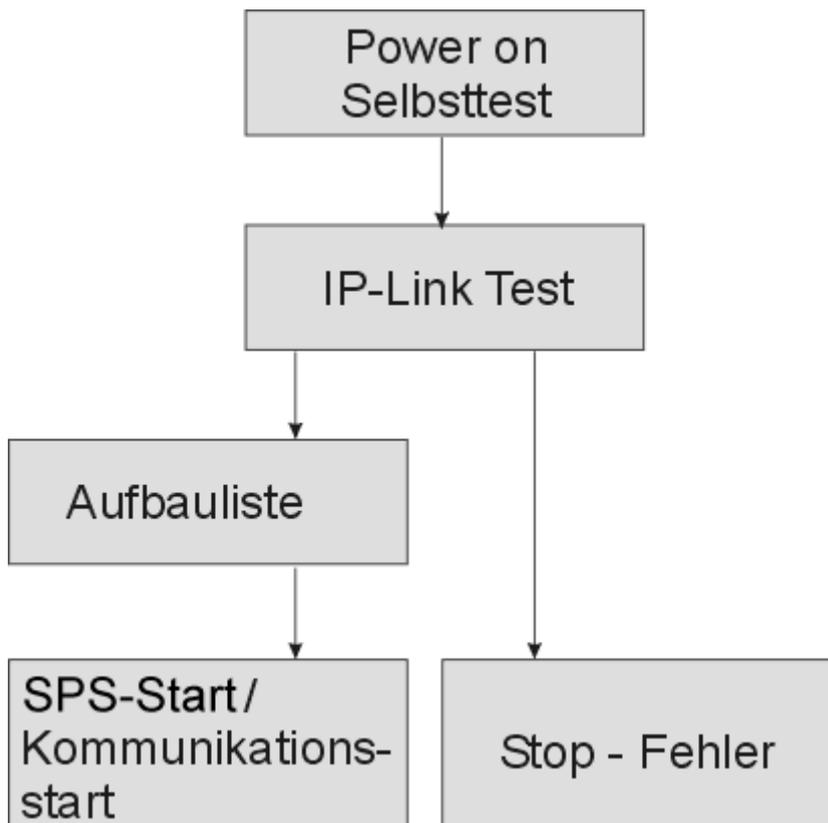
Index	Subindex	Name	Typ	Attrb.	Map.	Default-Wert	Bedeutung
<b>0x6426</b>	0	Anzahl Elemente	Unsigned8	ro	N	abhängig von Typ und Bestückung	Anzahl verfügbarer analoger Eingangskanäle
	1	delta value 1st input	Unsigned16	rw	Y	0x0000	Deltawert 1. Eingangskanal
	...	...	...	...	...	...	...
	0XFE	delta value 254. input	Unsigned16	rw	Y	0x0000	Deltawert 254. Eingangskanal

Werte ungleich 0 aktivieren die Deltafunktion für diesen Kanal. Ein PDO wird dann abgesetzt wenn sich der Wert seit dem letzten Senden um mehr als den Deltawert verändert hat. Zusätzlich muss die Ereignissteuerung aktiviert sein (Objekt 0x6423). Das Datenformat entspricht dem der Analogeingänge (Deltawert: nur positive Werte).

## 4 Parametrierung und Inbetriebnahme

### 4.1 Anlaufverhalten der Feldbus Box

Nach dem Einschalten prüft die Feldbus Box ihren Zustand, konfiguriert (wenn vorhanden) den IP-Link ist und erstellt anhand der Erweiterungsmodule eine Aufbauliste. Wenn die Feldbus Box eine dezentrale Steuerung besitzt (IL230x-C310) wird nach einer erfolgreichen Aufbauliste die lokale SPS gestartet. Beim Hochlauf des Moduls leuchten und blinken die E/A LEDs. Im fehlerfreien Zustand sollte nach ca. 2-3 sec keine E/A LED mehr Blinken. Sollte ein Fehler vorliegen, hängt es von der Fehlerart ab, welche LED blinkt (siehe Kapitel Diagnose LEDs).



## 4.2 Adresse

Vor Inbetriebnahme der Feldbus Box muss die Knotennummer (Node-ID) eingestellt werden. Die Knotennummer ist über die zwei Drehwahlschalter hinter der Klarsichtabdeckung einzustellen. Die Default-Einstellung beträgt 11. Es sind alle Adressen >0 erlaubt, dabei darf jede Adresse im Netzwerk nur einmal vorkommen. Die Adresse wird im ausgeschalteten Zustand der Feldbus Box geändert. Dafür lösen Sie die Abdeckung, und verändern Sie mit Hilfe eines Schraubendrehers die Schalter auf die gewünscht Position. Dabei ist zu beachten, dass die Schalter richtig einrasten. Der linke Schalter ist der Zehner Multiplikator und der rechte Schalter der Einer Multiplikator. Die Adressänderung wird aktiv, sobald das Gerät eingeschaltet wird.

### Beispiel

Sie wollen Adresse 34 einstellen.

Linker Drehwahlschalter 3

Rechter Drehwahlschalter 4



### 4.3 Baud-Rate

Bei der Feldbus Box muss die Baud-Rate nicht eingestellt werden. Diese Baugruppen verfügen über eine Auto-Baud-Ratenfunktion.

Über die Tabelle 100 kann eine feste Baudrate eingestellt werden (siehe unten).

Für die Auto-Baud-Ratenerkennung ist erforderlich, dass mehrere gültige Telegramme der gewünschten Baud-Rate auf dem Bus vorhanden sind. Während der Baud-Ratensuche blinken die RUN und die CAN ERR LED in schnellem Wechsel. Sobald eine Baud-Rate erkannt und übernommen wurde fährt die Feldbus Box mit der Initialisierung fort.

Ein Software-Reset führt nicht zur erneuten Aktivierung der Autobaud-Funktion, die vorher aktive Baud-Rate bleibt erhalten.

#### Bit Timing

Bit Timing

Folgende Baud-Raten und Bittiming Registereinstellungen werden von den Beckhoff CANopen-Geräten unterstützt:

Baud-Rate [kBaud]	BTR0	BTR1	Sampling Point
1000	0x00	0x14	75%
800	0x00	0x16	80%
500	0x00	0x1C	87%
250	0x01	0x1C	87%
125	0x03	0x1C	87%
100	0x04	0x1C	87%
50	0x09	0x1C	87%
20	0x18	0x1C	87%
10	0x31	0x1C	87%

Die angegebenen Bit-Timing Registereinstellungen (BTR0, BTR1) gelten z.B. für die CAN-Controller Philips 82C200, SJA1000, Intel 80C527, Siemens 80C167, und andere. Sie sind für maximale Buslänge optimiert.

#### Feste Baudrateneinstellung

Mit Hilfe der Konfigurations-Software KS2000 können Sie die Einstellungen de CANopen-Moduls ändern. Dies ist ab FW Stand C6 möglich. Die aktuelle Firmware-Version ihres des Moduls wird Ihnen von der Konfigurations-Software KS2000 angezeigt.



#### Hinweis

Um die Einstellungen verwenden zu können, müssen Sie zuvor den Register-Schreibschutz mit Hilfe der Konfigurations-Software KS2000 deaktivieren.

**Bedeutung der Einträge in Tabelle 100**

Tabelle 100, Offset	Beschreibung	Default
000	fest einzustellende Baudrate	0 (Siehe Tabelle )
001	reserviert	reserviert
002	reserviert	reserviert
003	reserviert	reserviert
004	reserviert	reserviert
005	Autobaud aktiv	0
006	reserviert	reserviert
007	reserviert	reserviert
008	reserviert	reserviert
009	reserviert	reserviert
010	Aktivieren einer festen Baudrate	0
011-018	reserviert	reserviert

**Beispiel zum Einstellen einer festen Baudrate**

In Register 0 der Tabelle 100 kann eine feste Baudrate hinterlegt werden. In Register 10 wird die Einstellung aktiviert.

Beispiel für 50kBaud:

- Deaktivieren Sie den Schreibschutz
- Öffnen Sie Tabelle 100.
- Schreiben Sie den Wert 0x0007 in Offset 0 (siehe Tabelle der unterstützten Baudraten)
- Schreiben Sie den Wert 0x8000 in Offset 10
- Schreiben Sie den Wert 0x0001 in Offset 5
- Starten Sie das Modul neu

Die feste Baudrate von 50 kBaud ist jetzt eingestellt!

*Tab. 1: Unterstützte Baudraten*

Wert in Tabelle 100, Offset 0	Baudrate
0x0000	1 MBaud/Auto
0x0001	800 kBaud
0x0002	500 kBaud
0x0003	400 kBaud
0x0004	250 kBaud
0x0005	125 kBaud
0x0006	100 kBaud
0x0007	50 kBaud
0x0008	20 kBaud
0x0009	10 kBaud

## 4.4 Mapping der Feldbus Boxen

Während bei anderen Feldbus-Systemen meist das gesamte Prozessabbild zyklisch übertragen wird, teilt CANopen die Prozessdaten in Prozessdatenobjekte (PDOs) mit maximal 8 Datenbytes auf. Unter PDO-Mapping versteht man die Abbildung der Applikationsobjekte (Echtzeitdaten) aus dem Objektverzeichnis in die Prozessdatenobjekte. Die CANopen-Geräteprofile sehen für jeden Gerätetyp ein Default Mapping vor, das für die meisten Anwendungen passend ist. So bildet das Default Mapping für digitale E/A einfach die Ein- bzw. Ausgänge ihrer physikalischen Reihenfolge gemäß in die Sende- bzw. Empfangs-Prozessdatenobjekte ab.

Im zweiten PDO finden sich die ersten 4 analogen Ein- bzw. Ausgänge.

Dementsprechend werden diese PDOs von den Beckhoff Feldbus I/O Modulen belegt - falls z.B. keine digitalen Ausgänge vorhanden sind, bleibt das RxPDO1 leer.

**Bei den Feldbus Boxen ist die PDO-Belegung demnach durch die jeweilige Signalvariante bestimmt: digitale Ein/Ausgangsdaten befinden sich in PDO1, analoge in PDO2, Sondersignale in PDO3.**

Die Koppler Box Module belegen die PDOs automatisch: die Koppler Box liest in der Aufstartphase ein, welche welche Erweiterungsbox Module vorhanden sind und ordnet die Daten den PDOs zu. Hierbei werden digitale, analoge und Sonderklemmen unterschieden und die PDOs jeweils "sortenrein" belegt. Es werden also keine unterschiedlichen Datentypen (z.B. digitale und analoge Eingänge) in ein PDO verpackt, sondern für jeden weiteren Datentyp wird ein neues PDO begonnen.

[Automatische PDO Belegung bei Beckhoff Buskopplern \[► 162\]](#)

Das aktuelle Mapping kann über entsprechende Einträge im Objektverzeichnis, die sogenannten Mapping-Tabellen, gelesen werden. An erster Stelle der Mapping Tabelle (Subindex 0) steht die Anzahl der gemappten Objekte, die im Anschluss aufgelistet sind. Die Tabellen befinden sich im Objektverzeichnis bei Index 0x1600ff für die RxPDOs bzw. 0x1A00ff für die TxPDOs.

### Prozessdatenzuordnung der Feldbus Box-Baugruppen

Aufgelistet sind die Objektverzeichniseinträge in denen sich die Prozessdaten der jeweiligen Baugruppe finden (Default-Einstellung). Durch Parameteränderungen (z.B. Prozessdatenlänge bei seriellen Schnittstellen) kann sich die Zuordnung ändern.

Details der Dateninhalte entnehmen Sie bitte der Feldbus-neutralen Dokumentation *Signalvarianten (Feldbus Box E/A-Module)* die Sie im Internet unter <http://www.beckhoff.de> im Bereich *Download* finden.

<b>Modul</b>	<b>Prozessdaten (Eingänge)</b>	<b>Prozessdaten (Ausgänge)</b>
IP10xx-B510 IE10xx	8 digitale Eingänge (0x6000)	-
IP15xx-B510 IE15xx	1 x 5-Byte Sonderklemme, Eingangsdaten (0x2A00)	1 x 5-Byte Sonderklemme, Ausgangsdaten (0x2B00)
IP20xx-B510 IE20xx	-	8 digitale Ausgänge (0x6200)
IP23xx-B510 IL230x-B510 IE23xx	4 digitale Eingänge (0x6000). Weitere digitale Eingangsdaten werden lückenlos angefügt (im Objekt 0x6000 und im TxPDO).	4 digitale Ausgänge (0x6200). Weitere digitale Ausgangsdaten werden lückenlos angefügt (im Objekt 0x6200 und im RxPDO).
IP240x-B510 IE240x	8 digitale Eingänge (0x6000)	8 digitale Ausgänge (0x6200)
IP25xx-B510 IE25xx	2 x 3-Byte Sonderklemme, Eingangsdaten (0x2600)	2 x 3-Byte Sonderklemme, Ausgangsdaten (0x2700)
IP3xxx-B510 IE3xxx	4 x analoge Eingänge (0x6401)	-
IP41xx-B510 IE41xx	-	4 x analoge Ausgänge (0x6411)
IP5009-B510 IE5009	1 x 4-Byte Sonderklemme, Eingangsdaten (0x2800)	1 x 4-Byte Sonderklemme, Ausgangsdaten (0x2900)
IP5109-B510 IE5109	1 x 6-Byte Sonderklemme, Eingangsdaten (0x2C00)	1 x 6 Byte Sonderklemme, Ausgangsdaten (0x2D00)
IP6xxx-B510 IE6xxx	1 x 4-Byte Sonderklemme, Eingangsdaten (0x2800)	1 x 4-Byte Sonderklemme, Ausgangsdaten (0x2900)

## 4.5 Konfiguration Feldbus

### 4.5.1 Konfigurationsdateien

Die Parameter und Einstellmöglichkeiten von CANopen-Geräten sind in Konfigurationsdateien (Electronic Data Sheet, eds) aufgelistet. Diese eds-Dateien können von Konfigurationstools gelesen werden. Struktur und Syntax der eds-Dateien ist in CiA DSP 306 definiert. Auf der Homepage des Verbandes CAN in Automation (<http://www.can-cia.de>) steht ein Tool zum Download bereit, mit dem die eds-Dateien auf Übereinstimmung mit der Norm geprüft werden können.

Die eds-Dateien für die BECKHOFF CANopen Buskomponenten finden Sie auf der BECKHOFF Homepage (<http://www.beckhoff.de>) sowie auf den BECKHOFF Produkt-CDs.

## 4.5.2 Überblick

Die Beckhoff CANopen Buskomponenten verfügen über vielfältige Konfigurations- und Einstellmöglichkeiten. Dennoch ist der Konfigurationsaufwand minimal, da für alle Parameter sinnvolle Default-Werte vorhanden sind. Mit diesen Voreinstellungen sind die Anforderungen der meisten Applikationen problemlos zu erfüllen.

Die folgende Liste gibt einen Überblick über die wichtigsten Konfigurationsmöglichkeiten der Geräte:

### Knotenadresse

[Knotenadresse \[► 94\]](#)

Diese ist in jedem Fall einzustellen - und zwar so, dass keine Knotenadresse doppelt vergeben wird.

### Baud-Rate

[Baud-Rate \[► 95\]](#)

Die Auto-Baud-Ratenfunktion macht eine manuelle Konfiguration überflüssig. Bei den Feldbus Box Baugruppen wurde daher auf eine manuelle Einstellmöglichkeit verzichtet.

### PDO Parameter

#### PDO-Identifizier

[PDO-Identifizier \[► 30\]](#)

Die CANopen Default-Identifizier-Verteilung sieht Identifizier für bis zu 4 Empfangs-Prozessdatenobjekte (RxPDOs) und 4 Sende-Prozessdatenobjekte (TxPDOs) vor. Damit sind CAN-Identifizier für die Daten von z.B. 64 digitale Ein/Ausgänge und 12 analoge Ein/Ausgänge vorhanden. Falls mehr Ein/Ausgänge vorhanden sind, so werden deren Daten per Default in PDO 5...16 gemappt (Details siehe [Default Mapping \[► 162\]](#)). Identifizier für PDO 5...11 lassen sich einfach durch Beschreiben des Objektes 0x5500 freischalten. Sollten mehr als 11 PDOs benötigt werden oder die [Default-Identifizier-Verteilung \[► 42\]](#) nicht den Applikationsanforderungen entsprechen, so lassen sich die Identifizier individuell einstellen (siehe Objekte 0x1400ff bzw. 0x1800ff).

#### PDO Kommunikationsart

PDO Kommunikationsart

Für jedes Prozessdatenobjekt kann individuell eingestellt werden, nach welchem Prinzip es kommuniziert wird: ereignisgesteuert (default), gepollt oder synchronisiert.

#### PDO-Mapping

[PDO-Mapping \[► 35\]](#)

Die Daten der Ein- und Ausgänge werden den Prozessdatenobjekten beim Aufstarten der Baugruppen zugeordnet (Default Mapping). Bei Bedarf kann diese Zuordnung (Mapping) verändert werden (siehe Objekte 0x1600ff bzw. 0x1A00ff).

#### Heartbeat/Guarding

Heartbeat/Guarding

Auch ohne spezielle Konfiguration antworten die Baugruppen auf Guarding-Anforderungen. Falls die Baugruppen selbsttätig Statusinformationen senden sollen (Heartbeat) oder falls die Baugruppen auf den Ausfall der Anforderungs-Telegramme oder des Master-Heartbeats reagieren sollen, so sind die entsprechenden Parameter einzustellen (Guarding: Objekt 0x100C ff.; Heartbeat: Objekt 0x1016 ff.)

**Die Liste sämtlicher über CAN erreichbarer Parameter findet sich im Objektverzeichnis.**

**Hinweis**

Die Objekte aus dem Objektverzeichnis sind per SDO-Zugriff, jedoch nicht generell über die Konfigurations-Software KS2000 erreichbar. Dagegen sind alle Register, die per KS2000 konfiguriert werden können, auch per SDO-Zugriff auf das Objektverzeichnis (Objekte [0x4500](#)removed link: [0x4500](#) und [0x4501](#)removed link: [0x4501](#)) erreichbar - wenn auch nicht mit dem gleichen Bedienungskomfort wie mit der Konfigurations-Software.

---

**Sehen Sie dazu auch**

 Konfigurations-Software KS2000 [[▶ 109](#)]

### 4.5.3 Konfiguration mit TwinCAT

Die Automatisierungssoftware TwinCAT ist eine vollständige Automatisierungslösung für PC-kompatible Rechner. TwinCAT erweitert jeden kompatiblen PC um eine Echtzeitsteuerung, eine IEC 61131-3 Multi-SPS, NC-Positionierung, die entsprechende Programmierumgebung und Bedienoberfläche. TwinCAT unterstützt mehrere verschiedene CANopen PC-Karten. Beckhoff empfiehlt die CANopen PCI Masterkarte FC5101, die auch als zweikanalige Variante (FC5102) erhältlich ist.

#### System Manager

Zur Konfiguration der FC510x CANopen PCI-Karte 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 die Mappings des Datenverkehrs.



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

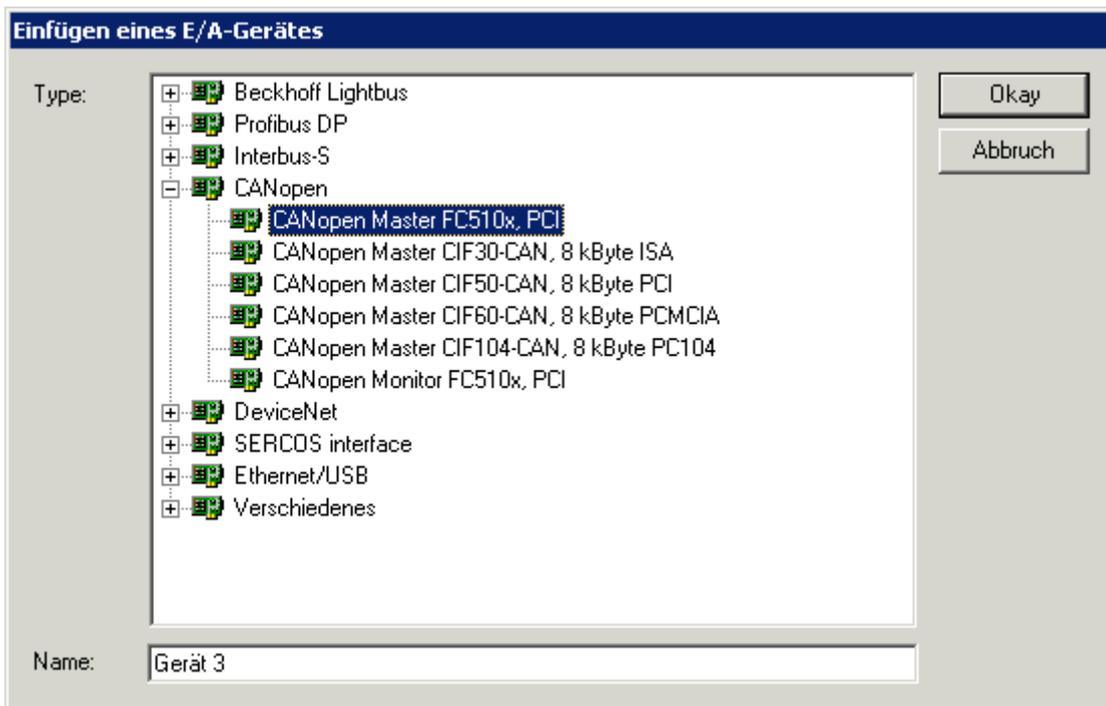
- ActiveX-Control (ADS-OCX) für z.B. Visual Basic, Visual C++, Delphi, etc
- DLL-Interface (ADS-DLL) für z.B. Visual C++ Projekte
- Script-Interface (ADS-Script DLL) für z.B. VBScript, JScript, etc.

#### Der TwinCAT System Manager weist folgende Eigenschaften auf:

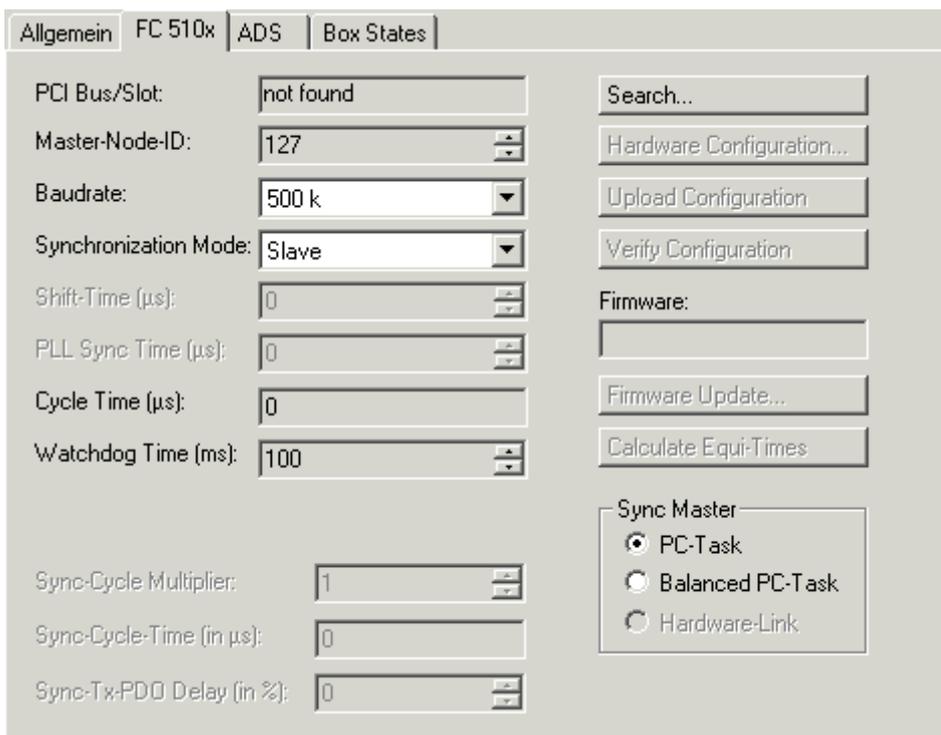
- Verbindung zwischen Server-Prozessabbildern und E/A-Kanälen bitweise
- Standard-Datenformate, z.B. Arrays und Strukturen
- Benutzerdefinierte Datenformate
- fortlaufende Verbindung von Variablen
- Drag und Drop
- Import und Export auf allen Ebenen

#### Vorgehen bei der Konfiguration der CANopen Ein/Ausgabebaugruppen

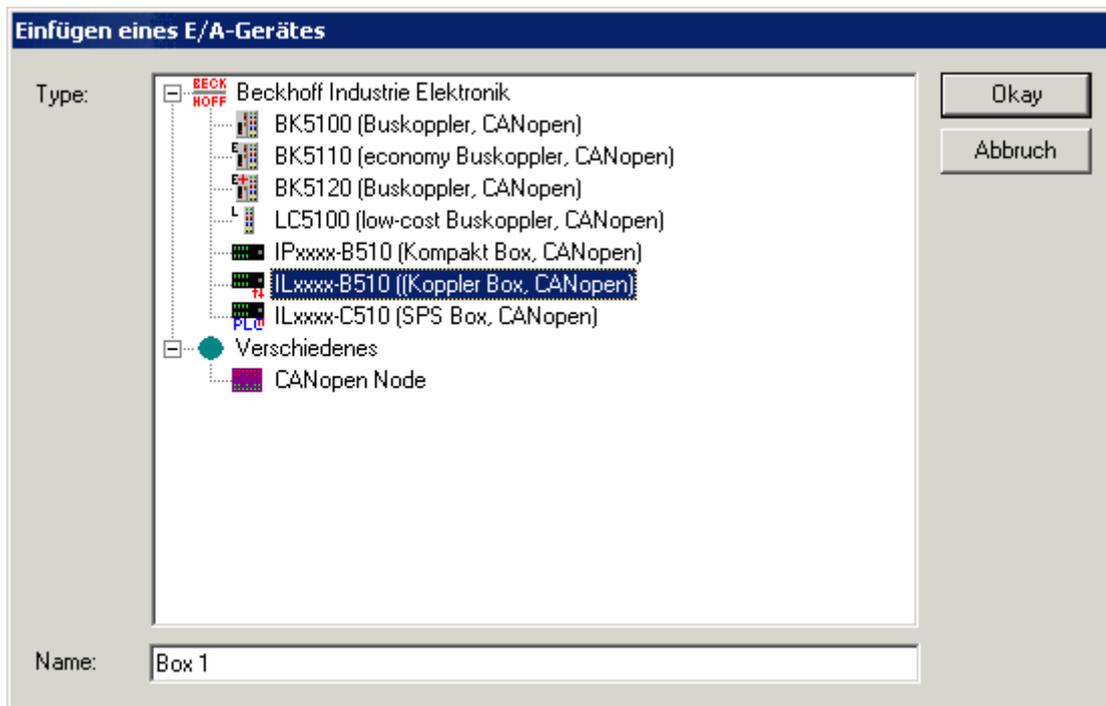
1. Zunächst wird die entsprechende CANopen Master PC-Karte ausgewählt und in die E/A-Konfiguration eingefügt



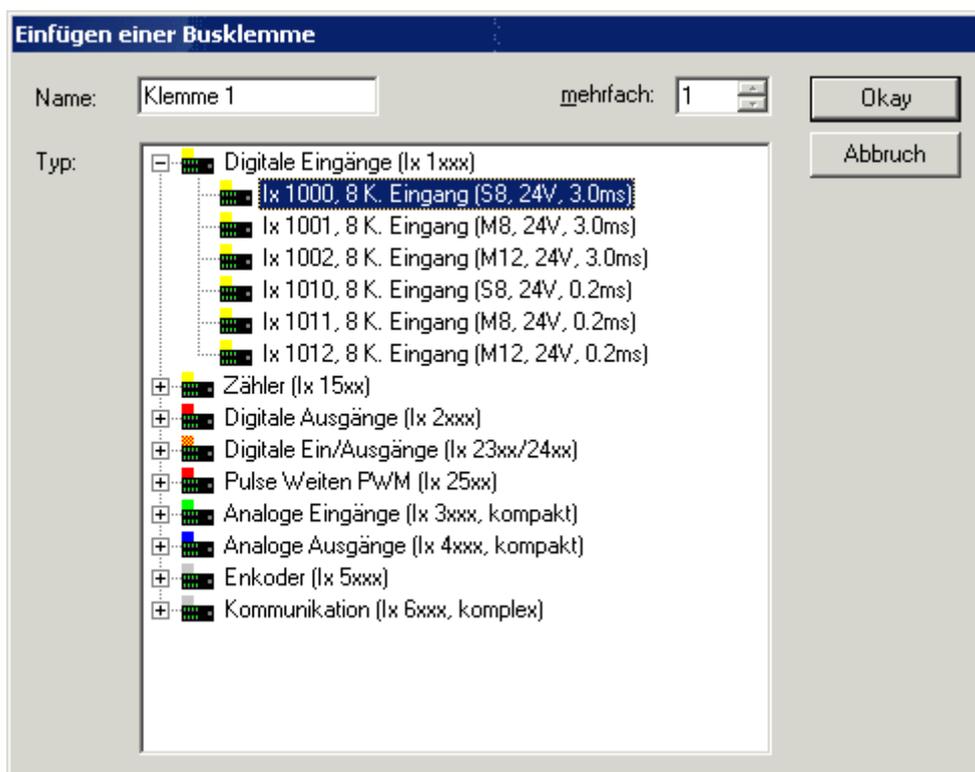
2. Nun werden die Baud-Rate und ggf. die Master-Node-ID (für das Heartbeat Protokoll) eingestellt.



3. Nach der Master-Karte werden dann die Busknoten eingefügt:



4. An die CANopen Kompakt oder Koppler Box werden nun die entsprechenden E/A-Varianten und Erweiterungsboxen angefügt.



5. Nun werden die Kommunikationseigenschaften für diesen Busknoten konfiguriert:

**Node Id**

Stellt die Node-ID des CAN Bus Teilnehmers ein (zwischen 1 und 99). Dieser Wert muss mit dem an der Feldbus Box eingestellten Wert übereinstimmen.

**Guard Time**

Zykluszeit für die Knotenüberwachung (Nodeguarding). Im Fall der FC5101 wird diese Zeit für als Producer Heartbeat Time verwendet.

**Life Time Factor**

Mit Guard Time multipliziert ergibt sich die Watchdog-Zeit für die Überwachung des Masters durch den Koppler (Lifeguarding). Lifeguarding ist deaktiviert, wenn Life Time Factor zu null gesetzt wird. Bei Busknoten, die Heartbeat unterstützen, wird die Watchdog Zeit als Consumer Heartbeat Time verwendet.

**Inhibit Time**

Gibt den minimalen Sendeabstand für PDOs (Telegramme) mit analogen und Sondersignalen an. Wenn mehr als 64 digitale Signale vorhanden, werden diese auch mit dieser Inhibit-Zeit versehen.

**Event Time**

Hier wird die Event Time für PDO 1 und 2 (Rx + Tx) dieses Knotens eingestellt.

**K-Bus-Update**

Berechnet die voraussichtliche Dauer für einen vollständigen Update des K-Busses (ist abhängig von Anzahl und Art der angeschlossenen Klemmen).

**Trans.Type**

Gibt den Transmission Type für digitale bzw. analoge Eingangstelegramme an. 254 + 255 entspricht der Ereignisgesteuerten Übertragung, 1...240 sind synchrone Übertragungsarten.

**Firmware Update**

Ermöglicht die Aktualisierung der Koppler-Firmware über die serielle Schnittstelle (erfordert Schnittstellenkabel des KS2000 Softwarepakets).

**Diagnose Eingänge**

FC510x: Jeder CANopen Feldbusknoten erhält ein Diagnose-Eingangsbyte (Node-State), das den Status des jeweiligen Slaves zur Laufzeit signalisiert und z.B. mit der SPS verknüpft werden kann. Zusätzlich wird über das Bit "Diag-Flag" signalisiert, ob eine neue Diagnose Information auf der Karte vorhanden ist. Diese kann dann über ADS READ gelesen werden.

**Karteireiter SDOs**

Obj. idx	Sub. idx	Length	Value (dec)	Value (hex)
<0x1400>	2	1	255	0xFF
<0x1401>	2	1	255	0xFF
<0x1401>	3	2	0	0x0
<0x1800>	2	1	255	0xFF
<0x1801>	2	1	255	0xFF
<0x1801>	3	2	0	0x0
<0x5500>	0	4	4294901760	0xFFFF0000
<0x6423>	0	1	1	0x1

Auf dieser Seite werden SDO Einträge angezeigt/verwaltet, die beim Startup zum Knoten geschickt werden. Einträge deren Objekt-Index in spitzen Klammern stehen, sind automatisch aufgrund der aktuellen Klemmenkonfiguration erzeugt worden. Weitere Einträge können über *Anfügen*, *Einfügen*, *Löschen* und *Bearbeiten* verwaltet werden.

**Karteireiter ADS**

Um SDO-Objekte auch zur Laufzeit schreiben und Lesen zu können (z.B. aus der SPS heraus), kann dem Knoten (Buskoppler) ein ADS-Port zugewiesen werden (CIFx0-CAN). Die FC510x verfügt stets über einen ADS-Port für jeden Knoten, da die Diagnoseinformationen über ADS transportiert werden. Über diesen können SDO-Objekte per ADS Read Request bzw. Write Request gelesen und geschrieben werden.

Der ADS IndexGroup beinhaltet den CANopen Object Index und der ADS IndexOffset beinhaltet den CANopen SubIndex.

**CANopen-Emergency-Objekt**

Einige CANopen-Statusdaten sowie empfangene Emergency-Objekte eines Nodes können per ADS von jedem TwinCAT Programm gelesen bzw. an jedes TwinCAT-Programm gemeldet werden. Die Datenstrukturen und Adressen unterscheiden sich zwischen FC510x und CIFx0-CAN.

**Hinweis**

Nähere Informationen zur Konfiguration der CANopen Busknoten und Masterkarten unter TwinCAT entnehmen Sie bitte der TwinCAT Dokumentation bzw. dem Handbuch der entsprechenden Masterkarte.

---

#### 4.5.4 Konfiguration mit Fremdsteuerungen

CANopen-Schnittstellen gibt es für eine Vielzahl von speicherprogrammierbaren Steuerungen (SPS), embedded Steuerungen und Industrie-PCs. Die Bandbreite der Konfigurationswerkzeuge für diese CANopen-Schnittstellen ist groß: sie reicht von der einfachen "CAN-Schicht2-Schnittstelle", bei der der Anwender jedes einzelne CAN Objekt einrichten muss und damit CANopen quasi nachbilden muss, bis zum komfortablem Konfigurationstool mit "Drag-and-Drop"-Funktionalität.

Im vorliegenden Handbuch sind alle benötigten CAN Objekte bewusst bis hinunter zur Bit-Darstellung auf dem CAN-Bus beschrieben. Damit können die Beckhoff CANopen-Geräte auch direkt von einer einfachen CAN Schnittstelle aus angesprochen werden. Hier mag insbesondere das Kapitel [Schnelleinstieg für erfahrene Anwender \[►\\_134\]](#) hilfreich sein.

Für die Konfiguration mit allgemeinen CANopen Konfigurationstools stehen die [eds-Dateien \[►\\_99\]](#) zum Download zur Verfügung. Meist genügt es bei diesen Tools das Default Mapping der Ein-/Ausgabebaugruppen nachzubilden.

Für nähere Angaben zur Konfiguration muss auf die Handbücher der jeweiligen Tool-Hersteller verwiesen werden.

## 4.6 Konfiguration der komplexen E/A Module

### 4.6.1 Konfigurations-Software KS2000

Die Konfigurations-Software KS2000 ermöglicht die Projektierung, Inbetriebnahme und Parametrierung von Feldbuskopplern und den dazugehörigen Busklemmen sowie der Feldbus Box Module. Die Verbindung zwischen Feldbuskoppler/Feldbus Box und PC wird über ein serielles Konfigurationskabel oder über den Feldbus hergestellt.



#### Projektierung

Sie können mit der Konfigurations-Software KS2000 die Feldbusstationen offline projektieren, d. h. vor der Inbetriebnahme den Aufbau der Feldbusstation mit sämtlichen Einstellungen der Buskoppler und Busklemmen bzw. der Feldbus Box Module vorbereiten. Diese Konfiguration kann später in der Inbetriebnahmephase per Download an die Feldbusstation übertragen werden. Zur Dokumentation wird Ihnen der Aufbau der Feldbusstation, eine Stückliste der verwendeten Feldbus-Komponenten, eine Liste der von Ihnen geänderten Parameter etc. aufbereitet. Bereits existierende Feldbusstationen stehen nach einem Upload zur weiteren Bearbeitung zur Verfügung.

#### Parametrierung

KS2000 bietet auf einfache Art den Zugriff auf die Parameter einer Feldbusstation: Für sämtliche Buskoppler und alle intelligenten Busklemmen sowie Feldbus Box Module stehen spezifische Dialoge zur Verfügung, mit deren Hilfe die Einstellungen leicht modifiziert werden können. Alternativ haben Sie vollen Zugriff auf sämtliche internen Register. Die Bedeutung der Register entnehmen Sie bitte der Registerbeschreibung.

**Inbetriebnahme**

KS2000 erleichtert die Inbetriebnahme von Maschinenteilen bzw. deren Feldbusstationen: Projektierte Einstellungen können per Download auf die Feldbus-Module übertragen werden. Nach dem *Login* auf die Feldbusstation besteht die Möglichkeit, Einstellungen an Koppler, Klemmen und Feldbus Box Modulen direkt *online* vorzunehmen. Dazu stehen die gleichen Dialoge und der Registerzugriff wie in der Projektierungsphase zur Verfügung.

KS2000 bietet den Zugriff auf die Prozessabbilder von Buskoppler und Feldbus Box:

- Sie können per Monitoring das Ein- und Ausgangsabbild beobachten.
- Zur Inbetriebnahme der Ausgangsmodule können im Ausgangsprozessabbild Werte vorgegeben werden.

Sämtliche Möglichkeiten des Online-Modus können parallel zum eigentlichen Feldbus-Betrieb der Feldbusstation vorgenommen werden. Das Feldbus-Protokoll hat dabei natürlich stets die höhere Priorität.

**4.6.2 Parametrierung durch Register**

**4.6.2.1 Allgemeine Registerbeschreibung**

Bei den komplexen Modulen können verschiedene Betriebsarten bzw. Funktionalitäten eingestellt werden. Die *Allgemeine Registerbeschreibung* erläutert den Inhalt der Register, die bei allen komplexen Modulen identisch sind. Die modulspezifischen Register werden in dem darauffolgendem Kapitel erklärt. Der Zugriff auf die internen Register der Module wird im Kapitel *Register-Kommunikation* beschrieben.

**Allgemeine Registerbeschreibung**

Komplexe Module die einen Prozessor besitzen, sind in der Lage mit der übergeordneten Steuerung bidirektional Daten auszutauschen. Diese Module werden im folgenden als intelligente Module bezeichnet. Zu ihnen zählen die analogen Eingänge (0 bis 10 V, -10 bis 10 V, 0 bis 20 mA, 4 bis 20 mA), die analogen Ausgänge (0 bis 10 V, -10 bis 10 V, 0 bis 20 mA, 4 bis 20 mA), serielle Schnittstellenmodule (RS485, RS232, TTY, Datenaustausch-Module), Zähler-Module, Encoder-Interface, SSI-Interface, PWM-Module und alle anderen parametrierbare Module.

Alle intelligenten Modulen besitzen intern eine in ihren wesentlichen Eigenschaften identisch aufgebaute Datenstruktur. Dieser Datenbereich ist wortweise organisiert und umfasst 64 Speicherplätze. Über diese Struktur sind die wesentlichen Daten und Parameter der Module les- und einstellbar. Zusätzlich sind Funktionsaufrufe mit entsprechenden Parametern möglich. Jeder logische Kanal einer intelligenten Module besitzt eine solche Struktur (4-Kanal analoge Module besitzen also 4 Registersätze).

Diese Struktur gliedert sich in folgende Bereiche:

Bereich	Adresse
Prozessvariablen	0-7
Typ-Register	8-15
Hersteller- Parameter	16-30
Anwender- Parameter	31-47
Erweiterter Anwenderbereich	48-63

**Register R0-R7 (im internen RAM des Moduls)**

Die Prozessvariablen können ergänzend zum eigentlichen Prozessabbild genutzt werden und sind in ihrer Funktion Modulspezifisch.

**R0-R5**

Diese Register besitzen eine vom Modul-Typ abhängige Funktion.

**R6**

Diagnose-Register: Das Diagnose-Register kann zusätzliche Diagnose-Information enthalten. So werden z.B. bei seriellen Schnittstellenmodulen Paritäts-Fehler, die während der Datenübertragung aufgetreten sind, angezeigt.

**R7**

Kommandoregister

- High-Byte\_Write = Funktionsparameter
- Low-Byte\_Write = Funktionsnummer
- High-Byte\_Read = Funktionsergebnis
- Low-Byte\_Read = Funktionsnummer

**Register R8-R15 (im internen ROM des Moduls)**

Die Typ- und Systemparameter sind fest vom Hersteller programmiert und können vom Anwender nur gelesen und nicht verändert werden.

**R8**

Feldbus Box Typ: Der Feldbus Box Typ in Register R8 wird zur Identifizierung der Feldbus Box benötigt.

**R9**

Softwareversion x.y.: Die Software-Version kann als ASCII-Zeichenfolge gelesen werden.

**R10**

Datenlänge: R10 beinhaltet die Anzahl der gemultiplexten Schieberegister und deren Länge in Bit. Der Buskoppler sieht diese Struktur.

**R11**

Signalkanäle: Im Vergleich zu R10 steht hier die Anzahl der logisch vorhandenen Kanäle. So kann z.B. ein physikalisch vorhandenes Schieberegister durchaus aus mehreren Signalkanälen bestehen.

**R12**

Minimale Datenlänge: Das jeweilige Byte enthält die minimal zu übertragene Datenlänge eines Kanals. Ist das MSB gesetzt, so ist das Control/Status-Byte für die Funktion des Moduls nicht zwingend notwendig, und wird bei entsprechender Konfiguration des Buskopplers nicht zur Steuerung übertragen. Die Information steht

- bei einem Ausgangsmodul im High-Byte
- bei einem Eingangsmodul im Low-Byte.

**R13**

Datentypregister

Datentypregister	Beschreibung
0x00	Modul ohne gültigen Datentyp
0x01	Byte-Array
0x02	Struktur 1 Byte n Bytes
0x03	Word-Array
0x04	Struktur 1 Byte n Worte
0x05	Doppelwort-Array
0x06	Struktur 1 Byte n Doppelworte
0x07	Struktur 1 Byte 1 Wort
0x08	Struktur 1 Byte 1 Doppelwort
0x11	Byte-Array mit variabler logischer Kanallänge
0x12	Struktur 1 Byte n Bytes mit variabler logischer Kanallänge (z.B. 60xx)
0x13	Word-Array mit variabler logischer Kanallänge
0x14	Struktur 1 Byte n Worte mit variabler logischer Kanallänge
0x15	Doppelwort-Array mit variabler logischer Kanallänge
0x16	Struktur 1 Byte n Doppelworte mit variabler logischer Kanallänge

**R14**

reserviert

**R15**

Alignment-Bits (RAM): Mit den Alignment-Bits wird das Analogmodul im Prozessabbild auf eine Bytegrenze gelegt.

**Register R16-R30 (Hersteller-Parameter, serielles EEPROM)**

Die Hersteller-Parameter werden vom Hersteller für jeden Modultyp modulspezifisch festgelegt, können jedoch mit der Konfigurations-Software KS2000 oder über Registerkommunikation durch die Steuerung geändert werden. Die Hersteller-Parameter sind spannungsausfallsicher im seriellen EEPROM der Klemme gespeichert. Zu Änderung der Hersteller-Parameter müssen Sie zuvor in Register R31 ein Code-Wort setzen.

**Register R31-R47 (Anwendungs-Parameter, serielles EEPROM)**

Die Anwender-Parameter sind Modulspezifisch. Sie können mit der Konfigurations-Software KS2000 oder über Registerkommunikation durch die Steuerung geändert werden. Die Anwender-Parameter sind spannungsausfallsicher im seriellen EEPROM der Klemme gespeichert. Zu Änderung der Anwender-Parameter müssen Sie zuvor in Register R31 das Anwender-Code-Wort setzen.

**R31**

Code-Wort-Register im RAM: Damit Parameter im Anwender-Bereich geändert werden können muss hier das Code-Wort 0x1235 eingetragen werden. Wird ein abweichender Wert in dieses Register eingetragen, so wird der Schreibschutz gesetzt. Bei inaktivem Schreibschutz wird das Code-Wort beim Lesen des Register zurückgegeben, ist der Schreibschutz aktiv enthält das Register den Wert Null.

**R32**

Feature-Register: Dieses Register legt die Betriebsarten der Klemme fest. So kann z.B. bei den analogen E/A-Modulen eine anwenderspezifische Skalierung aktiviert werden.

### R33 bis R63

Klemmenspezifische Register: Diese Register sind vom Klemmentyp abhängig.

### Register R47 bis R63 (Registererweiterung für zusätzliche Funktionen)

Diese Register sind für zusätzliche Funktionen vorgesehen.

## 4.6.2.2 Registerkommunikation über SDO

### CANopen Default-Einstellung: kompakte Darstellung der analogen Prozessdaten

Bei den CANopen Baugruppen sind analoge Signale per Default kompakt und nicht komplex ins Prozessabbild gemappt. Auf diese Weise werden die PDOs optimal ausgenutzt - je 4 Kanäle belegen ein PDO. In der komplexen Darstellung werden für jeden Kanal 6 Byte Prozessabbildgröße benötigt: im Tx-PDO 1 Control-Byte sowie 2 Datenbytes sowie im RxPDO 1 Status-Byte sowie ebenfalls 2 Datenbytes.

Durch die kompakte Darstellung der Prozessdaten ist der Zugriff auf die Register allerdings nicht über die Prozessdaten möglich, da Control- und Status-Byte nicht übertragen werden. Bei CANopen erfolgt der Zugriff auf sämtliche Register der analogen Signalbaugruppen somit über die Service Daten Objekte (SDO).

### Sonderfunktionsbaugruppen: auch komplexe Darstellung

Bei Sonderfunktionsbaugruppen wie schnelle Zähler (IE/IP1502), PWM-Ausgänge (IE/IP25x2) oder Winkel-/ Wegmessbaugruppen (IE/IP5x09) werden das Control- und Status-Byte auch bei CANopen ins Prozessabbild gemappt. Bei diesen Baugruppen ist der Zugriff auf die Register wahlweise via SDO oder per Prozessdaten-Register-Kommunikation möglich.

### Zugriff auf Register des CANopen Moduls

Der Zugriff auf die Register der CANopen Anschaltung ("Koppler") erfolgt über das Objekt 0x4500. In diesem Registermodell befinden sich Parameter wie

- aktivieren Auto-Reset IP-Link
- Reaktion auf Ausfall IP-Link
- Einblenden von Diagnosedaten
- Identifikation der Signalmodule
- etc.

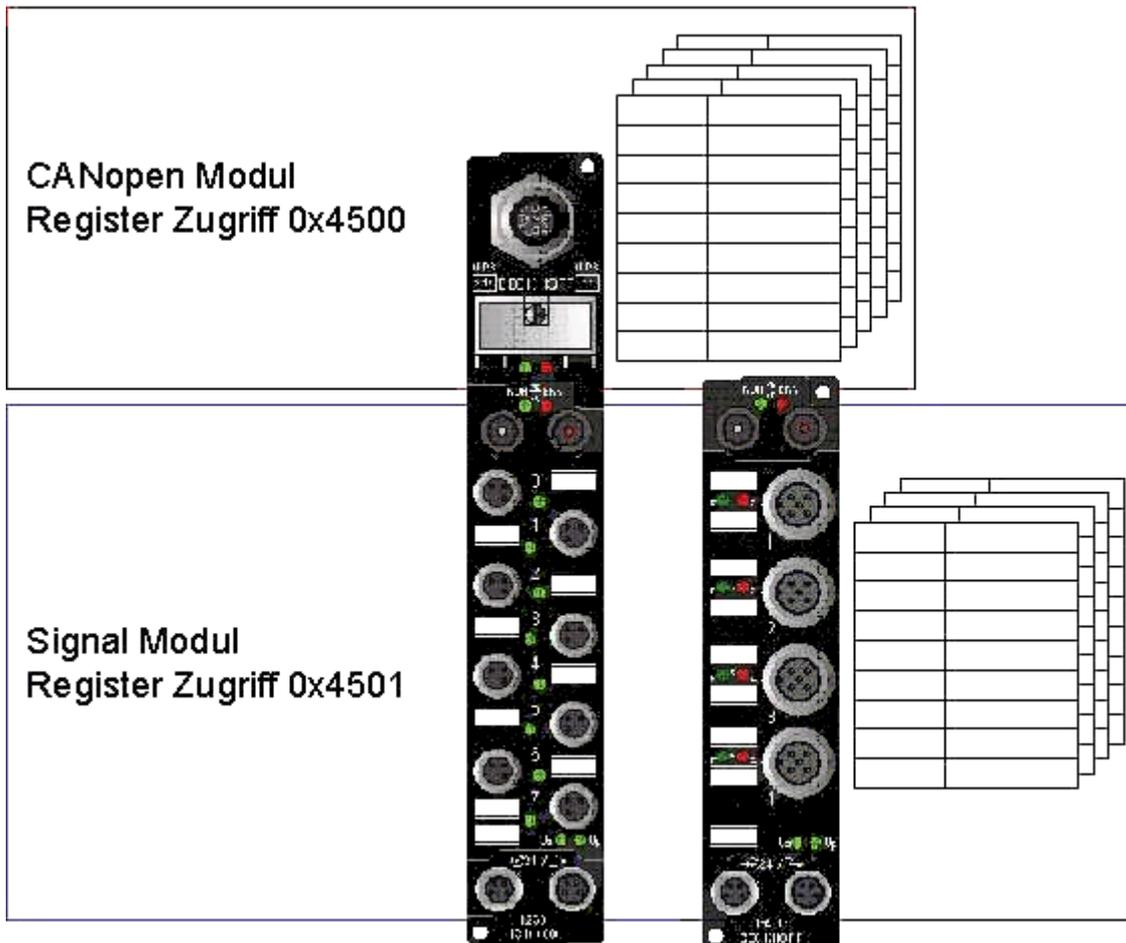
Details, Beispiele und die wichtigsten Einstellmöglichkeiten siehe Objektbeschreibung Register-Kommunikation Busknoten (0x4500).

### Zugriff auf Modul-Register

Der Zugriff auf die Register der intelligenten Signalmodule erfolgt über das Objekt 0x4501. Alle Signalmodule mit eigenem Prozessor verfügen über ein Registermodell, das signalspezifische Parameter enthält, z.B.

- Analogkanäle skalieren
- Sensortypen auswählen
- Filter einstellen
- etc.

Bei der Objektbeschreibung Register-Kommunikation Erweiterungsbox (0x4501) finden sich nähere Angaben und Beispiele.



### 4.6.2.3 Beispiel der Register-Kommunikation

(Beispiele zur Register-Kommunikation via SDO siehe Objektverzeichnis, Objekt 0x4501)

Wenn Bit 7 des Control-Bytes gesetzt wird, werden die ersten zwei Byte der Nutzdaten nicht zum Prozessdatenaustausch verwendet, sondern in den Registersatz der Klemme geschrieben oder daraus gelesen.

In Bit 6 des Control-Bytes legen Sie fest, ob ein Register gelesen oder beschrieben werden soll. Wenn das Bit 6 nicht gesetzt ist, wird ein Register gelesen, ohne es zu verändern. Der Wert kann dem Eingangs-Prozessabbild entnommen werden.

Wird das Bit 6 gesetzt, werden die Nutzdaten in ein Register geschrieben. Sobald das Status-Byte im Eingangs-Prozessabbild eine Quittung geliefert hat, ist der Vorgang abgeschlossen (siehe Beispiel). In die Bits 0 bis 5 des Control-Bytes wird die Adresse des anzusprechenden Registers eingetragen.

REG=1	W/R	A5	A4	A3	A2	A1	A0
-------	-----	----	----	----	----	----	----

REG: Registerbit  
 REG = 0: Prozessdaten  
 REG = 1: Register-Kommunikation

R/W: Zugriff auf Registerstruktur  
 W/R = 0: Register lesen  
 W/R = 1: Register schreiben

A5..A0: Registeradresse  
 A5...A0 sind insgesamt 64 Register adressierbar

**Beispiel 1**

Lesen des Registers 8 einer IP/IE1502. Die Baugruppe verfügt über zwei Kanäle, die sich mit jeweils 5 Byte ins Prozessabbild mappen.

Byte 0 (Control-Byte)	Byte 1 (Data In, D0)	Byte 2 (Data In, D1)	Byte 3 (Data In, D2)	Byte 5 (Data In, D3)
0x88	0xXX	0xXX	0xXX	0xXX

Bit 0.7 und Bit 0.3 ist gesetzt, bedeutet Register-Kommunikation aktiv, nur lesen da Bit 0.6 *low* ist und auslesen des Registers 8. Das Ausgangsdatenwort ist beim lesenden Zugriff ohne Bedeutung. Will man ein Register verändern, so schreibt man in das Ausgangswort die gewünschte Einstellung hinein.

Die Box liefert die folgende Typ-Bezeichnung zurück (0x05DE entspricht dem unsigned Integer 1502). Besonderheit bei den Feldbus Boxen:

Die letzte Ziffer bedeutet nicht den Anschluss (0 für S8, 1 für M 8 und 2 für M12) sondern gibt die Anzahl der Kanäle wieder.

Byte 0 (Status-Byte)	Byte 1 (Data In, D0)	Byte 2 (Data In, D1)	Byte 3 (Data In, D2)	Byte 5 (Data In, D3)
0x88	0x00	0x00	0x05	0xDE

**i Hinweis**

Um Register zu beschreiben muss das Passwort (0x1235) in Register 31 geschrieben werden, damit wird der Schreibschutz deaktiviert. Aktiviert wird dieser durch das Schreiben eines Wertes ungleich 0x1235. Beachten Sie das einige Einstellungen in den Registern erst nach erneutem Starten (Power-Restart) aktiviert werden.

**Beispiel 2**

Ablauf einer Register-Kommunikation zum Schreiben eines Registers

1. Schreiben des Register 31

Byte 0 (Control-Byte)	Byte 1 (Data In, High-Byte)	Byte 2 (Data In, Low-Byte)
0xDF	0x12	0x35

Antwort des Moduls/Busklemme

Byte 0 (Control-Byte)	Byte 1 (Data In, High-Byte)	Byte 2 (Data In, Low-Byte)
0x9F	0x00	0x00

2. Lesen des Register 31

Byte 0 (Control-Byte)	Byte 1 (Data In, High-Byte)	Byte 2 (Data In, Low-Byte)
0x9F	0xXX	0xXX

Antwort des Moduls/Busklemme

Byte 0 (Control-Byte)	Byte 1 (Data In, High-Byte)	Byte 2 (Data In, Low-Byte)
0x9F	0x12	0x35

3. Schreiben des Register 32

Byte 0 (Control-Byte)	Byte 1 (Data In, High-Byte)	Byte 2 (Data In, Low-Byte)
0xE0	0x00	0x02

Antwort des Moduls/Busklemme

Byte 0 (Control-Byte)	Byte 1 (Data In, High-Byte)	Byte 2 (Data In, Low-Byte)
0xA0	0x00	0x00

4. Lesen des Register 32

Byte 0 (Control-Byte)	Byte 1 (Data In, High-Byte)	Byte 2 (Data In, Low-Byte)
0xA0	0xFF	0xFF

Antwort des Moduls/Busklemme

Byte 0 (Control-Byte)	Byte 1 (Data In, High-Byte)	Byte 2 (Data In, Low-Byte)
0xA0	0x00	0x02

5. Schreiben des Register 31 Codewort zurücksetzen

Byte 0 (Control-Byte)	Byte 1 (Data In, High-Byte)	Byte 2 (Data In, Low-Byte)
0xDF	0x00	0x00

Antwort des Moduls/Busklemme

Byte 0 (Control-Byte)	Byte 1 (Data In, High-Byte)	Byte 2 (Data In, Low-Byte)
0x9F	0x00	0x00

# 5 Fehlerbehandlung und Diagnose

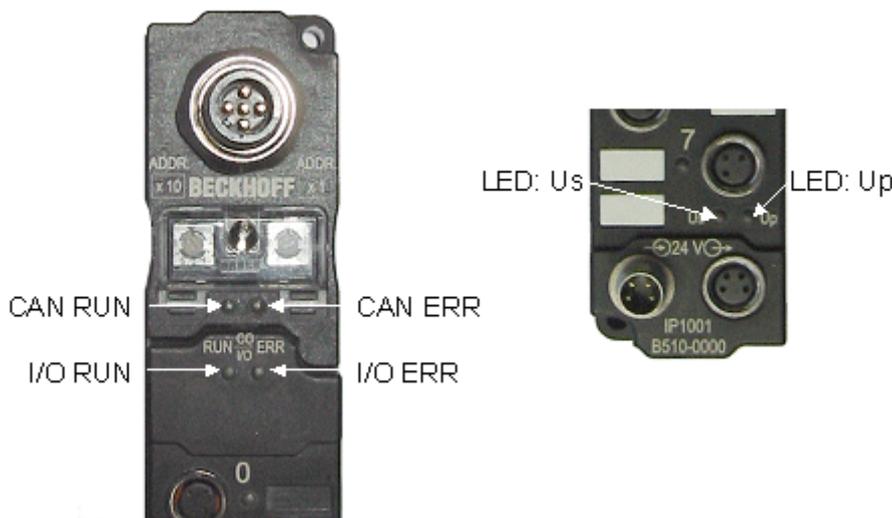
## 5.1 LEDs

### Übersicht

Zur Statusanzeige besitzt die CANopen Feldbus Box zwei Gruppen von LEDs. Die obere Gruppe (Feldbus-LEDs [[▶ 117](#)]) zeigt den Zustand des Feldbusses an.

Unter den Feldbusstatus LEDs befinden sich die zwei I/O LEDs (I/O RUN, I/O ERR). Diese dienen der Anzeige der Betriebszustände der lokalen und dezentralen I/Os und der Verbindung zu diesen via IP-Link.

Unten befinden sich zwei weitere grüne LEDs zur Anzeige der Versorgungsspannung. Die linke LED (Us) zeigt die 24V Versorgung der Feldbus Box an. Je nach Signalvariante wird aus dieser Versorgungsspannung auch die Sensorik gespeist (siehe I/O-Dokumentation). Die rechte LED (Up) signalisiert die Versorgung der Ausgänge.



### Feldbus-LEDs

#### Feldbus-LEDs

Die oberen beiden LEDs zeigen die Betriebszustände der CANopen-Kommunikation an. Dabei zeigt die CAN-ERR LED den Zustand der Busphysik sowie Protokollfehler. Die RUN LED zeigt den CANopen Status des Busknotens.

Das LED Verhalten entspricht der CAN in Automation CANopen-Empfehlung DRP303-3.

**CAN-ERR Blink-Code**

CAN-ERR	Bedeutung
<b>aus</b>	CAN Bus <b>fehlerfrei</b>
<b>schnelles Blitzen</b> (ca. 50ms an, ca. 50ms aus; im Wechsel mit RUN LED)	<b>AutoBaud-Raten-Erkennung</b> hat noch keine gültige Baud-Rate gefunden. Noch nicht genügend Telegramme auf dem Bus.
<b>1 x Blinken</b> (ca. 200ms an, 1s aus)	<b>CAN warning limit überschritten.</b> Es sind zu viele Error Frames auf dem Bus. Bitte Verdrahtung (z.B. Abschlusswiderstände, Schirmung, Leitungslänge, Stichleitungen) überprüfen. Weitere mögliche Ursache für Überschreitung des warning limits: kein weiterer Teilnehmer im Netz vorhanden (tritt z.B. beim ersten gestarteten Knoten auf).
<b>2 x Blinken</b> (jeweils ca. 200ms an, 200ms aus, gefolgt von 1s Pause)	Die <b>Guarding oder Heartbeat Überwachung</b> hat angesprochen, es werden keine Guarding bzw. Heartbeat-Telegramme mehr empfangen. Voraussetzung für Guarding Überwachung: Guard Time und Life Time Factor sind > 0 Voraussetzung für Heartbeat Überwachung: Consumer Heartbeat >0). Der Buskoppler ist Pre-Operational (PDOs abgeschaltet), die Ausgänge sind im Fehlerzustand.
<b>3 x Blinken</b> (jeweils ca. 200ms an, 200ms aus, gefolgt von 1s Pause)	Es ist ein <b>Synchronisations-Fehler</b> aufgetreten. Es wurden in der eingestellten Überwachungszeit (Objekt 0x1006 x 1,5) keine Sync. Telegramme empfangen. Der Busknoten ist Pre-Operational (PDOs abgeschaltet), die Ausgänge sind im Fehlerzustand.
<b>4 x Blinken</b> (jeweils ca. 200ms an, 200ms aus, gefolgt von 1s Pause)	<b>Event Timer Fehler:</b> Innerhalb der eingestellten Event Time (0x1400ff Subindex 5) hat der Buskoppler kein RxPDO empfangen. Der Busknoten ist Pre-Operational (PDOs abgeschaltet), die Ausgänge sind im Fehlerzustand.

**RUN Blink-Code**

RUN	Bedeutung
<b>aus</b>	Firmwarestand < C0: Busknoten ist im STOPPED Zustand. Keine Kommunikation mit SDO oder PDO möglich.
<b>schnelles Blitzen</b> (ca. 50ms an, ca. 50ms aus; im Wechsel mit CAN ERR LED)	<b>AutoBaud-Raten-Erkennung</b> hat noch keine gültige Baud-Rate gefunden. Noch nicht genügend Telegramme auf dem Bus.
<b>1 x Blinken</b> (ca. 200ms an, 1s aus)	Busknoten ist im <b>STOPPED</b> Zustand. Keine Kommunikation mit SDO oder PDO möglich.
<b>Wechselblinken</b> (jeweils ca. 200ms an, 200ms aus)	Busknoten ist im <b>PRE-OPERATIONAL</b> Zustand. Der Knoten wurde noch nicht gestartet.
<b>an</b>	Busknoten ist im <b>OPERATIONAL</b> Zustand.

**I/O LEDs**

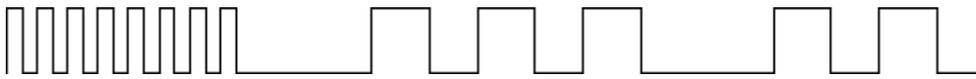
## I/O LEDs

Zwei LEDs, die I/O-LEDs, dienen der Anzeige der Betriebszustände der I/O und der Verbindung zu diesen via IP-Link. Die grüne LED (I/O RUN) leuchtet, um den fehlerfreien Betrieb anzuzeigen. Die rote LED (I/O ERR) blinkt zur Fehleranzeige mit zwei unterschiedlichen Frequenzen. Der Fehler wird in folgender Weise vom Blink-Code angezeigt:

1. Schnelles Blinken: Start des Fehler-Codes

2. erste langsame Sequenz: Fehler Code

3. zweite langsame Sequenz: Fehlerargument bzw. Ort des Fehlers



Start des Fehler-Codes

Fehler-Code

Fehlerargument

**Kompakt Box (ohne IP-Link Verbindung):**

Die I/O LEDs zeigen den Zustand der internen Kommunikation mit der Sensorplatine.

LED grün	LED rot*		Beschreibung	Abhilfe
aus	aus		kein lokaler Datenaustausch	CANopen-Kommunikationsfehler beseitigen und Kompakt Box in Pre-Operational oder Operational Zustand schalten.
aus	1	0	EEPROM - Prüfsummenfehler	Herstellereinstellung mit der KS2000 Software setzen
an	aus		Modul ist in Datenaustausch	kein Fehler

\* LED rot, linke Spalte: Fehler-Code, rechte Spalte Fehlerargument

**Koppler Box (mit IP-Link Verbindung)**

Die I/O LEDs zeigen den Zustand der IP-Link Verbindung an. IP-Link-Fehler sind meist auf unsachgemäßen Gebrauch der Lichtwellenleitung zurückzuführen.

LED grün	LED rot*		Beschreibung	Abhilfe
aus	aus		kein Datenaustausch	Modul im synchron Mode - Profibus zyklische Daten aktivieren
aus	1	0	EEPROM - Prüfsummenfehler	Herstellereinstellung mit der KS2000 Software setzen
aus	2		Reserve	-
aus	3	n	Bruchstelle wurde erkannt	n-tes Modul vor dem Empfänger des Masters
aus	4	n	zu viele Fehlerhafte Telegramme erkannt	vor dem n-ten Erweiterungsmodul ist die LWL-Verkabelung zu prüfen
aus	5	n	Registerzugriff auf komplexe Module gescheitert	n-tes Modul überprüfen
aus	11	n	Komplexes Modul arbeitet Fehlerhaft	n-tes Modul tauschen
aus	12	n	mehr als 120 Module im Ring	weniger Module anschließen
aus	13	n	n-tes Modul unbekannt	Firmware Update erforderlich
an	aus		Modul ist im Datenaustausch	kein Fehler

\* LED rot, linke Spalte: Fehler-Code, rechte Spalte Fehlerargument bzw. Ort des Fehlers.

Tab. 2: Erweiterungsbox

LED grün	LED rot	Beschreibung
aus	an	es werden kein Daten über IP-Link empfangen
aus	blinkt, flackert	es werden fehlerhafte IP-Link Protokolle empfangen (sehr schlechte Datenverbindung)
blinkt, flackert	blinkt, flackert	es werden fehlerhafte IP-Link Protokolle empfangen (schlechte Datenverbindung), muss noch nicht zum Fehler führen
blinkt, flackert, an	aus	es werden IP-Link Protokolle empfangen, kein Fehler

## 5.2 Diagnose-LEDs für lokale Fehler

### Lokale Fehler in einer Koppler Box (IL230x-Bxxx/Cxxx)

Unter den lokalen Fehlern ist gemeint, das ein Fehler in der Feldbus Box oder dem IP-Link aufgetreten ist. IP-Link-Fehler sind meist durch unsachgemäßen Gebrauch der Lichtwellenleitung zurück zu führen.

LED grün	LED rot		Beschreibung	Abhilfe
aus	aus		kein Datenaustausch	Modul im synchron Mode - zyklische Daten aktivieren
aus	1	0	EEPROM-Prüfsummenfehler	Herstellereinstellung setzen
aus	2		reserviert	-
aus	3	n	Bruchstelle wurde erkannt	n-tes Modul vor dem Empfänger des Masters
	3	n	m	(n*10)+m-tes Modul vor dem Empfänger des Masters
aus	4	n	zu viele fehlerhafte Telegramme erkannt (mehr als 25%)	vor dem n-ten Erweiterungsmodul (vor dem Empfänger des Masters) ist die LWL-Verkabelung zu prüfen
aus	5	n	Registerzugriff auf komplexe Module gescheitert	n-tes Modul überprüfen
aus	11	n	Komplexes Modul arbeitet fehlerhaft	n-tes Modul tauschen
aus	12	n	mehr als 120 Module im Ring	weniger Module anschließen
aus	13	n	n-tes Modul unbekannt	Firmware Update erforderlich
an	aus		Modul ist im Datenaustausch (kein Fehler)	-

**Lokale Fehler in einer Erweiterungsbox**

LED grün	LED rot	Beschreibung
aus	an	es werden kein Daten über den IP-Link empfangen
aus	blinkt, flackert	es werden fehlerhafte IP-Link Protokolle empfangen (sehr schlechte Datenverbindung)
blinkt, flackert	blinkt, flackert	es werden fehlerhafte IP-Link Protokolle empfangen (schlechte Datenverbindung), muss noch nicht zum Fehler führen
an	aus	es werden IP-Link Protokolle empfangen, kein Fehler

Fehlerhafte IP-Link Protokolle können entstehen durch:

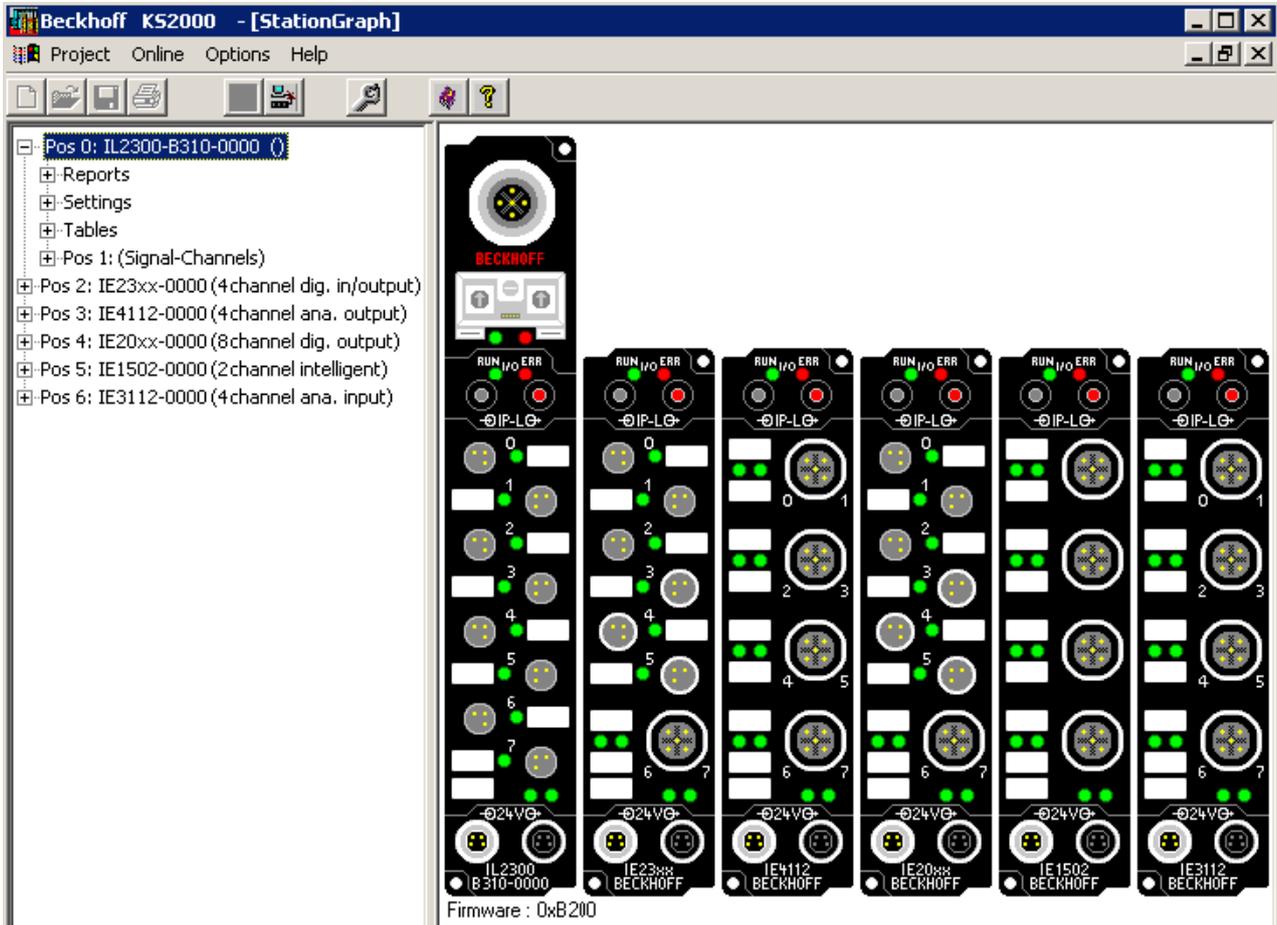
- schlecht konfektionierte IP-Link Steckverbinder
- IP-Link Leitung mit erhöhter Dämpfung durch z.B. Knick o.ä.
- defekte oder verschmutzte Sende LED (Modul vor dem fehlerhaften Modul)
- defekter oder verschmutzter Empfänger

Der interne IP-Link-Fehlerzähler [► 123] der Koppler Box kann mit der KS2000 Software ausgelesen werden.

## 5.3 Überprüfung der IP-Link-Verbindung

Ein IP-Link-Kabel mit auf beiden Seiten korrekt konfektionierten IP-Link-Steckern garantiert eine fehlerfrei Übertragung.

Eine nachträgliche Prüfung der Übertragungsqualität und eine Fehlersuche ist mit der Konfigurationssoftware KS2000 möglich.



Hierzu sollte die Steuerung (z.B. ein PC mit Profibus-Karte) am Feldbus des Koppler Box angeschlossen sein und diese zyklisch mit Daten versorgen, oder die Koppler Box sollte mit der KS2000 auf *freilaufend* geschaltet werden.

Als Ergebnis sollte die I/O RUN LED auf dem Kopplermodul hellgrün leuchten. Dies zeigt, dass ein Datenaustausch mit den angeschlossenen Erweiterungsbox Modulen stattfindet. Eine rot flackernde I/O ERR LED zeigt fehlerhafte IP-Link-Telegramme an! Telegramme werden wie bei auch jedem Feldbussystem im Fehlerfall wiederholt, so dass eine Übertragung der Daten gewährleistet ist.

### Fehlerzähler

In Tabelle 90, Offset 005 werden aufgetretene IP-Link-Fehler gezählt. Sporadisch auftretende Fehler bedeuten noch keine Probleme für die Kommunikation. Dieser Fehlerzähler wird nur durch ein Power ON/ OFF zurückgesetzt.

- Settings
- Tables
  - 000: Configuration Coupler
  - 009: Terminal typ (auto)
  - 087: Table 87
  - 088: Table 88
  - 090: Diagnostic coupler**
  - 091: Diagnostic processimage
  - 092: Diagnostic terminal channel 1
  - 093: Diagnostic terminal channel 2
  - 094: Diagnostic terminal channel 3
  - 095: Diagnostic terminal channel 4

<i>Register</i>				
	Offset	HEX	UINT	BIN
▶	000	0x0001	1	0000 0000 0000 0001
	001	0x0000	0	0000 0000 0000 0000
	002	0x0000	0	0000 0000 0000 0000
	003	0x0000	0	0000 0000 0000 0000
	004	0x0000	0	0000 0000 0000 0000
	005	0x002A	42	0000 0000 0010 1010
	006	0x0000	0	0000 0000 0000 0000
	007	0x0000	0	0000 0000 0000 0000
	008	0x0000	0	0000 0000 0000 0000
	009	0x0000	0	0000 0000 0000 0000

Sollten in kurzer Zeit viel Fehler auftreten, wird eine schwere Störung diagnostiziert und das Koppler-Modul meldet diesen Fehler. Dies ist am Offset 006 oder 007 zu erkennen. Die beiden Werte stehen dann auf einem Wert > 200 und die I/O ERR LEDs des Koppler-Moduls blinken mit dem entsprechenden Fehler-Code.

**Hinweis**

Die Konfigurations-Software KS2000 kommuniziert über einen seriellen Kanal mit der Koppler Box. Der Registerinhalt wird nicht andauernd aktualisiert, muss also manuell refreshed werden.

**Fehlerort**

Stellt das Koppler Modul einen Fehler fest, versucht es den Fehlerort aus dem Register der Erweiterungs-Box herauszulesen. Ist der Ring tatsächlich unterbrochen bzw. die Kommunikation stark gestört, ist dies nicht möglich. Dann wird nur der Ort des Bruchs dargestellt und zwar rückwärts vom Koppler gezählt (siehe IP-Link Fehlersuche).

Läuft die Kommunikation noch, kann in Tabelle 87 der Fehlerzähler je Erweiterungsmodul ausgelesen werden.

Hier bezieht sich der Offset auf die Position links im KS2000 Baum (siehe Grafik), d.h. in diesem Beispiel werden Fehler beim Offset 004 und 006 angezeigt.

Im IP-Link Aufbau ist der Fehler also bei der Übertragung zu Modul IE20xx und bei der Übertragung zu IE3112 zu suchen.

- Settings
- Tables
  - Pos 1: (Signal-Channels)
  - Pos 2: IE23xx-0000 (4 channel dig. in/output)
  - Pos 3: IE4112-0000 (4 channel ana. output)
  - Pos 4: IE20xx-0000 (8 channel dig. output)
  - Pos 5: IE1502-0000 (2 channel intelligent)
  - Pos 6: IE3112-0000 (4 channel ana. input)

<i>Register</i>				
	Offset	HEX	UINT	BIN
▶	000	0x0000	0	0000 0000 0000 0000
	001	0x0000	0	0000 0000 0000 0000
	002	0x0000	0	0000 0000 0000 0000
	003	0x0000	0	0000 0000 0000 0000
	004	0x0004	10	0000 0000 0000 1010
	005	0x0000	0	0000 0000 0000 0000
	006	0x0008	8	0000 0000 0000 1000
	007	0x0000	0	0000 0000 0000 0000
	008	0x0000	0	0000 0000 0000 0000

Der Fehler kann also liegen, an:

- dem Sender-Modul
- dem Empfänger-Modul
- dem Kabel oder
- den Steckern

Wird in Tabelle 90 ein Fehler angezeigt, aber in Tabelle 87 nicht, so ist die Fehlerursache in der Übertragungsstrecke zwischen dem letzten Erweiterungsmodul und dem Koppler zu suchen.

In fast allen Fällen sind Übertragungsfehler auf schlecht konfektionierte IP-Link-Stecker oder zu hohe Dämpfung im LWL-Kabel (durch scharfe Knicke o.ä.) zurückzuführen.

Tabelle 87 wird bei einer IP-Link-Unterbrechung nicht aktualisiert, da diese Werte direkt aus den Erweiterungs-Modulen kommen. Diese können dann über den IP-Link natürlich nicht mehr ausgelesen werden.

---

**i Hinweis**

Falls Sie eine Koppler-Box (z.B. IL2300-Bxxx, IL2301-Bxxx oder IL2302-Bxxx ) ganz ohne Erweiterungs-Box-Module (IExxxx) betreiben möchten, müssen Sie Sende- und Empfangs-Anschluss diese Koppler-Box über ein IP-Link-Kabel direkt miteinander verbinden! Hierfür eignet sich besonders der IP-Link-Verbindungsstecker ZK1020-0101-1000.

---

## 5.4 Emergency-Objekt

Um anderen Teilnehmern am CANopen-Bus interne Gerätefehler oder CAN-Busfehler mitteilen zu können verfügen der CANopen Buskoppler über das Emergency-Objekt. Es ist mit einer hohen Priorität versehen und liefert wertvolle Informationen über den Zustand des Gerätes und des Netzes.

### HINWEIS

#### Achtung

Es wird dringend empfohlen, die Emergency-Objekte auszuwerten - sie stellen eine wertvolle Informationsquelle dar.

#### Struktur der Emergency-Nachricht

Das Emergency-Objekt ist stets 8 Byte lang; es enthält zunächst den 2-Byte Error Code, dann das 1-Byte Error Register und schließlich den 5 Byte großen Additional Code. Dieser teilt sich in ein 2-Byte Bitfeld und ein 3-Byte Parameterfeld auf:

11-bit Identifier	8 Byte Nutzdaten							
0x80 (128 <sub>dez</sub> ) + Node-ID	EC0	EC1	EReg	Bitfeld0: Comm	Bitfeld1: DevErr	EMCY Trigger	Info0	Info1

*Tab. 3: Legende*

Parameter	Erläuterung	
EC0	Error Code Low-Byte. Nicht genutzt (stets null)	
EC1	Error Code High-Byte. 0x50 = Gerätefehler, 0x81 = Kommunikationsfehler, 0x00 = Error Reset	
EReg	Error Register. 0x81 = Gerätefehler, 0x91 = Kommunikationsfehler	
Bitfeld 0: Comm	Bitfeld Kommunikations-Fehler:	
	0x01	Guarding verzögert od. ausgefallen
	0x02	Sync verzögert oder ausgefallen
	0x04	Falsche PDO-Länge parametrier
	0x08	Timeout Event-Timer: RxPDO nicht rechtzeitig empfangen
	0x10	Receive Queue Overrun
	0x20	Transmit Queue Overrun
	0x40	CAN Bus OFF
	0x80	CAN Warning Limit überschritten
Bitfeld 1: DevErr	Bitfeld Geräte-Fehler:	
	0x01	Klemmenfehler
	0x02	K-Bus-Fehler / IP-Link Fehler
	0x03	-
	0x04	EEPROM-Fehler
	0x10	Nicht unterstützte Klemme gesteckt (BK5110, LC5100)
	0x80	Geänderte HW-Konfiguration.

Parameter	Erläuterung
EMCY Trigger	Das Byte <i>Emergency Trigger</i> enthält jeweils den Code des Fehlers, der das Emergency-Telegramm ausgelöst hat. Wenn ein Fehler beseitigt wird, so wird ein Emergency-Telegramm mit dem Error Code 0x0000 gesendet und der Emergency Trigger enthält die Beschreibung des Fehlers, der beseitigt wurde. Dabei werden noch aktuelle Fehler in den Bitfeldern signalisiert. Nachdem der Buskoppler fehlerfrei ist, sendet er ein Emergency-Telegramm, das außer im Emergency Trigger nur Nullen enthält.
0x01	CAN Warning Limit überschritten (zu viele Error Frames)
0x02	CAN Bus OFF Zustand wurde erreicht. Da der Koppler dann kein Emergency mehr senden kann, wird nach Verlassen des Bus-Off Zustandes (erneute CAN Controller Initialisierung) ein Emergency mit Trigger 0x40 gesendet
0x03	Transmit Queue Overrun Sendepuffer Überlauf: CAN Nachrichten gingen verloren
0x04	Receive Queue Overrun: Empfangs-Puffer Überlauf: CAN Nachrichten gingen verloren
0x06	Falsche PDO-Länge parametrier (Mapping überprüfen). Info 0: parametrierte (erwartete) PDO-Länge in Bytes Info 1: aktuelle PDO-Länge (ergibt sich aus den addierten Längen der gemappten Objekte)
0x07	Sync verzögert (Time-Out nach Communication Cycle Period, Index 0x1006) oder ausgefallen
0x08	Guarding bzw. Heartbeat verzögert (Time-Out nach Guard-Time x Life Time Factor bzw. nach Consumer Heartbeat Time) oder ausgefallen
0x09	Geänderte HW-Konfiguration. Seit dem letzten Abspeichern wurde die Klemmenbestückung bzw. Zusammensetzung der Erweiterungsmodule geändert.
0x0A	Timeout Event Timer : RxPDO nicht rechtzeitig empfangen
0x0B	logischer" Tx-Queue Überlauf: SYNC Intervall zu kurz. Der Koppler konnte vor dem nächsten SYNC Telegramm nicht alle TxPDOs absetzen. Die TxPDOs werden dann z.B. jedes zweite SYNC Intervall abgesetzt. Abhilfe: SYNC Intervall verlängern oder Transmission Type erhöhen. Ggf. I/O Anzahl an dieser Busstation reduzieren (z.B. I/Os zur Nachbarstation verschieben)
0x0C	Nicht unterstützte Klemme gesteckt (BK5110 oder LC5100). Info 1: Klemmennummer (1...64)
0x0E	EEPROM-Fehler; Beim Speichern der Konfiguration in das EEPROM ist ein Fehler aufgetreten
0x0F	K-Bus-Fehler Info 0: Fehlerart

Parameter	Erläuterung
Info 0, Info 1	Enthält zusätzliche Fehler-Info; Bedeutung hängt von Emergency Trigger ab (siehe oben)

### Beispiel zum Emergency-Verhalten

- Der CAN-Fehlerzähler eines Buskopplers überschreitet das Warning Limit (zu viele Error Frames). Er sendet ein Emergency-Telegramm mit dem Identifier 0x80 + Knotenadresse (Default-Einstellung) mit folgendem Inhalt:  
00 81 91 80 00 01 00 00

Die ersten drei Bytes (0x00 81 91) kennzeichnen einen Kommunikationsfehler, das Bitfeld0 (0x80) signalisiert *CAN Warning Limit überschritten*. Der EMCY Trigger (0x01) zeigt, dass das Emergency wegen des Überschreitens des Warning Limits ausgelöst wurde.
- Gleich anschließend tritt am zweiten Kanal der 4-20 mA Analogeingangsklemme, die an der zehnten Stelle gesteckt ist, ein Kabelbruch auf. Der Buskoppler sendet ein erneutes Emergency-Telegramm mit folgendem Inhalt:  
00 50 91 80 01 10 0A 82

Die ersten beiden Bytes (0x00 50) zeigen einen Hardware Fehler, im Error Register (0x91) sind die Bits 0 (Generic Error), 4 (Communication) und 7 (Manufacturer Specific) gesetzt. Im Bitfeld0 (0x80) ist Bit7 gesetzt, das CAN Warning Limit ist weiterhin überschritten. Im Bitfeld1 (0x01) ist Bit 0 gesetzt, also Klemmenfehler. Der EMCY Trigger (0x10) zeigt, dass dieser Klemmenfehler das Emergency-Telegramm ausgelöst hat. Schließlich zeigt Info0 (0x0A) die Klemmennummer (10) und Info1 (0x82) mit Bit1 und Bit7, dass Kanal2 einen Fehler hat.
- Wenn nun der Fehlerzähler das Warning Limit wieder unterschreitet, sendet der Koppler das folgende Emergency-Telegramm:  
00 00 81 00 01 01 0A 82

Der Error Code (00 00) der ersten beiden Bytes zeigt an, dass ein Fehler zurückgesetzt wurde. Das Error Register (0x81) zeigt weiterhin einen Gerätefehler, da der Kabelbruch ja weiterhin aktuell ist. Das Bitfeld0 (0x00) zeigt, dass der Kommunikationsfehler nicht mehr ansteht. Nach Bitfeld1 (0x01) ist der Klemmenfehler weiterhin vorhanden. Der EMCY Trigger (0x01) gibt das Rücksetzen des CAN Warning Limits als Sendegrund an, in Info0 und Info1 wird weiterhin der Diagnose Status Code der Klemmen angezeigt.
- Nachdem der Kabelbruch behoben wurde wird auch dieser Fehler zurückgesetzt und der Koppler sendet folgendes Emergency-Telegramm:  
00 00 00 00 00 00 00 00

## 5.5 CANopen 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*.



#### Error Frames

Überschrittenes Warning Limit, Error Passive oder Bus-Off Zustand werden zunächst bei demjenigen 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 PDOs auslösen), so werden auch seine Telegramme mit großer Wahrscheinlichkeit zunächst gestört - entsprechend erreicht sein Fehlerzähler als erster kritische Zustände.

---

### Node-ID / Baud Rate Einstellung

Es muss sorgfältig darauf geachtet werden, dass keine Knotenadresse doppelt vergeben ist: für jedes CAN-Datentelegramm darf es nur einen Sender geben.

#### Test 1

Knotenadressen überprüfen. Falls die CAN Kommunikation wenigstens zeitweise funktioniert und alle Geräte die Boot-Up-Nachricht unterstützen, so kann die Adressvergabe auch durch Aufzeichnen der Boot-Up-Nachrichten nach dem Einschalten der Geräte überprüft werden - hierdurch wird aber kein Vertauschen von Knotenadressen erkannt.

#### Test 2

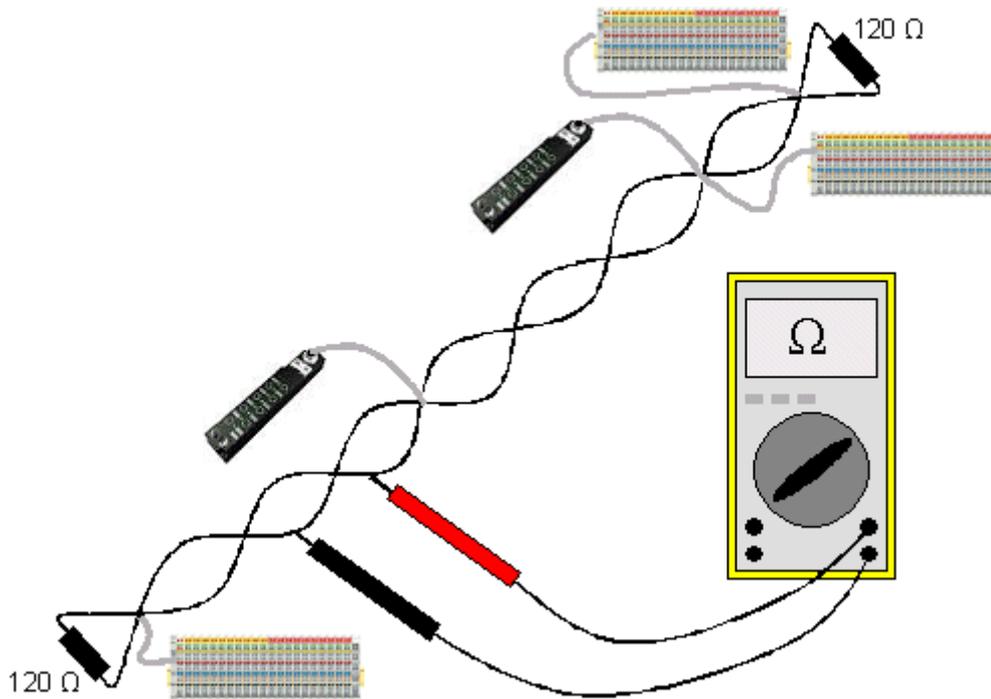
Überprüfen, ob überall die gleiche Baud-Rate eingestellt ist. Bei Sondergeräten: Wenn Bittiming Parameter zugänglich, stimmen diese mit den CANopen-Definitionen überein (Abtastzeitpunkt, SJW, Oszillator).

### Test der 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.

**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 (Baud-Ratenabhängig) 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 Amp 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 Stickleitungsenden 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**

Am Besten funktioniert in der Regel der "Low-tech-Ansatz": Teile des Netzes abhängen und beobachten, wann der Fehler verschwindet.

Aber: Dies funktioniert nicht gut 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 CANopen-Implementierung, unglückliches Timing im Boot-Up etc.) Ursache von Störungen. Hier ist dann ein Mitschrieb (Trace) des Busverkehrs mit anschließender Auswertung durch CANopen Experten erforderlich - das Beckhoff Support Team kann hier helfen.

Für solch einen Trace eignet sich 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.

Protokollprobleme lassen sich vermeiden, indem auf den Einsatz von Geräten verzichtet wird, die nicht Conformance getestet sind. Der offizielle CANopen Conformance Test und das entsprechende Zertifikat sind beim CAN in Automation Verband (<http://www.can-cia.de>) erhältlich.

## 6 Anhang

### 6.1 Schnelleinstieg für erfahrene Anwender

#### Zielgruppe

Diese Kurz-Übersicht wendet sich an Anwender, die CAN bereits kennen. Sie zeigt, welche CAN-Nachrichten erforderlich sind, um mit BECKHOFF CANopen Ein-/Ausgabebaugruppen in der Ausgangskonfiguration (mit Default-Einstellungen) zu arbeiten.

In jedem Fall ist die ausführliche Dokumentation zu lesen und zu berücksichtigen!

#### Hardware-Konfiguration

Über den DIP-Schalter sind auf den Buskopplern eine einheitliche Übertragungsrate sowie jeweils unterschiedliche Knotenadressen (Node-ID) einzustellen. Die Belegung des Schalters ist auf den Baugruppen aufgedruckt. Dabei ist zu beachten, dass CANopen die Adresse "0" zum Ansprechen aller Baugruppen (Broadcast) verwendet, diese also nicht auf der Baugruppe eingestellt werden darf.

#### Start der Baugruppen

CANopen ermöglicht es, die Baugruppen mit einem einzigen Netzwerkmanagement-Telegramm zu starten:

11-bit Identifier	2 Byte Nutzdaten								
0x00	0x01	0x00							

Dabei enthält das erste Datenbyte das Start-Kommando (Start\_Remote\_Node), das zweite Datenbyte die Knotenadresse (hier: 0, damit werden alle Knoten angesprochen).

Nachdem die Baugruppen gestartet wurden, sind die Ein- und Ausgänge freigeschaltet. In der Grundkonfiguration (Default-Einstellung) kommunizieren die Baugruppen im ereignisgesteuerten Modus, d.h. Änderungen der digitalen Eingänge werden sofort übermittelt, nach dem Empfang von Telegrammen mit Ausgangsdaten werden die Ausgänge sofort entsprechend gesetzt

#### CAN-Identifizier

Die CAN-Identifizier für die Ein- und Ausgangsdaten werden aus der Knotenadresse (1-63) abgeleitet:

Datenart	Default CAN-Identifizier
digitale Eingänge 1...64	0x180 (=384 <sub>dez</sub> ) + Knotenadresse
digitale Ausgänge 1...64	0x200 (=512 <sub>dez</sub> ) + Knotenadresse
analoge Eingänge 1...4	0x280 (=640 <sub>dez</sub> ) + Knotenadresse
analoge Ausgänge 1...4	0x300 (=768 <sub>dez</sub> ) + Knotenadresse
analoge Eingänge 5...8*	0x380 (=896 <sub>dez</sub> ) + Knotenadresse
analoge Ausgänge 5...8*	0x400 (=1024 <sub>dez</sub> ) + Knotenadresse
analoge Eingänge 9...12*	0x480 (=1152 <sub>dez</sub> ) + Knotenadresse
analoge Ausgänge 9...12*	0x500 (=1280 <sub>dez</sub> ) + Knotenadresse

\* falls mehr als 64 digitale Ein- bzw. Ausgänge vorhanden sind verschiebt sich der Bereich entsprechend (siehe Abschnitt Default Mapping).

#### Digitale Eingänge

Die CAN-Nachrichten mit digitalen Eingangsdaten stellen sich wie folgt dar:

11-bit Identifier	1-8 Byte Nutzdaten (je nach Zahl der Eingangsklemmen bzw. Erweiterungsbaugruppen)							
0x180(=384 <sub>dez</sub> ) + Node-ID	E0	E1	E2	E3	E4	E5	E6	E7

E0: Eingangsbytes auf Eingangsklemmen (bzw. Feldbus Box Baugruppen), von links nach rechts.

**Digitale Ausgänge**

Die CAN-Nachrichten mit digitalen Ausgangsdaten haben folgende Struktur:

11-bit Identifier	1-8 Byte Nutzdaten (je nach Zahl der Ausgangsklemmen bzw. Erweiterungsbaugruppen)							
0x200(=512 <sub>dez</sub> ) + Node-ID	A0	A1	A2	A3	A4	A5	A6	A7

A0: Ausgangsbytes auf Ausgangsklemmen (bzw. Feldbus Box Baugruppen), von links nach rechts.

**Analoge Eingänge**

So sehen CAN-Nachrichten mit analogen Eingangsdaten aus:

11-bit Identifier	4-8 Byte Nutzdaten (je nach Zahl der analogen Eingänge)							
0x280(640 <sub>dez</sub> ) + Node-ID	E0.0	E0.1	E1.0	E1.1	E2.0	E2.1	E3.0	E3.1

E x.0...E x.1: Analogeingang x. Die detaillierte Beschreibung des Datenformats findet sich im Objektverzeichnis bei Objekt 0x6401.

**Sendeverhalten Analogeingänge**

Um ein "Überfluten" des Busses mit sich ständig ändernden Analogeingangswerten zu vermeiden, erzeugen die Analogeingangs-Baugruppen nach CANopen im Defaultzustand keine Datentelegramme. Die Analogdaten lassen sich über Remote-Zugriff (Remote Transmit Request, CAN-Nachricht ohne Daten mit gesetztem RTR-Bit) auf die Analogeingangs-Telegramme auslesen. Alternativ kann selbstverständlich auch die Baugruppe so umkonfiguriert werden, dass eine Änderung des Eingangswerts zum Absetzen eines Datentelegramms führt. Hierzu wird ein Wert > 0 auf Index 0x6423 des Objektverzeichnisses geschrieben. Das entsprechende SDO-Telegramm sieht wie folgt aus:

11-bit Identifier	8 Byte Nutzdaten							
0x600(=768 <sub>dez</sub> ) + Node-ID	0x22	0x23	0x64	0x00	0x01	0x00	0x00	0x00

Es empfiehlt sich, analoge Eingangsdaten statt per Ereignissteuerung (jede Änderung des LSB ist ein Ereignis und führt zum Versenden des entsprechenden Telegramms) zyklisch synchron oder per event timer zu verschicken. Falls die Ereignissteuerung beibehalten werden soll, so sollte die Datenmenge durch Setzen eines Delta-Wertes (Objektverzeichnis Index 0x6426), von Grenzwerten (0x6424 +0x6425) oder einer Inhibit-Time (erneute Datenübertragung frühestens nach Ablauf der Inhibit-Time, 0x1801ff) reduziert werden. Details zur Parameterkommunikation finden sich im Kapitel Servicedaten: SDO ▶ 39].

**Analoge Ausgänge**

So sehen CAN-Nachrichten mit analogen Ausgangsdaten aus:

11-bit Identifier	4-8 Byte Nutzdaten (je nach Zahl der analogen Ausgänge)							
0x300(=768 <sub>dez</sub> ) + Node-ID	A0.0	A0.1	A1.0	A1.1	A2.0	A2.1	A3.0	A3.1

A x.0...A x.1: Analogausgang x. Die detaillierte Beschreibung des Datenformats findet sich im Objektverzeichnis bei Objekt 0x6411.

### Default Identifier

Eine tabellarische Übersicht über alle Default-Identifier findet sich im Anhang. Mit Hilfe dieser Übersicht lassen sich die von einem CAN-Monitor angezeigten CAN-Nachrichten schnell und einfach identifizieren.

### Stop der Baugruppen

Falls erforderlich kann die Prozessdaten-Kommunikation der Baugruppen mit folgendem Telegramm gestoppt werden:

11-bit Identifier	2 Byte Nutzdaten							
0x00	0x80	0xYZ						

0xXX: Knotenadresse; 0xYZ=0x00 adressiert alle Baugruppen

### Guarding

Die oben beschriebenen Telegramme reichen für viele Anwendungen bereits aus. Da die Baugruppen per Default im ereignisgesteuerten Modus arbeiten (kein zyklischer Datenaustausch), wird ein Ausfall einer Baugruppe jedoch nicht unbedingt erkannt. Abhilfe schafft hier die Überwachung der Baugruppen durch zyklisches Abfragen des Status, das sogenannte Node-Guarding.

Hierzu wird zyklisch ein Statustelegamm über Remote-Transmit-Request (RTR) angefordert:

11-bit Identifier	keine Nutzdaten im Anforderungstelegamm (RTR)
0x700(=1792 <sub>dez</sub> ) + Node-ID	(RTR-Bit im Header gesetzt)

Die Baugruppen antworten mit einem Telegramm, das ein Status-Byte enthält.

11-bit Identifier	1 Byte Nutzdaten							
0x700(=1792 <sub>dez</sub> ) + Node-ID	0xYZ							

0xYZ: Status-Byte:

Bits 6...0 enthalten den Knoten-Status (0x7F=127:Pre-Operational, 0x05=Operational; 0x04= Stopped bzw. Prepared).

Bit 7 = Toggle Bit (kippt nach jedem Senden).

Damit der Buskoppler einen Ausfall des Netzwerk-Masters erkennt (Watchdog-Funktion), müssen noch die Guard-Time (Objekt 0x100C) und der Life-Time-Factor (Objekt 0x100D) auf Werte ungleich 0 gesetzt werden. (Reaktionszeit bei Ausfall: Guard-Time x Life Time Factor).

### Heartbeat

Alternativ zum Guarding kann die Baugruppe auch über den sogenannten Heartbeat überwacht werden. Hierbei setzt der Knoten zyklisch sein Statustelegamm (Heartbeat) ab. Datenanforderungstelegamm (Remote Frames) sind nicht erforderlich.

Um das Heartbeat-Telegramm zu aktivieren muss die Producer Heartbeat Time gesetzt werden. Dies erfolgt mit folgendem SDO [► 39]-Telegramm:

11-bit Identifier	8 Byte Nutzdaten							
0x600(=768 <sub>dez</sub> ) + Node-ID	0x22	0x17	0x10	0x00	0xcd	0xab	0x00	0x00

wobei 0xabcd die gewünschte Heartbeat-Zykluszeit in Millisekunden ist.

Mit den beschriebenen Telegrammen sind Sie nun in der Lage, die Baugruppen zu starten und zu stoppen, Eingänge zu lesen, Ausgänge zu schreiben und die Baugruppen zu überwachen. Versäumen Sie nicht, das Handbuch sorgfältig zu lesen. Nur so können Sie die vielfältigen Features der BECKHOFF CANopen Buskoppler wirklich nutzen.

## 6.2 CAN Identifier-Liste

Die hier aufgeführte Liste soll bei der Identifizierung und Zuordnung von CANopen Nachrichten helfen. Aufgeführt sind alle von der CANopen Default Identifier Verteilung zugeordneten Identifier, sowie die von BECKHOFF via Objekt 0x5500 vergebenen herstellerspezifischen Default Identifier (nur in Netzen mit Knotenadressen <64 zu verwenden).

In der \*chm-Ausgabe der Dokumentation dienen die folgenden Werte als Suchhilfe und "Einsprungpunkte" in die umfangreiche Identifier-Tabelle:

Dezimal: [400 \[▶ 138\]](#) [500 \[▶ 138\]](#) [600 \[▶ 138\]](#) [700 \[▶ 138\]](#) [800 \[▶ 138\]](#) [900 \[▶ 138\]](#) [1000 \[▶ 138\]](#) [1100 \[▶ 138\]](#) [1200 \[▶ 138\]](#) [1300 \[▶ 138\]](#) [1400 \[▶ 138\]](#) [1500 \[▶ 138\]](#) [1600 \[▶ 138\]](#) [1700 \[▶ 138\]](#) [1800 \[▶ 138\]](#) [1900 \[▶ 138\]](#)

Hexadezimal: [0x181 \[▶ 138\]](#) [0x1C1 \[▶ 138\]](#) [0x201 \[▶ 138\]](#) [0x301 \[▶ 138\]](#) [0x401 \[▶ 138\]](#) [0x501 \[▶ 138\]](#) [0x601 \[▶ 138\]](#) [0x701 \[▶ 138\]](#)

Die Identifier-Verteilung via Objekt 0x5500 folgt diesem Schema:

Objekt	resultierende COB-ID (dez)	resultierende COB-ID (hex)
<b>Emergency</b>	129 bis 191 [255]	0x81 bis 0xBF [0xFF]
<a href="#">TxPDO1 [▶ 138]</a>	385 bis 447 [511]	0x181 bis 0x1BF [0x1FF]
<a href="#">RxPDO1 [▶ 138]</a>	513 bis 575 [639]	0x201 bis 0x23F [0x27F]
<a href="#">TxPDO2 [▶ 138]</a>	641 bis 676 [767]	0x281 bis 0x2BF [0x2FF]
<a href="#">RxPDO2 [▶ 138]</a>	769 bis 831 [895]	0x301 bis 0x33F [0x37F]
<a href="#">TxPDO3 [▶ 138]</a>	897 bis 959 [1023]	0x381 bis 0x3BF [0x3FF]
<a href="#">RxPDO3 [▶ 138]</a>	1025 bis 1087 [1151]	0x401 bis 0x43F [0x47F]
<a href="#">TxPDO4 [▶ 138]</a>	1153 bis 1215 [1279]	0x481 bis 0x4BF [0x4FF]
<a href="#">RxPDO4 [▶ 138]</a>	1281 bis 1343 [1407]	0x501 bis 0x53F [0x57F]
<a href="#">TxPDO5 [▶ 138]</a>	1665 bis 1727	0x681 bis 0x6BF
<a href="#">RxPDO5 [▶ 138]</a>	1921 bis 1983	0x781 bis 0x7BF
<a href="#">TxPDO6 [▶ 138]</a>	449 bis 511	0x1C1 bis 0x1FF
<a href="#">RxPDO6 [▶ 138]</a>	577 bis 639	0x241 bis 0x27F
<a href="#">TxPDO7 [▶ 138]</a>	705 bis 767	0x2C1 bis 0x2FF
<a href="#">RxPDO7 [▶ 138]</a>	833 bis 895	0x341 bis 0x37F
<a href="#">TxPDO8 [▶ 138]</a>	961 bis 1023	0x3C1 bis 0x3FF
<a href="#">RxPDO8 [▶ 138]</a>	1089 bis 1151	0x441 bis 0x47F
<a href="#">TxPDO9 [▶ 138]</a>	1217 bis 1279	0x4C1 bis 0x4FF
<a href="#">RxPDO9 [▶ 138]</a>	1345 bis 1407	0x541 bis 0x57F
<a href="#">TxPDO10 [▶ 138]</a>	1473 bis 1535	0x5C1 bis 0x5FF
<a href="#">RxPDO10 [▶ 138]</a>	1601 bis 1663	0x641 bis 0x67F
<a href="#">TxPDO11 [▶ 138]</a>	1729 bis 1791	0x6C1 bis 0x6FF
<a href="#">RxPDO11 [▶ 138]</a>	1857 bis 1919	0x741 bis 0x77F
<b>SDO (Tx)</b>	1409 bis 1471 [1535]	0x581 bis 0x5BF [0x5FF]
<b>SDO (Rx)</b>	1537 bis 1599 [1663]	0x601 bis 0x63F [0x67F]
<a href="#">Guarding / Heartbeat / Bootup [▶ 138]</a>	1793 bis 1855 [1919]	0x701 bis 0x73F [0x77F]

### Identifierliste

Mit \* gekennzeichnete Identifier werden auf den Buskopplern nach Beschreiben von Index 0x5500 herstellerspezifisch vergeben.

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
0	0x00	NMT	874	0x36A	RxPDO7*, Nd.42	1430	0x596	SDO Tx Nd.22
128	0x80	SYNC	875	0x36B	RxPDO7*, Nd.43	1431	0x597	SDO Tx Nd.23
129	0x81	EMCY Nd.1	876	0x36C	RxPDO7*, Nd.44	1432	0x598	SDO Tx Nd.24
130	0x82	EMCY Nd.2	877	0x36D	RxPDO7*, Nd.45	1433	0x599	SDO Tx Nd.25
131	0x83	EMCY Nd.3	878	0x36E	RxPDO7*, Nd.46	1434	0x59A	SDO Tx Nd.26
132	0x84	EMCY Nd.4	879	0x36F	RxPDO7*, Nd.47	1435	0x59B	SDO Tx Nd.27
133	0x85	EMCY Nd.5	880	0x370	RxPDO7*, Nd.48	1436	0x59C	SDO Tx Nd.28
134	0x86	EMCY Nd.6	881	0x371	RxPDO7*, Nd.49	1437	0x59D	SDO Tx Nd.29
135	0x87	EMCY Nd.7	882	0x372	RxPDO7*, Nd.50	1438	0x59E	SDO Tx Nd.30
136	0x88	EMCY Nd.8	883	0x373	RxPDO7*, Nd.51	1439	0x59F	SDO Tx Nd.31
137	0x89	EMCY Nd.9	884	0x374	RxPDO7*, Nd.52	1440	0x5A0	SDO Tx Nd.32
138	0x8A	EMCY Nd.10	885	0x375	RxPDO7*, Nd.53	1441	0x5A1	SDO Tx Nd.33
139	0x8B	EMCY Nd.11	886	0x376	RxPDO7*, Nd.54	1442	0x5A2	SDO Tx Nd.34
140	0x8C	EMCY Nd.12	887	0x377	RxPDO7*, Nd.55	1443	0x5A3	SDO Tx Nd.35
141	0x8D	EMCY Nd.13	888	0x378	RxPDO7*, Nd.56	1444	0x5A4	SDO Tx Nd.36
142	0x8E	EMCY Nd.14	889	0x379	RxPDO7*, Nd.57	1445	0x5A5	SDO Tx Nd.37
143	0x8F	EMCY Nd.15	890	0x37A	RxPDO7*, Nd.58	1446	0x5A6	SDO Tx Nd.38
144	0x90	EMCY Nd.16	891	0x37B	RxPDO7*, Nd.59	1447	0x5A7	SDO Tx Nd.39
145	0x91	EMCY Nd.17	892	0x37C	RxPDO7*, Nd.60	1448	0x5A8	SDO Tx Nd.40
146	0x92	EMCY Nd.18	893	0x37D	RxPDO7*, Nd.61	1449	0x5A9	SDO Tx Nd.41
147	0x93	EMCY Nd.19	894	0x37E	RxPDO7*, Nd.62	1450	0x5AA	SDO Tx Nd.42
148	0x94	EMCY Nd.20	895	0x37F	RxPDO7*, Nd.63	1451	0x5AB	SDO Tx Nd.43
149	0x95	EMCY Nd.21	897	0x381	TxPDO3*, Nd.1	1452	0x5AC	SDO Tx Nd.44
150	0x96	EMCY Nd.22	898	0x382	TxPDO3*, Nd.2	1453	0x5AD	SDO Tx Nd.45
151	0x97	EMCY Nd.23	899	0x383	TxPDO3*, Nd.3	1454	0x5AE	SDO Tx Nd.46
152	0x98	EMCY Nd.24	900	0x384	TxPDO3*, Nd.4	1455	0x5AF	SDO Tx Nd.47

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
153	0x99	EMCY Nd.25	901	0x385	TxPDO3*, Nd.5	1456	0x5B0	SDO Tx Nd.48
154	0x9A	EMCY Nd.26	902	0x386	TxPDO3*, Nd.6	1457	0x5B1	SDO Tx Nd.49
155	0x9B	EMCY Nd.27	903	0x387	TxPDO3*, Nd.7	1458	0x5B2	SDO Tx Nd.50
156	0x9C	EMCY Nd.28	904	0x388	TxPDO3*, Nd.8	1459	0x5B3	SDO Tx Nd.51
157	0x9D	EMCY Nd.29	905	0x389	TxPDO3*, Nd.9	1460	0x5B4	SDO Tx Nd.52
158	0x9E	EMCY Nd.30	906	0x38A	TxPDO3*, Nd.10	1461	0x5B5	SDO Tx Nd.53
159	0x9F	EMCY Nd.31	907	0x38B	TxPDO3*, Nd.11	1462	0x5B6	SDO Tx Nd.54
160	0xA0	EMCY Nd.32	908	0x38C	TxPDO3*, Nd.12	1463	0x5B7	SDO Tx Nd.55
161	0xA1	EMCY Nd.33	909	0x38D	TxPDO3*, Nd.13	1464	0x5B8	SDO Tx Nd.56
162	0xA2	EMCY Nd.34	910	0x38E	TxPDO3*, Nd.14	1465	0x5B9	SDO Tx Nd.57
163	0xA3	EMCY Nd.35	911	0x38F	TxPDO3*, Nd.15	1466	0x5BA	SDO Tx Nd.58
164	0xA4	EMCY Nd.36	912	0x390	TxPDO3*, Nd.16	1467	0x5BB	SDO Tx Nd.59
165	0xA5	EMCY Nd.37	913	0x391	TxPDO3*, Nd.17	1468	0x5BC	SDO Tx Nd.60
166	0xA6	EMCY Nd.38	914	0x392	TxPDO3*, Nd.18	1469	0x5BD	SDO Tx Nd.61
167	0xA7	EMCY Nd.39	915	0x393	TxPDO3*, Nd.19	1470	0x5BE	SDO Tx Nd.62
168	0xA8	EMCY Nd.40	916	0x394	TxPDO3*, Nd.20	1471	0x5BF	SDO Tx Nd.63
169	0xA9	EMCY Nd.41	917	0x395	TxPDO3*, Nd.21	1473	0x5C1	TxPDO10 *, Nd.1
170	0xAA	EMCY Nd.42	918	0x396	TxPDO3*, Nd.22	1474	0x5C2	TxPDO10 *, Nd.2
171	0xAB	EMCY Nd.43	919	0x397	TxPDO3*, Nd.23	1475	0x5C3	TxPDO10 *, Nd.3
172	0xAC	EMCY Nd.44	920	0x398	TxPDO3*, Nd.24	1476	0x5C4	TxPDO10 *, Nd.4
173	0xAD	EMCY Nd.45	921	0x399	TxPDO3*, Nd.25	1477	0x5C5	TxPDO10 *, Nd.5
174	0xAE	EMCY Nd.46	922	0x39A	TxPDO3*, Nd.26	1478	0x5C6	TxPDO10 *, Nd.6
175	0xAF	EMCY Nd.47	923	0x39B	TxPDO3*, Nd.27	1479	0x5C7	TxPDO10 *, Nd.7
176	0xB0	EMCY Nd.48	924	0x39C	TxPDO3*, Nd.28	1480	0x5C8	TxPDO10 *, Nd.8
177	0xB1	EMCY Nd.49	925	0x39D	TxPDO3*, Nd.29	1481	0x5C9	TxPDO10 *, Nd.9
178	0xB2	EMCY Nd.50	926	0x39E	TxPDO3*, Nd.30	1482	0x5CA	TxPDO10 *, Nd.10

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
179	0xB3	EMCY Nd.51	927	0x39F	TxPDO3*, Nd.31	1483	0x5CB	TxPDO10*, Nd.11
180	0xB4	EMCY Nd.52	928	0x3A0	TxPDO3*, Nd.32	1484	0x5CC	TxPDO10*, Nd.12
181	0xB5	EMCY Nd.53	929	0x3A1	TxPDO3*, Nd.33	1485	0x5CD	TxPDO10*, Nd.13
182	0xB6	EMCY Nd.54	930	0x3A2	TxPDO3*, Nd.34	1486	0x5CE	TxPDO10*, Nd.14
183	0xB7	EMCY Nd.55	931	0x3A3	TxPDO3*, Nd.35	1487	0x5CF	TxPDO10*, Nd.15
184	0xB8	EMCY Nd.56	932	0x3A4	TxPDO3*, Nd.36	1488	0x5D0	TxPDO10*, Nd.16
185	0xB9	EMCY Nd.57	933	0x3A5	TxPDO3*, Nd.37	1489	0x5D1	TxPDO10*, Nd.17
186	0xBA	EMCY Nd.58	934	0x3A6	TxPDO3*, Nd.38	1490	0x5D2	TxPDO10*, Nd.18
187	0xBB	EMCY Nd.59	935	0x3A7	TxPDO3*, Nd.39	1491	0x5D3	TxPDO10*, Nd.19
188	0xBC	EMCY Nd.60	936	0x3A8	TxPDO3*, Nd.40	1492	0x5D4	TxPDO10*, Nd.20
189	0xBD	EMCY Nd.61	937	0x3A9	TxPDO3*, Nd.41	1493	0x5D5	TxPDO10*, Nd.21
190	0xBE	EMCY Nd.62	938	0x3AA	TxPDO3*, Nd.42	1494	0x5D6	TxPDO10*, Nd.22
191	0xBF	EMCY Nd.63	939	0x3AB	TxPDO3*, Nd.43	1495	0x5D7	TxPDO10*, Nd.23
385	0x181	TxPDO1, DI, Nd.1	940	0x3AC	TxPDO3*, Nd.44	1496	0x5D8	TxPDO10*, Nd.24
386	0x182	TxPDO1, DI, Nd.2	941	0x3AD	TxPDO3*, Nd.45	1497	0x5D9	TxPDO10*, Nd.25
387	0x183	TxPDO1, DI, Nd.3	942	0x3AE	TxPDO3*, Nd.46	1498	0x5DA	TxPDO10*, Nd.26
388	0x184	TxPDO1, DI, Nd.4	943	0x3AF	TxPDO3*, Nd.47	1499	0x5DB	TxPDO10*, Nd.27
389	0x185	TxPDO1, DI, Nd.5	944	0x3B0	TxPDO3*, Nd.48	1500	0x5DC	TxPDO10*, Nd.28
390	0x186	TxPDO1, DI, Nd.6	945	0x3B1	TxPDO3*, Nd.49	1501	0xDE	TxPDO10*, Nd.29
391	0x187	TxPDO1, DI, Nd.7	946	0x3B2	TxPDO3*, Nd.50	1502	0x5DE	TxPDO10*, Nd.30
392	0x188	TxPDO1, DI, Nd.8	947	0x3B3	TxPDO3*, Nd.51	1503	0x5DF	TxPDO10*, Nd.31
393	0x189	TxPDO1, DI, Nd.9	948	0x3B4	TxPDO3*, Nd.52	1504	0x5E0	TxPDO10*, Nd.32
394	0x18A	TxPDO1, DI, Nd.10	949	0x3B5	TxPDO3*, Nd.53	1505	0x5E1	TxPDO10*, Nd.33
395	0x18B	TxPDO1, DI, Nd.11	950	0x3B6	TxPDO3*, Nd.54	1506	0x5E2	TxPDO10*, Nd.34
396	0x18C	TxPDO1, DI, Nd.12	951	0x3B7	TxPDO3*, Nd.55	1507	0x5E3	TxPDO10*, Nd.35
397	0x18D	TxPDO1, DI, Nd.13	952	0x3B8	TxPDO3*, Nd.56	1508	0x5E4	TxPDO10*, Nd.36

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
398	0x18E	TxPDO1, DI, Nd.14	953	0x3B9	TxPDO3*, Nd.57	1509	0x5E5	TxPDO10*, Nd.37
399	0x18F	TxPDO1, DI, Nd.15	954	0x3BA	TxPDO3*, Nd.58	1510	0x5E6	TxPDO10*, Nd.38
400	0x190	TxPDO1, DI, Nd.16	955	0x3BB	TxPDO3*, Nd.59	1511	0x5E7	TxPDO10*, Nd.39
401	0x191	TxPDO1, DI, Nd.17	956	0x3BC	TxPDO3*, Nd.60	1512	0x5E8	TxPDO10*, Nd.40
402	0x192	TxPDO1, DI, Nd.18	957	0x3BD	TxPDO3*, Nd.61	1513	0x5E9	TxPDO10*, Nd.41
403	0x193	TxPDO1, DI, Nd.19	958	0x3BE	TxPDO3*, Nd.62	1514	0x5EA	TxPDO10*, Nd.42
404	0x194	TxPDO1, DI, Nd.20	959	0x3BF	TxPDO3*, Nd.63	1515	0x5EB	TxPDO10*, Nd.43
405	0x195	TxPDO1, DI, Nd.21	961	0x3C1	TxPDO8*, Nd.1	1516	0x5EC	TxPDO10*, Nd.44
406	0x196	TxPDO1, DI, Nd.22	962	0x3C2	TxPDO8*, Nd.2	1517	0x5ED	TxPDO10*, Nd.45
407	0x197	TxPDO1, DI, Nd.23	963	0x3C3	TxPDO8*, Nd.3	1518	0x5EE	TxPDO10*, Nd.46
408	0x198	TxPDO1, DI, Nd.24	964	0x3C4	TxPDO8*, Nd.4	1519	0x5EF	TxPDO10*, Nd.47
409	0x199	TxPDO1, DI, Nd.25	965	0x3C5	TxPDO8*, Nd.5	1520	0x5F0	TxPDO10*, Nd.48
410	0x19A	TxPDO1, DI, Nd.26	966	0x3C6	TxPDO8*, Nd.6	1521	0x5F1	TxPDO10*, Nd.49
411	0x19B	TxPDO1, DI, Nd.27	967	0x3C7	TxPDO8*, Nd.7	1522	0x5F2	TxPDO10*, Nd.50
412	0x19C	TxPDO1, DI, Nd.28	968	0x3C8	TxPDO8*, Nd.8	1523	0x5F3	TxPDO10*, Nd.51
413	0x19D	TxPDO1, DI, Nd.29	969	0x3C9	TxPDO8*, Nd.9	1524	0x5F4	TxPDO10*, Nd.52
414	0x19E	TxPDO1, DI, Nd.30	970	0x3CA	TxPDO8*, Nd.10	1525	0x5F5	TxPDO10*, Nd.53
415	0x19F	TxPDO1, DI, Nd.31	971	0x3CB	TxPDO8*, Nd.11	1526	0x5F6	TxPDO10*, Nd.54
416	0x1A0	TxPDO1, DI, Nd.32	972	0x3CC	TxPDO8*, Nd.12	1527	0x5F7	TxPDO10*, Nd.55
417	0x1A1	TxPDO1, DI, Nd.33	973	0x3CD	TxPDO8*, Nd.13	1528	0x5F8	TxPDO10*, Nd.56
418	0x1A2	TxPDO1, DI, Nd.34	974	0x3CE	TxPDO8*, Nd.14	1529	0x5F9	TxPDO10*, Nd.57
419	0x1A3	TxPDO1, DI, Nd.35	975	0x3CF	TxPDO8*, Nd.15	1530	0x5FA	TxPDO10*, Nd.58
420	0x1A4	TxPDO1, DI, Nd.36	976	0x3D0	TxPDO8*, Nd.16	1531	0x5FB	TxPDO10*, Nd.59
421	0x1A5	TxPDO1, DI, Nd.37	977	0x3D1	TxPDO8*, Nd.17	1532	0x5FC	TxPDO10*, Nd.60
422	0x1A6	TxPDO1, DI, Nd.38	978	0x3D2	TxPDO8*, Nd.18	1533	0x5FD	TxPDO10*, Nd.61
423	0x1A7	TxPDO1, DI, Nd.39	979	0x3D3	TxPDO8*, Nd.19	1534	0x5FE	TxPDO10*, Nd.62

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
424	0x1A8	TxPDO1, DI, Nd.40	980	0x3D4	TxPDO8*, Nd.20	1535	0x5FF	TxPDO10*, Nd.63
425	0x1A9	TxPDO1, DI, Nd.41	981	0x3D5	TxPDO8*, Nd.21	1537	0x601	SDO Rx Nd.1
426	0x1AA	TxPDO1, DI, Nd.42	982	0x3D6	TxPDO8*, Nd.22	1538	0x602	SDO Rx Nd.2
427	0x1AB	TxPDO1, DI, Nd.43	983	0x3D7	TxPDO8*, Nd.23	1539	0x603	SDO Rx Nd.3
428	0x1AC	TxPDO1, DI, Nd.44	984	0x3D8	TxPDO8*, Nd.24	1540	0x604	SDO Rx Nd.4
429	0x1AD	TxPDO1, DI, Nd.45	985	0x3D9	TxPDO8*, Nd.25	1541	0x605	SDO Rx Nd.5
430	0x1AE	TxPDO1, DI, Nd.46	986	0x3DA	TxPDO8*, Nd.26	1542	0x606	SDO Rx Nd.6
431	0x1AF	TxPDO1, DI, Nd.47	987	0x3DB	TxPDO8*, Nd.27	1543	0x607	SDO Rx Nd.7
432	0x1B0	TxPDO1, DI, Nd.48	988	0x3DC	TxPDO8*, Nd.28	1544	0x608	SDO Rx Nd.8
433	0x1B1	TxPDO1, DI, Nd.49	989	0x3DD	TxPDO8*, Nd.29	1545	0x609	SDO Rx Nd.9
434	0x1B2	TxPDO1, DI, Nd.50	990	0x3DE	TxPDO8*, Nd.30	1546	0x60A	SDO Rx Nd.10
435	0x1B3	TxPDO1, DI, Nd.51	991	0x3DF	TxPDO8*, Nd.31	1547	0x60B	SDO Rx Nd.11
436	0x1B4	TxPDO1, DI, Nd.52	992	0x3E0	TxPDO8*, Nd.32	1548	0x60C	SDO Rx Nd.12
437	0x1B5	TxPDO1, DI, Nd.53	993	0x3E1	TxPDO8*, Nd.33	1549	0x60D	SDO Rx Nd.13
438	0x1B6	TxPDO1, DI, Nd.54	994	0x3E2	TxPDO8*, Nd.34	1550	0x60E	SDO Rx Nd.14
439	0x1B7	TxPDO1, DI, Nd.55	995	0x3E3	TxPDO8*, Nd.35	1551	0x60F	SDO Rx Nd.15
440	0x1B8	TxPDO1, DI, Nd.56	996	0x3E4	TxPDO8*, Nd.36	1552	0x610	SDO Rx Nd.16
441	0x1B9	TxPDO1, DI, Nd.57	997	0x3E5	TxPDO8*, Nd.37	1553	0x611	SDO Rx Nd.17
442	0x1BA	TxPDO1, DI, Nd.58	998	0x3E6	TxPDO8*, Nd.38	1554	0x612	SDO Rx Nd.18
443	0x1BB	TxPDO1, DI, Nd.59	999	0x3E7	TxPDO8*, Nd.39	1555	0x613	SDO Rx Nd.19
444	0x1BC	TxPDO1, DI, Nd.60	1000	0x3E8	TxPDO8*, Nd.40	1556	0x614	SDO Rx Nd.20
445	0x1BD	TxPDO1, DI, Nd.61	1001	0x3E9	TxPDO8*, Nd.41	1557	0x615	SDO Rx Nd.21
446	0x1BE	TxPDO1, DI, Nd.62	1002	0x3EA	TxPDO8*, Nd.42	1558	0x616	SDO Rx Nd.22
447	0x1BF	TxPDO1, DI, Nd.63	1003	0x3EB	TxPDO8*, Nd.43	1559	0x617	SDO Rx Nd.23
449	0x1C1	TxPDO6*, Nd.1	1004	0x3EC	TxPDO8*, Nd.44	1560	0x618	SDO Rx Nd.24
450	0x1C2	TxPDO6*, Nd.2	1005	0x3ED	TxPDO8*, Nd.45	1561	0x619	SDO Rx Nd.25

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
451	0x1C3	TxPDO6*, Nd.3	1006	0x3EE	TxPDO8*, Nd.46	1562	0x61A	SDO Rx Nd.26
452	0x1C4	TxPDO6*, Nd.4	1007	0x3EF	TxPDO8*, Nd.47	1563	0x61B	SDO Rx Nd.27
453	0x1C5	TxPDO6*, Nd.5	1008	0x3F0	TxPDO8*, Nd.48	1564	0x61C	SDO Rx Nd.28
454	0x1C6	TxPDO6*, Nd.6	1009	0x3F1	TxPDO8*, Nd.49	1565	0x61D	SDO Rx Nd.29
455	0x1C7	TxPDO6*, Nd.7	1010	0x3F2	TxPDO8*, Nd.50	1566	0x61E	SDO Rx Nd.30
456	0x1C8	TxPDO6*, Nd.8	1011	0x3F3	TxPDO8*, Nd.51	1567	0x61F	SDO Rx Nd.31
457	0x1C9	TxPDO6*, Nd.9	1012	0x3F4	TxPDO8*, Nd.52	1568	0x620	SDO Rx Nd.32
458	0x1CA	TxPDO6*, Nd.10	1013	0x3F5	TxPDO8*, Nd.53	1569	0x621	SDO Rx Nd.33
459	0x1CB	TxPDO6*, Nd.11	1014	0x3F6	TxPDO8*, Nd.54	1570	0x622	SDO Rx Nd.34
460	0x1CC	TxPDO6*, Nd.12	1015	0x3F7	TxPDO8*, Nd.55	1571	0x623	SDO Rx Nd.35
461	0x1CD	TxPDO6*, Nd.13	1016	0x3F8	TxPDO8*, Nd.56	1572	0x624	SDO Rx Nd.36
462	0x1CE	TxPDO6*, Nd.14	1017	0x3F9	TxPDO8*, Nd.57	1573	0x625	SDO Rx Nd.37
463	0x1CF	TxPDO6*, Nd.15	1018	0x3FA	TxPDO8*, Nd.58	1574	0x626	SDO Rx Nd.38
464	0x1D0	TxPDO6*, Nd.16	1019	0x3FB	TxPDO8*, Nd.59	1575	0x627	SDO Rx Nd.39
465	0x1D1	TxPDO6*, Nd.17	1020	0x3FC	TxPDO8*, Nd.60	1576	0x628	SDO Rx Nd.40
466	0x1D2	TxPDO6*, Nd.18	1021	0x3FD	TxPDO8*, Nd.61	1577	0x629	SDO Rx Nd.41
467	0x1D3	TxPDO6*, Nd.19	1022	0x3FE	TxPDO8*, Nd.62	1578	0x62A	SDO Rx Nd.42
468	0x1D4	TxPDO6*, Nd.20	1023	0x3FF	TxPDO8*, Nd.63	1579	0x62B	SDO Rx Nd.43
469	0x1D5	TxPDO6*, Nd.21	1025	0x401	RxPDO3*, Nd.1	1580	0x62C	SDO Rx Nd.44
470	0x1D6	TxPDO6*, Nd.22	1026	0x402	RxPDO3*, Nd.2	1581	0x62D	SDO Rx Nd.45
471	0x1D7	TxPDO6*, Nd.23	1027	0x403	RxPDO3*, Nd.3	1582	0x62E	SDO Rx Nd.46
472	0x1D8	TxPDO6*, Nd.24	1028	0x404	RxPDO3*, Nd.4	1583	0x62F	SDO Rx Nd.47
473	0x1D9	TxPDO6*, Nd.25	1029	0x405	RxPDO3*, Nd.5	1584	0x630	SDO Rx Nd.48
474	0x1DA	TxPDO6*, Nd.26	1030	0x406	RxPDO3*, Nd.6	1585	0x631	SDO Rx Nd.49
475	0x1DB	TxPDO6*, Nd.27	1031	0x407	RxPDO3*, Nd.7	1586	0x632	SDO Rx Nd.50
476	0x1DC	TxPDO6*, Nd.28	1032	0x408	RxPDO3*, Nd.8	1587	0x633	SDO Rx Nd.51

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
477	0x1DD	TxPDO6*, Nd.29	1033	0x409	RxPDO3*, Nd.9	1588	0x634	SDO Rx Nd.52
478	0x1DE	TxPDO6*, Nd.30	1034	0x40A	RxPDO3*, Nd.10	1589	0x635	SDO Rx Nd.53
479	0x1DF	TxPDO6*, Nd.31	1035	0x40B	RxPDO3*, Nd.11	1590	0x636	SDO Rx Nd.54
480	0x1E0	TxPDO6*, Nd.32	1036	0x40C	RxPDO3*, Nd.12	1591	0x637	SDO Rx Nd.55
481	0x1E1	TxPDO6*, Nd.33	1037	0x40D	RxPDO3*, Nd.13	1592	0x638	SDO Rx Nd.56
482	0x1E2	TxPDO6*, Nd.34	1038	0x40E	RxPDO3*, Nd.14	1593	0x639	SDO Rx Nd.57
483	0x1E3	TxPDO6*, Nd.35	1039	0x40F	RxPDO3*, Nd.15	1594	0x63A	SDO Rx Nd.58
484	0x1E4	TxPDO6*, Nd.36	1040	0x410	RxPDO3*, Nd.16	1595	0x63B	SDO Rx Nd.59
485	0x1E5	TxPDO6*, Nd.37	1041	0x411	RxPDO3*, Nd.17	1596	0x63C	SDO Rx Nd.60
486	0x1E6	TxPDO6*, Nd.38	1042	0x412	RxPDO3*, Nd.18	1597	0x63D	SDO Rx Nd.61
487	0x1E7	TxPDO6*, Nd.39	1043	0x413	RxPDO3*, Nd.19	1598	0x63E	SDO Rx Nd.62
488	0x1E8	TxPDO6*, Nd.40	1044	0x414	RxPDO3*, Nd.20	1599	0x63F	SDO Rx Nd.63
489	0x1E9	TxPDO6*, Nd.41	1045	0x415	RxPDO3*, Nd.21	1601	0x641	RxPDO10 *, Nd.1
490	0x1EA	TxPDO6*, Nd.42	1046	0x416	RxPDO3*, Nd.22	1602	0x642	RxPDO10 *, Nd.2
491	0x1EB	TxPDO6*, Nd.43	1047	0x417	RxPDO3*, Nd.23	1603	0x643	RxPDO10 *, Nd.3
492	0x1EC	TxPDO6*, Nd.44	1048	0x418	RxPDO3*, Nd.24	1604	0x644	RxPDO10 *, Nd.4
493	0x1ED	TxPDO6*, Nd.45	1049	0x419	RxPDO3*, Nd.25	1605	0x645	RxPDO10 *, Nd.5
494	0x1EE	TxPDO6*, Nd.46	1050	0x41A	RxPDO3*, Nd.26	1606	0x646	RxPDO10 *, Nd.6
495	0x1EF	TxPDO6*, Nd.47	1051	0x41B	RxPDO3*, Nd.27	1607	0x647	RxPDO10 *, Nd.7
496	0x1F0	TxPDO6*, Nd.48	1052	0x41C	RxPDO3*, Nd.28	1608	0x648	RxPDO10 *, Nd.8
497	0x1F1	TxPDO6*, Nd.49	1053	0x41D	RxPDO3*, Nd.29	1609	0x649	RxPDO10 *, Nd.9
498	0x1F2	TxPDO6*, Nd.50	1054	0x41E	RxPDO3*, Nd.30	1610	0x64A	RxPDO10 *, Nd.10
499	0x1F3	TxPDO6*, Nd.51	1055	0x41F	RxPDO3*, Nd.31	1611	0x64B	RxPDO10 *, Nd.11
500	0x1F4	TxPDO6*, Nd.52	1056	0x420	RxPDO3*, Nd.32	1612	0x64C	RxPDO10 *, Nd.12
501	0x1F5	TxPDO6*, Nd.53	1057	0x421	RxPDO3*, Nd.33	1613	0x64D	RxPDO10 *, Nd.13
502	0x1F6	TxPDO6*, Nd.54	1058	0x422	RxPDO3*, Nd.34	1614	0x64E	RxPDO10 *, Nd.14

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
503	0x1F7	TxPDO6*, Nd.55	1059	0x423	RxPDO3*, Nd.35	1615	0x64F	RxPDO10 *, Nd.15
504	0x1F8	TxPDO6*, Nd.56	1060	0x424	RxPDO3*, Nd.36	1616	0x650	RxPDO10 *, Nd.16
505	0x1F9	TxPDO6*, Nd.57	1061	0x425	RxPDO3*, Nd.37	1617	0x651	RxPDO10 *, Nd.17
506	0x1FA	TxPDO6*, Nd.58	1062	0x426	RxPDO3*, Nd.38	1618	0x652	RxPDO10 *, Nd.18
507	0x1FB	TxPDO6*, Nd.59	1063	0x427	RxPDO3*, Nd.39	1619	0x653	RxPDO10 *, Nd.19
508	0x1FC	TxPDO6*, Nd.60	1064	0x428	RxPDO3*, Nd.40	1620	0x654	RxPDO10 *, Nd.20
509	0x1FD	TxPDO6*, Nd.61	1065	0x429	RxPDO3*, Nd.41	1621	0x655	RxPDO10 *, Nd.21
510	0x1FE	TxPDO6*, Nd.62	1066	0x42A	RxPDO3*, Nd.42	1622	0x656	RxPDO10 *, Nd.22
511	0x1FF	TxPDO6*, Nd.63	1067	0x42B	RxPDO3*, Nd.43	1623	0x657	RxPDO10 *, Nd.23
513	0x201	RxPDO1, DO, Nd.1	1068	0x42C	RxPDO3*, Nd.44	1624	0x658	RxPDO10 *, Nd.24
514	0x202	RxPDO1, DO, Nd.2	1069	0x42D	RxPDO3*, Nd.45	1625	0x659	RxPDO10 *, Nd.25
515	0x203	RxPDO1, DO, Nd.3	1070	0x42E	RxPDO3*, Nd.46	1626	0x65A	RxPDO10 *, Nd.26
516	0x204	RxPDO1, DO, Nd.4	1071	0x42F	RxPDO3*, Nd.47	1627	0x65B	RxPDO10 *, Nd.27
517	0x205	RxPDO1, DO, Nd.5	1072	0x430	RxPDO3*, Nd.48	1628	0x65C	RxPDO10 *, Nd.28
518	0x206	RxPDO1, DO, Nd.6	1073	0x431	RxPDO3*, Nd.49	1629	0x65D	RxPDO10 *, Nd.29
519	0x207	RxPDO1, DO, Nd.7	1074	0x432	RxPDO3*, Nd.50	1630	0x65E	RxPDO10 *, Nd.30
520	0x208	RxPDO1, DO, Nd.8	1075	0x433	RxPDO3*, Nd.51	1631	0x65F	RxPDO10 *, Nd.31
521	0x209	RxPDO1, DO, Nd.9	1076	0x434	RxPDO3*, Nd.52	1632	0x660	RxPDO10 *, Nd.32
522	0x20A	RxPDO1, DO, Nd.10	1077	0x435	RxPDO3*, Nd.53	1633	0x661	RxPDO10 *, Nd.33
523	0x20B	RxPDO1, DO, Nd.11	1078	0x436	RxPDO3*, Nd.54	1634	0x662	RxPDO10 *, Nd.34
524	0x20C	RxPDO1, DO, Nd.12	1079	0x437	RxPDO3*, Nd.55	1635	0x663	RxPDO10 *, Nd.35
525	0x20D	RxPDO1, DO, Nd.13	1080	0x438	RxPDO3*, Nd.56	1636	0x664	RxPDO10 *, Nd.36
526	0x20E	RxPDO1, DO, Nd.14	1081	0x439	RxPDO3*, Nd.57	1637	0x665	RxPDO10 *, Nd.37
527	0x20F	RxPDO1, DO, Nd.15	1082	0x43A	RxPDO3*, Nd.58	1638	0x666	RxPDO10 *, Nd.38
528	0x210	RxPDO1, DO, Nd.16	1083	0x43B	RxPDO3*, Nd.59	1639	0x667	RxPDO10 *, Nd.39
529	0x211	RxPDO1, DO, Nd.17	1084	0x43C	RxPDO3*, Nd.60	1640	0x668	RxPDO10 *, Nd.40

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
530	0x212	RxPDO1, DO, Nd.18	1085	0x43D	RxPDO3*, Nd.61	1641	0x669	RxPDO10*, Nd.41
531	0x213	RxPDO1, DO, Nd.19	1086	0x43E	RxPDO3*, Nd.62	1642	0x66A	RxPDO10*, Nd.42
532	0x214	RxPDO1, DO, Nd.20	1087	0x43F	RxPDO3*, Nd.63	1643	0x66B	RxPDO10*, Nd.43
533	0x215	RxPDO1, DO, Nd.21	1089	0x441	RxPDO8*, Nd.1	1644	0x66C	RxPDO10*, Nd.44
534	0x216	RxPDO1, DO, Nd.22	1090	0x442	RxPDO8*, Nd.2	1645	0x66D	RxPDO10*, Nd.45
535	0x217	RxPDO1, DO, Nd.23	1091	0x443	RxPDO8*, Nd.3	1646	0x66E	RxPDO10*, Nd.46
536	0x218	RxPDO1, DO, Nd.24	1092	0x444	RxPDO8*, Nd.4	1647	0x66F	RxPDO10*, Nd.47
537	0x219	RxPDO1, DO, Nd.25	1093	0x445	RxPDO8*, Nd.5	1648	0x670	RxPDO10*, Nd.48
538	0x21A	RxPDO1, DO, Nd.26	1094	0x446	RxPDO8*, Nd.6	1649	0x671	RxPDO10*, Nd.49
539	0x21B	RxPDO1, DO, Nd.27	1095	0x447	RxPDO8*, Nd.7	1650	0x672	RxPDO10*, Nd.50
540	0x21C	RxPDO1, DO, Nd.28	1096	0x448	RxPDO8*, Nd.8	1651	0x673	RxPDO10*, Nd.51
541	0x21D	RxPDO1, DO, Nd.29	1097	0x449	RxPDO8*, Nd.9	1652	0x674	RxPDO10*, Nd.52
542	0x21E	RxPDO1, DO, Nd.30	1098	0x44A	RxPDO8*, Nd.10	1653	0x675	RxPDO10*, Nd.53
543	0x21F	RxPDO1, DO, Nd.31	1099	0x44B	RxPDO8*, Nd.11	1654	0x676	RxPDO10*, Nd.54
544	0x220	RxPDO1, DO, Nd.32	1100	0x44C	RxPDO8*, Nd.12	1655	0x677	RxPDO10*, Nd.55
545	0x221	RxPDO1, DO, Nd.33	1101	0x44D	RxPDO8*, Nd.13	1656	0x678	RxPDO10*, Nd.56
546	0x222	RxPDO1, DO, Nd.34	1102	0x44E	RxPDO8*, Nd.14	1657	0x679	RxPDO10*, Nd.57
547	0x223	RxPDO1, DO, Nd.35	1103	0x44F	RxPDO8*, Nd.15	1658	0x67A	RxPDO10*, Nd.58
548	0x224	RxPDO1, DO, Nd.36	1104	0x450	RxPDO8*, Nd.16	1659	0x67B	RxPDO10*, Nd.59
549	0x225	RxPDO1, DO, Nd.37	1105	0x451	RxPDO8*, Nd.17	1660	0x67C	RxPDO10*, Nd.60
550	0x226	RxPDO1, DO, Nd.38	1106	0x452	RxPDO8*, Nd.18	1661	0x67D	RxPDO10*, Nd.61
551	0x227	RxPDO1, DO, Nd.39	1107	0x453	RxPDO8*, Nd.19	1662	0x67E	RxPDO10*, Nd.62
552	0x228	RxPDO1, DO, Nd.40	1108	0x454	RxPDO8*, Nd.20	1663	0x67F	RxPDO10*, Nd.63
553	0x229	RxPDO1, DO, Nd.41	1109	0x455	RxPDO8*, Nd.21	1665	0x681	TxPDO5*, Nd.1
554	0x22A	RxPDO1, DO, Nd.42	1110	0x456	RxPDO8*, Nd.22	1666	0x682	TxPDO5*, Nd.2
555	0x22B	RxPDO1, DO, Nd.43	1111	0x457	RxPDO8*, Nd.23	1667	0x683	TxPDO5*, Nd.3

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
556	0x22C	RxPDO1, DO, Nd.44	1112	0x458	RxPDO8*, Nd.24	1668	0x684	TxPDO5*, Nd.4
557	0x22D	RxPDO1, DO, Nd.45	1113	0x459	RxPDO8*, Nd.25	1669	0x685	TxPDO5*, Nd.5
558	0x22E	RxPDO1, DO, Nd.46	1114	0x45A	RxPDO8*, Nd.26	1670	0x686	TxPDO5*, Nd.6
559	0x22F	RxPDO1, DO, Nd.47	1115	0x45B	RxPDO8*, Nd.27	1671	0x687	TxPDO5*, Nd.7
560	0x230	RxPDO1, DO, Nd.48	1116	0x45C	RxPDO8*, Nd.28	1672	0x688	TxPDO5*, Nd.8
561	0x231	RxPDO1, DO, Nd.49	1117	0x45D	RxPDO8*, Nd.29	1673	0x689	TxPDO5*, Nd.9
562	0x232	RxPDO1, DO, Nd.50	1118	0x45E	RxPDO8*, Nd.30	1674	0x68A	TxPDO5*, Nd.10
563	0x233	RxPDO1, DO, Nd.51	1119	0x45F	RxPDO8*, Nd.31	1675	0x68B	TxPDO5*, Nd.11
564	0x234	RxPDO1, DO, Nd.52	1120	0x460	RxPDO8*, Nd.32	1676	0x68C	TxPDO5*, Nd.12
565	0x235	RxPDO1, DO, Nd.53	1121	0x461	RxPDO8*, Nd.33	1677	0x68D	TxPDO5*, Nd.13
566	0x236	RxPDO1, DO, Nd.54	1122	0x462	RxPDO8*, Nd.34	1678	0x68E	TxPDO5*, Nd.14
567	0x237	RxPDO1, DO, Nd.55	1123	0x463	RxPDO8*, Nd.35	1679	0x68F	TxPDO5*, Nd.15
568	0x238	RxPDO1, DO, Nd.56	1124	0x464	RxPDO8*, Nd.36	1680	0x690	TxPDO5*, Nd.16
569	0x239	RxPDO1, DO, Nd.57	1125	0x465	RxPDO8*, Nd.37	1681	0x691	TxPDO5*, Nd.17
570	0x23A	RxPDO1, DO, Nd.58	1126	0x466	RxPDO8*, Nd.38	1682	0x692	TxPDO5*, Nd.18
571	0x23B	RxPDO1, DO, Nd.59	1127	0x467	RxPDO8*, Nd.39	1683	0x693	TxPDO5*, Nd.19
572	0x23C	RxPDO1, DO, Nd.60	1128	0x468	RxPDO8*, Nd.40	1684	0x694	TxPDO5*, Nd.20
573	0x23D	RxPDO1, DO, Nd.61	1129	0x469	RxPDO8*, Nd.41	1685	0x695	TxPDO5*, Nd.21
574	0x23E	RxPDO1, DO, Nd.62	1130	0x46A	RxPDO8*, Nd.42	1686	0x696	TxPDO5*, Nd.22
575	0x23F	RxPDO1, DO, Nd.63	1131	0x46B	RxPDO8*, Nd.43	1687	0x697	TxPDO5*, Nd.23
577	0x241	RxPDO6*, Nd.1	1132	0x46C	RxPDO8*, Nd.44	1688	0x698	TxPDO5*, Nd.24
578	0x242	RxPDO6*, Nd.2	1133	0x46D	RxPDO8*, Nd.45	1689	0x699	TxPDO5*, Nd.25
579	0x243	RxPDO6*, Nd.3	1134	0x46E	RxPDO8*, Nd.46	1690	0x69A	TxPDO5*, Nd.26
580	0x244	RxPDO6*, Nd.4	1135	0x46F	RxPDO8*, Nd.47	1691	0x69B	TxPDO5*, Nd.27
581	0x245	RxPDO6*, Nd.5	1136	0x470	RxPDO8*, Nd.48	1692	0x69C	TxPDO5*, Nd.28
582	0x246	RxPDO6*, Nd.6	1137	0x471	RxPDO8*, Nd.49	1693	0x69D	TxPDO5*, Nd.29

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
583	0x247	RxPDO6*, Nd.7	1138	0x472	RxPDO8*, Nd.50	1694	0x69E	TxPDO5*, Nd.30
584	0x248	RxPDO6*, Nd.8	1139	0x473	RxPDO8*, Nd.51	1695	0x69F	TxPDO5*, Nd.31
585	0x249	RxPDO6*, Nd.9	1140	0x474	RxPDO8*, Nd.52	1696	0x6A0	TxPDO5*, Nd.32
586	0x24A	RxPDO6*, Nd.10	1141	0x475	RxPDO8*, Nd.53	1697	0x6A1	TxPDO5*, Nd.33
587	0x24B	RxPDO6*, Nd.11	1142	0x476	RxPDO8*, Nd.54	1698	0x6A2	TxPDO5*, Nd.34
588	0x24C	RxPDO6*, Nd.12	1143	0x477	RxPDO8*, Nd.55	1699	0x6A3	TxPDO5*, Nd.35
589	0x24D	RxPDO6*, Nd.13	1144	0x478	RxPDO8*, Nd.56	1700	0x6A4	TxPDO5*, Nd.36
590	0x24E	RxPDO6*, Nd.14	1145	0x479	RxPDO8*, Nd.57	1701	0x6A5	TxPDO5*, Nd.37
591	0x24F	RxPDO6*, Nd.15	1146	0x47A	RxPDO8*, Nd.58	1702	0x6A6	TxPDO5*, Nd.38
592	0x250	RxPDO6*, Nd.16	1147	0x47B	RxPDO8*, Nd.59	1703	0x6A7	TxPDO5*, Nd.39
593	0x251	RxPDO6*, Nd.17	1148	0x47C	RxPDO8*, Nd.60	1704	0x6A8	TxPDO5*, Nd.40
594	0x252	RxPDO6*, Nd.18	1149	0x47D	RxPDO8*, Nd.61	1705	0x6A9	TxPDO5*, Nd.41
595	0x253	RxPDO6*, Nd.19	1150	0x47E	RxPDO8*, Nd.62	1706	0x6AA	TxPDO5*, Nd.42
596	0x254	RxPDO6*, Nd.20	1151	0x47F	RxPDO8*, Nd.63	1707	0x6AB	TxPDO5*, Nd.43
597	0x255	RxPDO6*, Nd.21	1153	0x481	TxPDO4*, Nd.1	1708	0x6AC	TxPDO5*, Nd.44
598	0x256	RxPDO6*, Nd.22	1154	0x482	TxPDO4*, Nd.2	1709	0x6AD	TxPDO5*, Nd.45
599	0x257	RxPDO6*, Nd.23	1155	0x483	TxPDO4*, Nd.3	1710	0x6AE	TxPDO5*, Nd.46
600	0x258	RxPDO6*, Nd.24	1156	0x484	TxPDO4*, Nd.4	1711	0x6AF	TxPDO5*, Nd.47
601	0x259	RxPDO6*, Nd.25	1157	0x485	TxPDO4*, Nd.5	1712	0x6B0	TxPDO5*, Nd.48
602	0x25A	RxPDO6*, Nd.26	1158	0x486	TxPDO4*, Nd.6	1713	0x6B1	TxPDO5*, Nd.49
603	0x25B	RxPDO6*, Nd.27	1159	0x487	TxPDO4*, Nd.7	1714	0x6B2	TxPDO5*, Nd.50
604	0x25C	RxPDO6*, Nd.28	1160	0x488	TxPDO4*, Nd.8	1715	0x6B3	TxPDO5*, Nd.51
605	0x25D	RxPDO6*, Nd.29	1161	0x489	TxPDO4*, Nd.9	1716	0x6B4	TxPDO5*, Nd.52
606	0x25E	RxPDO6*, Nd.30	1162	0x48A	TxPDO4*, Nd.10	1717	0x6B5	TxPDO5*, Nd.53
607	0x25F	RxPDO6*, Nd.31	1163	0x48B	TxPDO4*, Nd.11	1718	0x6B6	TxPDO5*, Nd.54
608	0x260	RxPDO6*, Nd.32	1164	0x48C	TxPDO4*, Nd.12	1719	0x6B7	TxPDO5*, Nd.55

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
609	0x261	RxPDO6*, Nd.33	1165	0x48D	TxPDO4*, Nd.13	1720	0x6B8	TxPDO5*, Nd.56
610	0x262	RxPDO6*, Nd.34	1166	0x48E	TxPDO4*, Nd.14	1721	0x6B9	TxPDO5*, Nd.57
611	0x263	RxPDO6*, Nd.35	1167	0x48F	TxPDO4*, Nd.15	1722	0x6BA	TxPDO5*, Nd.58
612	0x264	RxPDO6*, Nd.36	1168	0x490	TxPDO4*, Nd.16	1723	0x6BB	TxPDO5*, Nd.59
613	0x265	RxPDO6*, Nd.3	1169	0x491	TxPDO4*, Nd.17	1724	0x6BC	TxPDO5*, Nd.60
614	0x266	RxPDO6*, Nd.8	1170	0x492	TxPDO4*, Nd.18	1725	0x6BD	TxPDO5*, Nd.61
615	0x267	RxPDO6*, Nd.39	1171	0x493	TxPDO4*, Nd.19	1726	0x6BE	TxPDO5*, Nd.62
616	0x268	RxPDO6*, N.40	1172	0x494	TxPDO4*, Nd.20	1727	0x6BF	TxPDO5*, Nd.63
617	0x269	RxPDO6*, d.41	1173	0x495	TxPDO4*, Nd.21	1729	0x6C1	TxPDO11 *, Nd.1
618	0x26A	RxPDO6*, Nd.42	1174	0x496	TxPDO4*, Nd.22	1730	0x6C2	TxPDO11 *, Nd.2
619	0x26B	RxPDO6*, Nd.43	1175	0x497	TxPDO4*, Nd.23	1731	0x6C3	TxPDO11 *, Nd.3
620	0x26C	RxPDO6*, Nd.44	1176	0x498	TxPDO4*, Nd.24	1732	0x6C4	TxPDO11 *, Nd.4
621	0x26D	RxPDO6*, Nd.45	1177	0x499	TxPDO4*, Nd.25	1733	0x6C5	TxPDO11 *, Nd.5
622	0x26E	RxPDO6*, Nd.46	1178	0x49A	TxPDO4*, Nd.26	1734	0x6C6	TxPDO11 *, Nd.6
623	0x26F	RxPDO6*, Nd.47	1179	0x49B	TxPDO4*, Nd.27	1735	0x6C7	TxPDO11 *, Nd.7
624	0x270	RxPDO6*, Nd.48	1180	0x49C	TxPDO4*, Nd.28	1736	0x6C8	TxPDO11 *, Nd.8
625	0x271	RxPDO6*, Nd.49	1181	0x49D	TxPDO4*, Nd.29	1737	0x6C9	TxPDO11 *, Nd.9
626	0x272	RxPDO6*, Nd.50	1182	0x49E	TxPDO4*, Nd.30	1738	0x6CA	TxPDO11 *, Nd.10
627	0x273	RxPDO6*, Nd.51	1183	0x49F	TxPDO4*, Nd.31	1739	0x6CB	TxPDO11 *, Nd.11
628	0x274	RxPDO6*, Nd.52	1184	0x4A0	TxPDO4*, Nd.32	1740	0x6CC	TxPDO11 *, Nd.12
629	0x275	RxPDO6*, Nd.53	1185	0x4A1	TxPDO4*, Nd.33	1741	0x6CD	TxPDO11 *, Nd.13
630	0x276	RxPDO6*, Nd.54	1186	0x4A2	TxPDO4*, Nd.34	1742	0x6CE	TxPDO11 *, Nd.14
631	0x277	RxPDO6*, Nd.55	1187	0x4A3	TxPDO4*, Nd.35	1743	0x6CF	TxPDO11 *, Nd.15
632	0x278	RxPDO6*, Nd.56	1188	0x4A4	TxPDO4*, Nd.36	1744	0x6D0	TxPDO11 *, Nd.16
633	0x279	RxPDO6*, Nd.57	1189	0x4A5	TxPDO4*, Nd.37	1745	0x6D1	TxPDO11 *, Nd.17
634	0x27A	RxPDO6*, Nd.58	1190	0x4A6	TxPDO4*, Nd.48	1746	0x6D2	TxPDO11 *, Nd.18

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
635	0x27B	RxPDO6*, Nd.59	1191	0x4A7	TxPDO4*, Nd.49	1747	0x6D3	TxPDO11 *, Nd.19
636	0x27C	RxPDO6*, Nd.60	1192	0x4A8	TxPDO4*, Nd.40	1748	0x6D4	TxPDO11 *, Nd.20
637	0x27D	RxPDO6*, Nd.61	1193	0x4A9	TxPDO4*, Nd.41	1749	0x6D5	TxPDO11 *, Nd.21
638	0x27E	RxPDO6*, Nd.62	1194	0x4AA	TxPDO4*, Nd.42	1750	0x6D6	TxPDO11 *, Nd.22
639	0x27F	RxPDO6*, Nd.63	1195	0x4AB	TxPDO4*, Nd.43	1751	0x6D7	TxPDO11 *, Nd.23
641	0x281	TxPDO2, AI, Nd.1	1196	0x4AC	TxPDO4*, Nd.44	1752	0x6D8	TxPDO11 *, Nd.24
642	0x282	TxPDO2, AI, Nd.2	1197	0x4AD	TxPDO4*, Nd.45	1753	0x6D9	TxPDO11 *, Nd.25
643	0x283	TxPDO2, AI, Nd.3	1198	0x4AE	TxPDO4*, Nd.46	1754	0x6DA	TxPDO11 *, Nd.26
644	0x284	TxPDO2, AI, Nd.4	1199	0x4AF	TxPDO4*, Nd.47	1755	0x6DB	TxPDO11 *, Nd.27
645	0x285	TxPDO2, AI, Nd.5	1200	0x4B0	TxPDO4*, Nd.48	1756	0x6DC	TxPDO11 *, Nd.28
646	0x286	TxPDO2, AI, Nd.6	1201	0x4B1	TxPDO4*, Nd.49	1757	0x6DD	TxPDO11 *, Nd.29
647	0x287	TxPDO2, AI, Nd.7	1202	0x4B2	TxPDO4*, Nd.50	1758	0x6DE	TxPDO11 *, Nd.30
648	0x288	TxPDO2, AI, Nd.8	1203	0x4B3	TxPDO4*, Nd.51	1759	0x6DF	TxPDO11 *, Nd.31
649	0x289	TxPDO2, AI, Nd.9	1204	0x4B4	TxPDO4*, Nd.52	1760	0x6E0	TxPDO11 *, Nd.32
650	0x28A	TxPDO2, AI, Nd.10	1205	0x4B5	TxPDO4*, Nd.53	1761	0x6E1	TxPDO11 *, Nd.33
651	0x28B	TxPDO2, AI, Nd.11	1206	0x4B6	TxPDO4*, Nd.54	1762	0x6E2	TxPDO11 *, Nd.34
652	0x28C	TxPDO2, AI, Nd.12	1207	0x4B7	TxPDO4*, Nd.55	1763	0x6E3	TxPDO11 *, Nd.35
653	0x28D	TxPDO2, AI, Nd.13	1208	0x4B8	TxPDO4*, Nd.56	1764	0x6E4	TxPDO11 *, Nd.36
654	0x28E	TxPDO2, AI, Nd.14	1209	0x4B9	TxPDO4*, Nd.57	1765	0x6E5	TxPDO11 *, Nd.37
655	0x28F	TxPDO2, AI, Nd.15	1210	0x4BA	TxPDO4*, Nd.58	1766	0x6E6	TxPDO11 *, Nd.38
656	0x290	TxPDO2, AI, Nd.16	1211	0x4BB	TxPDO4*, Nd.59	1767	0x6E7	TxPDO11 *, Nd.39
657	0x291	TxPDO2, AI, Nd.17	1212	0x4BC	TxPDO4*, Nd.60	1768	0x6E8	TxPDO11 *, Nd.40
658	0x292	TxPDO2, AI, Nd.18	1213	0x4BD	TxPDO4*, Nd.61	1769	0x6E9	TxPDO11 *, Nd.41
659	0x293	TxPDO2, AI, Nd.19	1214	0x4BE	TxPDO4*, Nd.62	1770	0x6EA	TxPDO11 *, Nd.42
660	0x294	TxPDO2, AI, Nd.20	1215	0x4BF	TxPDO4*, Nd.63	1771	0x6EB	TxPDO11 *, Nd.43
661	0x295	TxPDO2, AI, Nd.21	1217	0x4C1	TxPDO9*, Nd.1	1772	0x6EC	TxPDO11 *, Nd.44

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
662	0x296	TxPDO2, AI, Nd.22	1218	0x4C2	TxPDO9*, Nd.2	1773	0x6ED	TxPDO11*, Nd.45
663	0x297	TxPDO2, AI, Nd.23	1219	0x4C3	TxPDO9*, Nd.3	1774	0x6EE	TxPDO11*, Nd.46
664	0x298	TxPDO2, AI, Nd.24	1220	0x4C4	TxPDO9*, Nd.4	1775	0x6EF	TxPDO11*, Nd.47
665	0x299	TxPDO2, AI, Nd.25	1221	0x4C5	TxPDO9*, Nd.5	1776	0x6F0	TxPDO11*, Nd.48
666	0x29A	TxPDO2, AI, Nd.26	1222	0x4C6	TxPDO9*, Nd.6	1777	0x6F1	TxPDO11*, Nd.49
667	0x29B	TxPDO2, AI, Nd.27	1223	0x4C7	TxPDO9*, Nd.7	1778	0x6F2	TxPDO11*, Nd.50
668	0x29C	TxPDO2, AI, Nd.28	1224	0x4C8	TxPDO9*, Nd.8	1779	0x6F3	TxPDO11*, Nd.51
669	0x29D	TxPDO2, AI, Nd.29	1225	0x4C9	TxPDO9*, Nd.9	1780	0x6F4	TxPDO11*, Nd.52
670	0x29E	TxPDO2, AI, Nd.30	1226	0x4CA	TxPDO9*, Nd.10	1781	0x6F5	TxPDO11*, Nd.53
671	0x29F	TxPDO2, AI, Nd.31	1227	0x4CB	TxPDO9*, Nd.11	1782	0x6F6	TxPDO11*, Nd.54
672	0x2A0	TxPDO2, AI, Nd.32	1228	0x4CC	TxPDO9*, Nd.12	1783	0x6F7	TxPDO11*, Nd.55
673	0x2A1	TxPDO2, AI, Nd.33	1229	0x4CD	TxPDO9*, Nd.13	1784	0x6F8	TxPDO11*, Nd.56
674	0x2A2	TxPDO2, AI, Nd.34	1230	0x4CE	TxPDO9*, Nd.14	1785	0x6F9	TxPDO11*, Nd.57
675	0x2A3	TxPDO2, AI, Nd.35	1231	0x4CF	TxPDO9*, Nd.15	1786	0x6FA	TxPDO11*, Nd.58
676	0x2A4	TxPDO2, AI, Nd.36	1232	0x4D0	TxPDO9*, Nd.16	1787	0x6FB	TxPDO11*, Nd.59
677	0x2A5	TxPDO2, AI, Nd.37	1233	0x4D1	TxPDO9*, Nd.17	1788	0x6FC	TxPDO11*, Nd.60
678	0x2A6	TxPDO2, AI, Nd.38	1234	0x4D2	TxPDO9*, Nd.18	1789	0x6FD	TxPDO11*, Nd.61
679	0x2A7	TxPDO2, AI, Nd.39	1235	0x4D3	TxPDO9*, Nd.19	1790	0x6FE	TxPDO11*, Nd.62
680	0x2A8	TxPDO2, AI, Nd.40	1236	0x4D4	TxPDO9*, Nd.20	1791	0x6FF	TxPDO11*, Nd.63
681	0x2A9	TxPDO2, AI, Nd.41	1237	0x4D5	TxPDO9*, Nd.21	1793	0x701	Guarding Nd.1
682	0x2AA	TxPDO2, AI, Nd.42	1238	0x4D6	TxPDO9*, Nd.22	1794	0x702	Guarding Nd.2
683	0x2AB	TxPDO2, AI, Nd.43	1239	0x4D7	TxPDO9*, Nd.23	1795	0x703	Guarding Nd.3
684	0x2AC	TxPDO2, AI, Nd.44	1240	0x4D8	TxPDO9*, Nd.24	1796	0x704	Guarding Nd.4
685	0x2AD	TxPDO2, AI, Nd.45	1241	0x4D9	TxPDO9*, Nd.25	1797	0x705	Guarding Nd.5
686	0x2AE	TxPDO2, AI, Nd.46	1242	0x4DA	TxPDO9*, Nd.26	1798	0x706	Guarding Nd.6
687	0x2AF	TxPDO2, AI, Nd.47	1243	0x4DB	TxPDO9*, Nd.27	1799	0x707	Guarding Nd.7

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
688	0x2B0	TxPDO2, AI, Nd.48	1244	0x4DC	TxPDO9*, Nd.28	1800	0x708	Guarding Nd.8
689	0x2B1	TxPDO2, AI, Nd.49	1245	0x4DD	TxPDO9*, Nd.29	1801	0x709	Guarding Nd.9
690	0x2B2	TxPDO2, AI, Nd.50	1246	0x4DE	TxPDO9*, Nd.30	1802	0x70A	Guarding Nd.10
691	0x2B3	TxPDO2, AI, Nd.51	1247	0x4DF	TxPDO9*, Nd.31	1803	0x70B	Guarding Nd.11
692	0x2B4	TxPDO2, AI, Nd.52	1248	0x4E0	TxPDO9*, Nd.32	1804	0x70C	Guarding Nd.12
693	0x2B5	TxPDO2, AI, Nd.53	1249	0x4E1	TxPDO9*, Nd.33	1805	0x70D	Guarding Nd.13
694	0x2B6	TxPDO2, AI, Nd.54	1250	0x4E2	TxPDO9*, Nd.34	1806	0x70E	Guarding Nd.14
695	0x2B7	TxPDO2, AI, Nd.55	1251	0x4E3	TxPDO9*, Nd.35	1807	0x70F	Guarding Nd.15
696	0x2B8	TxPDO2, AI, Nd.56	1252	0x4E4	TxPDO9*, Nd.36	1808	0x710	Guarding Nd.16
697	0x2B9	TxPDO2, AI, Nd.57	1253	0x4E5	TxPDO9*, Nd.37	1809	0x711	Guarding Nd.17
698	0x2BA	TxPDO2, AI, Nd.58	1254	0x4E6	TxPDO9*, Nd.38	1810	0x712	Guarding Nd.18
699	0x2BB	TxPDO2, AI, Nd.59	1255	0x4E7	TxPDO9*, Nd.39	1811	0x713	Guarding Nd.19
700	0x2BC	TxPDO2, AI, Nd.60	1256	0x4E8	TxPDO9*, Nd.40	1812	0x714	Guarding Nd.20
701	0x2BD	TxPDO2, AI, Nd.61	1257	0x4E9	TxPDO9*, Nd.41	1813	0x715	Guarding Nd.21
702	0x2BE	TxPDO2, AI, Nd.62	1258	0x4EA	TxPDO9*, Nd.42	1814	0x716	Guarding Nd.22
703	0x2BF	TxPDO2, AI, Nd.63	1259	0x4EB	TxPDO9*, Nd.43	1815	0x717	Guarding Nd.23
705	0x2C1	TxPDO7*, Nd.1	1260	0x4EC	TxPDO9*, Nd.44	1816	0x718	Guarding Nd.24
706	0x2C2	TxPDO7*, Nd.2	1261	0x4ED	TxPDO9*, Nd.45	1817	0x719	Guarding Nd.25
707	0x2C3	TxPDO7*, Nd.3	1262	0x4EE	TxPDO9*, Nd.46	1818	0x71A	Guarding Nd.26
708	0x2C4	TxPDO7*, Nd.4	1263	0x4EF	TxPDO9*, Nd.47	1819	0x71B	Guarding Nd.27
709	0x2C5	TxPDO7*, Nd.5	1264	0x4F0	TxPDO9*, Nd.48	1820	0x71C	Guarding Nd.28
710	0x2C6	TxPDO7*, Nd.6	1265	0x4F1	TxPDO9*, Nd.49	1821	0x71D	Guarding Nd.29
711	0x2C7	TxPDO7*, Nd.7	1266	0x4F2	TxPDO9*, Nd.50	1822	0x71E	Guarding Nd.30
712	0x2C8	TxPDO7*, Nd.8	1267	0x4F3	TxPDO9*, Nd.51	1823	0x71F	Guarding Nd.31
713	0x2C9	TxPDO7*, Nd.9	1268	0x4F4	TxPDO9*, Nd.52	1824	0x720	Guarding Nd.32
714	0x2CA	TxPDO7*, Nd.10	1269	0x4F5	TxPDO9*, Nd.53	1825	0x721	Guarding Nd.33

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
715	0x2CB	TxPDO7*, Nd.11	1270	0x4F6	TxPDO9*, Nd.54	1826	0x722	Guarding Nd.34
716	0x2CC	TxPDO7*, Nd.12	1271	0x4F7	TxPDO9*, Nd.55	1827	0x723	Guarding Nd.35
717	0x2CD	TxPDO7*, Nd.13	1272	0x4F8	TxPDO9*, Nd.56	1828	0x724	Guarding Nd.36
718	0x2CE	TxPDO7*, Nd.14	1273	0x4F9	TxPDO9*, Nd.57	1829	0x725	Guarding Nd.37
719	0x2CF	TxPDO7*, Nd.15	1274	0x4FA	TxPDO9*, Nd.58	1830	0x726	Guarding Nd.38
720	0x2D0	TxPDO7*, Nd.16	1275	0x4FB	TxPDO9*, Nd.59	1831	0x727	Guarding Nd.39
721	0x2D1	TxPDO7*, Nd.17	1276	0x4FC	TxPDO9*, Nd.60	1832	0x728	Guarding Nd.40
722	0x2D2	TxPDO7*, Nd.18	1277	0x4FD	TxPDO9*, Nd.61	1833	0x729	Guarding Nd.41
723	0x2D3	TxPDO7*, Nd.19	1278	0x4FE	TxPDO9*, Nd.62	1834	0x72A	Guarding Nd.42
724	0x2D4	TxPDO7*, Nd.20	1279	0x4FF	TxPDO9*, Nd.63	1835	0x72B	Guarding Nd.43
725	0x2D5	TxPDO7*, Nd.21	1281	0x501	RxPDO4*, Nd.1	1836	0x72C	Guarding Nd.44
726	0x2D6	TxPDO7*, Nd.22	1282	0x502	RxPDO4*, Nd.2	1837	0x72D	Guarding Nd.45
727	0x2D7	TxPDO7*, Nd.23	1283	0x503	RxPDO4*, Nd.3	1838	0x72E	Guarding Nd.46
728	0x2D8	TxPDO7*, Nd.24	1284	0x504	RxPDO4*, Nd.4	1839	0x72F	Guarding Nd.47
729	0x2D9	TxPDO7*, Nd.25	1285	0x505	RxPDO4*, Nd.5	1840	0x730	Guarding Nd.48
730	0x2DA	TxPDO7*, Nd.26	1286	0x506	RxPDO4*, Nd.6	1841	0x731	Guarding Nd.49
731	0x2DB	TxPDO7*, Nd.27	1287	0x507	RxPDO4*, Nd.7	1842	0x732	Guarding Nd.50
732	0x2DC	TxPDO7*, Nd.28	1288	0x508	RxPDO4*, Nd.8	1843	0x733	Guarding Nd.51
733	0x2DD	TxPDO7*, Nd.29	1289	0x509	RxPDO4*, Nd.9	1844	0x734	Guarding Nd.52
734	0x2DE	TxPDO7*, Nd.30	1290	0x50A	RxPDO4*, Nd.10	1845	0x735	Guarding Nd.53
735	0x2DF	TxPDO7*, Nd.31	1291	0x50B	RxPDO4*, Nd.11	1846	0x736	Guarding Nd.54
736	0x2E0	TxPDO7*, Nd.32	1292	0x50C	RxPDO4*, Nd.12	1847	0x737	Guarding Nd.55
737	0x2E1	TxPDO7*, Nd.33	1293	0x50D	RxPDO4*, Nd.13	1848	0x738	Guarding Nd.56
738	0x2E2	TxPDO7*, Nd.34	1294	0x50E	RxPDO4*, Nd.14	1849	0x739	Guarding Nd.57
739	0x2E3	TxPDO7*, Nd.35	1295	0x50F	RxPDO4*, Nd.15	1850	0x73A	Guarding Nd.58
740	0x2E4	TxPDO7*, Nd.36	1296	0x510	RxPDO4*, Nd.16	1851	0x73B	Guarding Nd.59

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
741	0x2E5	TxPDO7*, Nd.37	1297	0x511	RxPDO4*, Nd.17	1852	0x73C	Guarding Nd.60
742	0x2E6	TxPDO7*, Nd.38	1298	0x512	RxPDO4*, Nd.18	1853	0x73D	Guarding Nd.61
743	0x2E7	TxPDO7*, Nd.39	1299	0x513	RxPDO4*, Nd.19	1854	0x73E	Guarding Nd.62
744	0x2E8	TxPDO7*, Nd.40	1300	0x514	RxPDO4*, Nd.20	1855	0x73F	Guarding Nd.63
745	0x2E9	TxPDO7*, Nd.41	1301	0x515	RxPDO4*, Nd.21	1857	0x741	RxPDO11 *, Nd.1
746	0x2EA	TxPDO7*, Nd.42	1302	0x516	RxPDO4*, Nd.22	1858	0x742	RxPDO11 *, Nd.2
747	0x2EB	TxPDO7*, Nd.43	1303	0x517	RxPDO4*, Nd.23	1859	0x743	RxPDO11 *, Nd.3
748	0x2EC	TxPDO7*, Nd.44	1304	0x518	RxPDO4*, Nd.24	1860	0x744	RxPDO11 *, Nd.4
749	0x2ED	TxPDO7*, Nd.45	1305	0x519	RxPDO4*, Nd.25	1861	0x745	RxPDO11 *, Nd.5
750	0x2EE	TxPDO7*, Nd.46	1306	0x51A	RxPDO4*, Nd.26	1862	0x746	RxPDO11 *, Nd.6
751	0x2EF	TxPDO7*, Nd.47	1307	0x51B	RxPDO4*, Nd.27	1863	0x747	RxPDO11 *, Nd.7
752	0x2F0	TxPDO7*, Nd.48	1308	0x51C	RxPDO4*, Nd.28	1864	0x748	RxPDO11 *, Nd.8
753	0x2F1	TxPDO7*, Nd.49	1309	0x51D	RxPDO4*, Nd.29	1865	0x749	RxPDO11 *, Nd.9
754	0x2F2	TxPDO7*, Nd.50	1310	0x51E	RxPDO4*, Nd.30	1866	0x74A	RxPDO11 *, Nd.10
755	0x2F3	TxPDO7*, Nd.51	1311	0x51F	RxPDO4*, Nd.31	1867	0x74B	RxPDO11 *, Nd.11
756	0x2F4	TxPDO7*, Nd.52	1312	0x520	RxPDO4*, Nd.32	1868	0x74C	RxPDO11 *, Nd.12
757	0x2F5	TxPDO7*, Nd.53	1313	0x521	RxPDO4*, Nd.33	1869	0x74D	RxPDO11 *, Nd.13
758	0x2F6	TxPDO7*, Nd.54	1314	0x522	RxPDO4*, Nd.34	1870	0x74E	RxPDO11 *, Nd.14
759	0x2F7	TxPDO7*, Nd.55	1315	0x523	RxPDO4*, Nd.35	1871	0x74F	RxPDO11 *, Nd.15
760	0x2F8	TxPDO7*, Nd.56	1316	0x524	RxPDO4*, Nd.36	1872	0x750	RxPDO11 *, Nd.16
761	0x2F9	TxPDO7*, Nd.57	1317	0x525	RxPDO4*, Nd.37	1873	0x751	RxPDO11 *, Nd.17
762	0x2FA	TxPDO7*, Nd.58	1318	0x526	RxPDO4*, Nd.38	1874	0x752	RxPDO11 *, Nd.18
763	0x2FB	TxPDO7*, Nd.59	1319	0x527	RxPDO4*, Nd.39	1875	0x753	RxPDO11 *, Nd.19
764	0x2FC	TxPDO7*, Nd.60	1320	0x528	RxPDO4*, Nd.40	1876	0x754	RxPDO11 *, Nd.20
765	0x2FD	TxPDO7*, Nd.61	1321	0x529	RxPDO4*, Nd.41	1877	0x755	RxPDO11 *, Nd.21
766	0x2FE	TxPDO7*, Nd.62	1322	0x52A	RxPDO4*, Nd.42	1878	0x756	RxPDO11 *, Nd.22

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
767	0x2FF	TxPDO7*, Nd.63	1323	0x52B	RxPDO4*, Nd.43	1879	0x757	RxPDO11 *, Nd.23
769	0x301	RxPDO2, AO, Nd.1	1324	0x52C	RxPDO4*, Nd.44	1880	0x758	RxPDO11 *, Nd.24
770	0x302	RxPDO2, AO, Nd.2	1325	0x52D	RxPDO4*, Nd.45	1881	0x759	RxPDO11 *, Nd.25
771	0x303	RxPDO2, AO, Nd.3	1326	0x52E	RxPDO4*, Nd.46	1882	0x75A	RxPDO11 *, Nd.26
772	0x304	RxPDO2, AO, Nd.4	1327	0x52F	RxPDO4*, Nd.47	1883	0x75B	RxPDO11 *, Nd.27
773	0x305	RxPDO2, AO, Nd.5	1328	0x530	RxPDO4*, Nd.48	1884	0x75C	RxPDO11 *, Nd.28
774	0x306	RxPDO2, AO, Nd.6	1329	0x531	RxPDO4*, Nd.49	1885	0x75D	RxPDO11 *, Nd.29
775	0x307	RxPDO2, AO, Nd.7	1330	0x532	RxPDO4*, Nd.50	1886	0x75E	RxPDO11 *, Nd.30
776	0x308	RxPDO2, AO, Nd.8	1331	0x533	RxPDO4*, Nd.51	1887	0x75F	RxPDO11 *, Nd.31
777	0x309	RxPDO2, AO, Nd.9	1332	0x534	RxPDO4*, Nd.52	1888	0x760	RxPDO11 *, Nd.32
778	0x30A	RxPDO2, AO, Nd.10	1333	0x535	RxPDO4*, Nd.53	1889	0x761	RxPDO11 *, Nd.33
779	0x30B	RxPDO2, AO, Nd.11	1334	0x536	RxPDO4*, Nd.54	1890	0x762	RxPDO11 *, Nd.34
780	0x30C	RxPDO2, AO, Nd.12	1335	0x537	RxPDO4*, Nd.55	1891	0x763	RxPDO11 *, Nd.35
781	0x30D	RxPDO2, AO, Nd.13	1336	0x538	RxPDO4*, Nd.56	1892	0x764	RxPDO11 *, Nd.36
782	0x30E	RxPDO2, AO, Nd.14	1337	0x539	RxPDO4*, Nd.57	1893	0x765	RxPDO11 *, Nd.37
783	0x30F	RxPDO2, AO, Nd.15	1338	0x53A	RxPDO4*, Nd.58	1894	0x766	RxPDO11 *, Nd.38
784	0x310	RxPDO2, AO, Nd.16	1339	0x53B	RxPDO4*, Nd.59	1895	0x767	RxPDO11 *, Nd.39
785	0x311	RxPDO2, AO, Nd.17	1340	0x53C	RxPDO4*, Nd.60	1896	0x768	RxPDO11 *, Nd.40
786	0x312	RxPDO2, AO, Nd.18	1341	0x53D	RxPDO4*, Nd.61	1897	0x769	RxPDO11 *, Nd.41
787	0x313	RxPDO2, AO, Nd.19	1342	0x53E	RxPDO4*, Nd.62	1898	0x76A	RxPDO11 *, Nd.42
788	0x314	RxPDO2, AO, Nd.20	1343	0x53F	RxPDO4*, Nd.63	1899	0x76B	RxPDO11 *, Nd.43
789	0x315	RxPDO2, AO, Nd.21	1345	0x541	RxPDO9*, Nd.1	1900	0x76C	RxPDO11 *, Nd.44
790	0x316	RxPDO2, AO, Nd.22	1346	0x542	RxPDO9*, Nd.2	1901	0x76D	RxPDO11 *, Nd.45
791	0x317	RxPDO2, AO, Nd.23	1347	0x543	RxPDO9*, Nd.3	1902	0x76E	RxPDO11 *, Nd.46
792	0x318	RxPDO2, AO, Nd.24	1348	0x544	RxPDO9*, Nd.4	1903	0x76F	RxPDO11 *, Nd.47
793	0x319	RxPDO2, AO, Nd.25	1349	0x545	RxPDO9*, Nd.5	1904	0x770	RxPDO11 *, Nd.48

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
794	0x31A	RxPDO2, AO, Nd.26	1350	0x546	RxPDO9*, Nd.6	1905	0x771	RxPDO11*, Nd.49
795	0x31B	RxPDO2, AO, Nd.27	1351	0x547	RxPDO9*, Nd.7	1906	0x772	RxPDO11*, Nd.50
796	0x31C	RxPDO2, AO, Nd.28	1352	0x548	RxPDO9*, Nd.8	1907	0x773	RxPDO11*, Nd.51
797	0x31D	RxPDO2, AO, Nd.29	1353	0x549	RxPDO9*, Nd.9	1908	0x774	RxPDO11*, Nd.52
798	0x31E	RxPDO2, AO, Nd.30	1354	0x54A	RxPDO9*, Nd.10	1909	0x775	RxPDO11*, Nd.53
799	0x31F	RxPDO2, AO, Nd.31	1355	0x54B	RxPDO9*, Nd.11	1910	0x776	RxPDO11*, Nd.54
800	0x320	RxPDO2, AO, Nd.32	1356	0x54C	RxPDO9*, Nd.12	1911	0x777	RxPDO11*, Nd.55
801	0x321	RxPDO2, AO, Nd.33	1357	0x54D	RxPDO9*, Nd.13	1912	0x778	RxPDO11*, Nd.56
802	0x322	RxPDO2, AO, Nd.34	1358	0x54E	RxPDO9*, Nd.14	1913	0x779	RxPDO11*, Nd.57
803	0x323	RxPDO2, AO, Nd.35	1359	0x54F	RxPDO9*, Nd.15	1914	0x77A	RxPDO11*, Nd.58
804	0x324	RxPDO2, AO, Nd.36	1360	0x550	RxPDO9*, Nd.16	1915	0x77B	RxPDO11*, Nd.59
805	0x325	RxPDO2, AO, Nd.37	1361	0x551	RxPDO9*, Nd.17	1916	0x77C	RxPDO11*, Nd.60
806	0x326	RxPDO2, AO, Nd.38	1362	0x552	RxPDO9*, Nd.18	1917	0x77D	RxPDO11*, Nd.61
807	0x327	RxPDO2, AO, Nd.39	1363	0x553	RxPDO9*, Nd.19	1918	0x77E	RxPDO11*, Nd.62
808	0x328	RxPDO2, AO, Nd.40	1364	0x554	RxPDO9*, Nd.20	1919	0x77F	RxPDO11*, Nd.63
809	0x329	RxPDO2, AO, Nd.41	1365	0x555	RxPDO9*, Nd.21	1921	0x781	RxPDO5*, Nd.1
810	0x32A	RxPDO2, AO, Nd.42	1366	0x556	RxPDO9*, Nd.22	1922	0x782	RxPDO5*, Nd.2
811	0x32B	RxPDO2, AO, Nd.43	1367	0x557	RxPDO9*, Nd.23	1923	0x783	RxPDO5*, Nd.3
812	0x32C	RxPDO2, AO, Nd.44	1368	0x558	RxPDO9*, Nd.24	1924	0x784	RxPDO5*, Nd.4
813	0x32D	RxPDO2, AO, Nd.45	1369	0x559	RxPDO9*, Nd.25	1925	0x785	RxPDO5*, Nd.5
814	0x32E	RxPDO2, AO, Nd.46	1370	0x55A	RxPDO9*, Nd.26	1926	0x786	RxPDO5*, Nd.6
815	0x32F	RxPDO2, AO, Nd.47	1371	0x55B	RxPDO9*, Nd.27	1927	0x787	RxPDO5*, Nd.7
816	0x330	RxPDO2, AO, Nd.48	1372	0x55C	RxPDO9*, Nd.28	1928	0x788	RxPDO5*, Nd.8
817	0x331	RxPDO2, AO, Nd.49	1373	0x55D	RxPDO9*, Nd.29	1929	0x789	RxPDO5*, Nd.9
818	0x332	RxPDO2, AO, Nd.50	1374	0x55E	RxPDO9*, Nd.30	1930	0x78A	RxPDO5*, Nd.10
819	0x333	RxPDO2, AO, Nd.51	1375	0x55F	RxPDO9*, Nd.31	1931	0x78B	RxPDO5*, Nd.11

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
820	0x334	RxPDO2, AO, Nd.52	1376	0x560	RxPDO9*, Nd.32	1932	0x78C	RxPDO5*, Nd.12
821	0x335	RxPDO2, AO, Nd.53	1377	0x561	RxPDO9*, Nd.33	1933	0x78D	RxPDO5*, Nd.13
822	0x336	RxPDO2, AO, Nd.54	1378	0x562	RxPDO9*, Nd.34	1934	0x78E	RxPDO5*, Nd.14
823	0x337	RxPDO2, AO, Nd.55	1379	0x563	RxPDO9*, Nd.35	1935	0x78F	RxPDO5*, Nd.15
824	0x338	RxPDO2, AO, Nd.56	1380	0x564	RxPDO9*, Nd.36	1936	0x790	RxPDO5*, Nd.16
825	0x339	RxPDO2, AO, Nd.57	1381	0x565	RxPDO9*, Nd.37	1937	0x791	RxPDO5*, Nd.17
826	0x33A	RxPDO2, AO, Nd.58	1382	0x566	RxPDO9*, Nd.38	1938	0x792	RxPDO5*, Nd.18
827	0x33B	RxPDO2, AO, Nd.59	1383	0x567	RxPDO9*, Nd.39	1939	0x793	RxPDO5*, Nd.19
828	0x33C	RxPDO2, AO, Nd.60	1384	0x568	RxPDO9*, Nd.40	1940	0x794	RxPDO5*, Nd.20
829	0x33D	RxPDO2, AO, Nd.61	1385	0x569	RxPDO9*, Nd.41	1941	0x795	RxPDO5*, Nd.21
830	0x33E	RxPDO2, AO, Nd.62	1386	0x56A	RxPDO9*, Nd.42	1942	0x796	RxPDO5*, Nd.22
831	0x33F	RxPDO2, AO, Nd.63	1387	0x56B	RxPDO9*, Nd.43	1943	0x797	RxPDO5*, Nd.23
833	0x341	RxPDO7*, Nd.1	1388	0x56C	RxPDO9*, Nd.44	1944	0x798	RxPDO5*, Nd.24
834	0x342	RxPDO7*, Nd.2	1389	0x56D	RxPDO9*, Nd.45	1945	0x799	RxPDO5*, Nd.25
835	0x343	RxPDO7*, Nd.3	1390	0x56E	RxPDO9*, Nd.46	1946	0x79A	RxPDO5*, Nd.26
836	0x344	RxPDO7*, Nd.4	1391	0x56F	RxPDO9*, Nd.47	1947	0x79B	RxPDO5*, Nd.27
837	0x345	RxPDO7*, Nd.5	1392	0x570	RxPDO9*, Nd.48	1948	0x79C	RxPDO5*, Nd.28
838	0x346	RxPDO7*, Nd.6	1393	0x571	RxPDO9*, Nd.49	1949	0x79D	RxPDO5*, Nd.29
839	0x347	RxPDO7*, Nd.7	1394	0x572	RxPDO9*, Nd.50	1950	0x79E	RxPDO5*, Nd.30
840	0x348	RxPDO7*, Nd.8	1395	0x573	RxPDO9*, Nd.51	1951	0x79F	RxPDO5*, Nd.31
841	0x349	RxPDO7*, Nd.9	1396	0x574	RxPDO9*, Nd.52	1952	0x7A0	RxPDO5*, Nd.32
842	0x34A	RxPDO7*, Nd.10	1397	0x575	RxPDO9*, Nd.53	1953	0x7A1	RxPDO5*, Nd.33
843	0x34B	RxPDO7*, Nd.11	1398	0x576	RxPDO9*, Nd.54	1954	0x7A2	RxPDO5*, Nd.34
844	0x34C	RxPDO7*, Nd.12	1399	0x577	RxPDO9*, Nd.55	1955	0x7A3	RxPDO5*, Nd.35
845	0x34D	RxPDO7*, Nd.13	1400	0x578	RxPDO9*, Nd.56	1956	0x7A4	RxPDO5*, Nd.36
846	0x34E	RxPDO7*, Nd.14	1401	0x579	RxPDO9*, Nd.57	1957	0x7A5	RxPDO5*, Nd.37

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
847	0x34F	RxPDO7*, Nd.15	1402	0x57A	RxPDO9*, Nd.58	1958	0x7A6	RxPDO5*, Nd.38
848	0x350	RxPDO7*, Nd.16	1403	0x57B	RxPDO9*, Nd.59	1959	0x7A7	RxPDO5*, Nd.39
849	0x351	RxPDO7*, Nd.17	1404	0x57C	RxPDO9*, Nd.60	1960	0x7A8	RxPDO5*, Nd.40
850	0x352	RxPDO7*, Nd.18	1405	0x57D	RxPDO9*, Nd.61	1961	0x7A9	RxPDO5*, Nd.41
851	0x353	RxPDO7*, Nd.19	1406	0x57E	RxPDO9*, Nd.62	1962	0x7AA	RxPDO5*, Nd.42
852	0x354	RxPDO7*, Nd.20	1407	0x57F	RxPDO9*, Nd.63	1963	0x7AB	RxPDO5*, Nd.43
853	0x355	RxPDO7*, Nd.21	1409	0x581	SDO Tx Nd.1	1964	0x7AC	RxPDO5*, Nd.44
854	0x356	RxPDO7*, Nd.22	1410	0x582	SDO Tx Nd.2	1965	0x7AD	RxPDO5*, Nd.45
855	0x357	RxPDO7*, Nd.23	1411	0x583	SDO Tx Nd.3	1966	0x7AE	RxPDO5*, Nd.46
856	0x358	RxPDO7*, Nd.24	1412	0x584	SDO Tx Nd.4	1967	0x7AF	RxPDO5*, Nd.47
857	0x359	RxPDO7*, Nd.25	1413	0x585	SDO Tx Nd.5	1968	0x7B0	RxPDO5*, Nd.48
858	0x35A	RxPDO7*, Nd.26	1414	0x586	SDO Tx Nd.6	1969	0x7B1	RxPDO5*, Nd.49
859	0x35B	RxPDO7*, Nd.27	1415	0x587	SDO Tx Nd.7	1970	0x7B2	RxPDO5*, Nd.50
860	0x35C	RxPDO7*, Nd.28	1416	0x588	SDO Tx Nd.8	1971	0x7B3	RxPDO5*, Nd.51
861	0x35D	RxPDO7*, Nd.29	1417	0x589	SDO Tx Nd.9	1972	0x7B4	RxPDO5*, Nd.52
862	0x35E	RxPDO7*, Nd.30	1418	0x58A	SDO Tx Nd.10	1973	0x7B5	RxPDO5*, Nd.53
863	0x35F	RxPDO7*, Nd.31	1419	0x58B	SDO Tx Nd.11	1974	0x7B6	RxPDO5*, Nd.54
864	0x360	RxPDO7*, Nd.32	1420	0x58C	SDO Tx Nd.12	1975	0x7B7	RxPDO5*, Nd.55
865	0x361	RxPDO7*, Nd.33	1421	0x58D	SDO Tx Nd.13	1976	0x7B8	RxPDO5*, Nd.56
866	0x362	RxPDO7*, Nd.34	1422	0x58E	SDO Tx Nd.14	1977	0x7B9	RxPDO5*, Nd.57
867	0x363	RxPDO7*, Nd.35	1423	0x58F	SDO Tx Nd.15	1978	0x7BA	RxPDO5*, Nd.58
868	0x364	RxPDO7*, Nd.36	1424	0x590	SDO Tx Nd.16	1979	0x7BB	RxPDO5*, Nd.59
869	0x365	RxPDO7*, Nd.37	1425	0x591	SDO Tx Nd.17	1980	0x7BC	RxPDO5*, Nd.60
870	0x366	RxPDO7*, Nd.38	1426	0x592	SDO Tx Nd.18	1981	0x7BD	RxPDO5*, Nd.61
871	0x367	RxPDO7*, Nd.39	1427	0x593	SDO Tx Nd.19	1982	0x7BE	RxPDO5*, Nd.62
872	0x368	RxPDO7*, Nd.40	1428	0x594	SDO Tx Nd.20	1983	0x7BF	RxPDO5*, Nd.63

dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ	dez	hex	Tele-gramm-typ
873	0x369	RxPDO7*, Nd.41	1429	0x595	SDO Tx Nd.21		0x	

## 6.3 CANopen Baudrate und Bit Timing

### Bit Timing

Folgende Baudraten und Bittiming Register Einstellungen werden von den Beckhoff CANopen Geräten unterstützt:

Baudrate [kBaud]	BTR0	BTR1	Sampling Point
1000	0x00	0x14	75%
800	0x00	0x16	80%
500	0x00	0x1C	87%
250	0x01	0x1C	87%
125	0x03	0x1C	87%
100	0x04	0x1C	87%
50	0x09	0x1C	87%
20	0x18	0x1C	87%
10	0x31	0x1C	87%

Die angegebenen Bit-Timing Register Einstellungen (BTR0, BTR1) gelten z.B. für die CAN-Controller Philips 82C200, SJA1000, Intel 80C527, Siemens 80C167, und andere. Sie sind für maximale Buslänge optimiert.

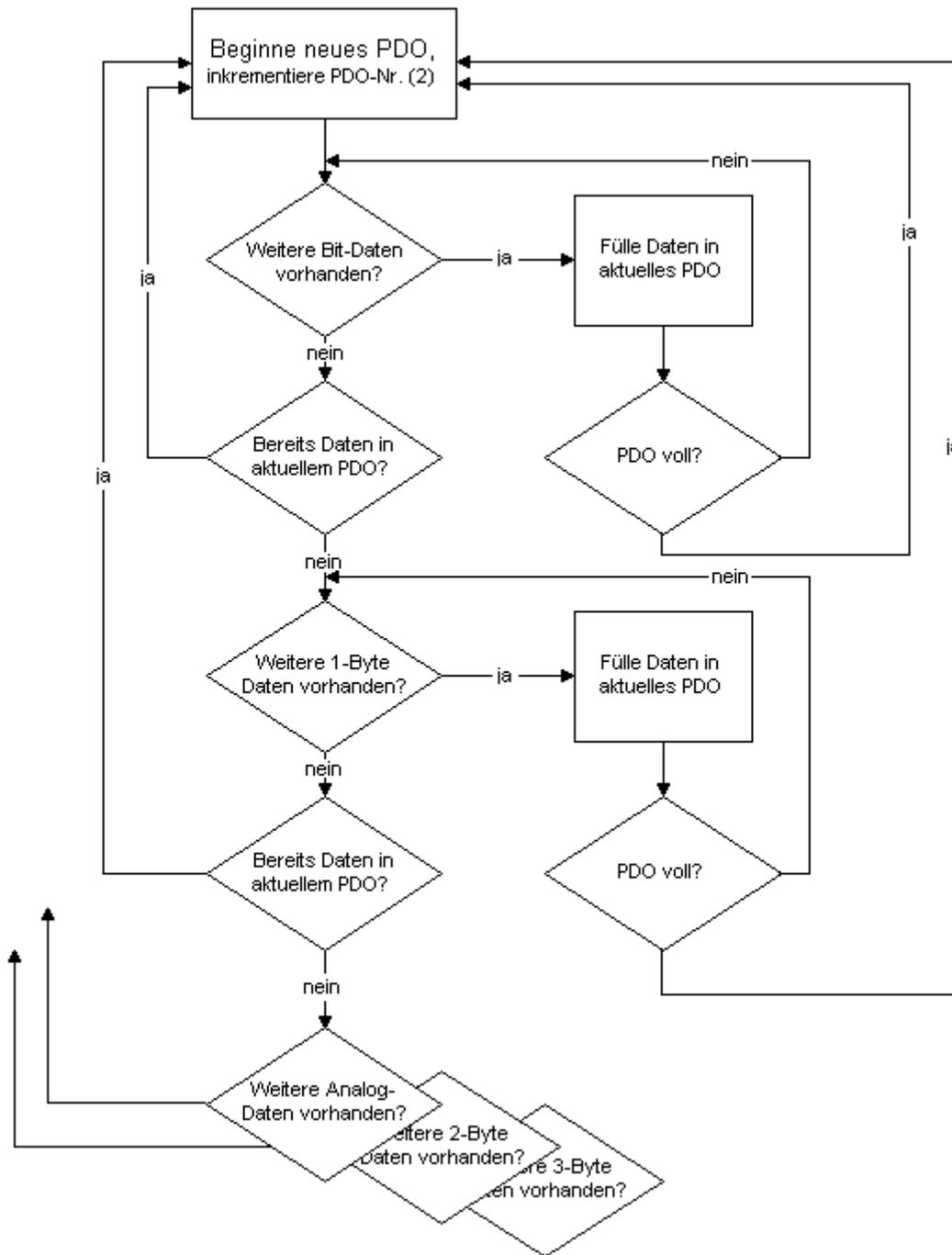
## 6.4 Automatisches PDO-Mapping

### **BK51x0, IL23x0-B510**

PDO1 und PDO2 werden mit digitalen und analogen Prozessdaten belegt. Für jedes weitere PDO geht der CANopen-Knoten nach dem im untenstehenden Ablaufdiagramm gezeigten Verfahren vor und belegt die PDOs in der folgenden Reihenfolge mit Prozessdaten:

1. Digitale E/As (wenn mehr als 64 vorhanden sind)
2. 1- Byte Sonderklemmen
3. Analoge E/As
4. 2- Byte Sonderklemmen
5. 3- Byte Sonderklemmen
6. ...10. 8- Byte Sonderklemmen

Datentypen werden nicht gemischt! Es wird für jeden neuen Datentyp ein neues PDO befüllt ([Beispiel \[▶\\_163\]](#) siehe unten).



**Beispiel**

Beispiel

Ein BK5120 (CANopen-Koppler) hat

- 78 digitale Ein- und 48 digitale Ausgänge
- 6 analoge Eingänge und 4 analoge Ausgänge
- eine KL5001 (SSI-Geber Interface: per default 4 Byte Eingänge)
- eine KL6001 (serielle Schnittstelle: per default 4 Byte Eingänge und 4 Byte Ausgänge)
- eine KL5111 (Inkrementalkoder-Interface) (6 Byte Eingänge und 6 Byte Ausgänge)
- eine KL6201 AS-i Masterklemme mit Default-Einstellung (22 Byte Prozessdateninterface)

PDO	Dateninhalt (Mapping)	Objektverzeichnis	PDO	Dateninhalt (Mapping)	Objektverzeichnis
RxPDO1	5 Bytes digitale Ausgänge 1..48	0x6200, SI 1..5	TxPDO1	8 Bytes digitale Eingänge 1..64	0x6000, SI 1..8
RxPDO2	8 Bytes analoge Ausgänge 1..4	0x6411, SI 1..4	TxPDO2	4 Bytes analoge Eingänge 1..4	0x6401, SI 1..4
RxPDO3	4 Bytes serielle Schnittstelle	0x2900, SI 1	TxPDO3	2 Bytes digitale Eingänge 65...78	0x6000, SI 9..10
RxPDO4	6 Byte Encoder Ausgänge	0x2D00, SI 1	TxPDO4	analoge Eingänge 5 und 6	0x6401, SI 5..6
RxPDO5	8 Byte ASI Master 1: Parameterdaten-Block	0x3100, SI 1	TxPDO5	8 Bytes: 4 Bytes SSI und 4 Bytes serielle Schnittstelle	0x2800, SI 1..2
RxPDO6	8 Byte ASI Master 1: Prozessdaten-Block Outputs ASI Slave 1..15	0x3100, SI 2	TxPDO6	6 Byte Encoder Ein	0x2C00, SI 1
RxPDO7	8 Byte ASI Master 1: Prozessdaten-Block Outputs ASI Slave 16...31	0x3100, SI 3	TxPDO7	8 Byte ASI Master 1: Parameterdaten-Block	0x3000, SI 1
			TxPDO8	8 Byte ASI Master 1: Prozessdaten-Block Inputs ASI Slave 1..15	0x3000, SI 2
			TxPDO9	8 Byte ASI Master 1: Prozessdaten-Block Inputs ASI Slave 16...31	0x3000, SI 3

## 6.5 Allgemeine Betriebsbedingungen

### Schutzarten nach IP-Code

In der Norm IEC 60529 (DIN EN 60529) sind die Schutzgrade festgelegt und nach verschiedenen Klassen eingeteilt. Die Bezeichnung erfolgt in nachstehender Weise.

1. Ziffer: Staub- und Berührungsschutz	Bedeutung
0	Nicht geschützt
1	Geschützt gegen den Zugang zu gefährlichen Teilen mit dem Handrücken. Geschützt gegen feste Fremdkörper Ø50 mm
2	Geschützt gegen den Zugang zu gefährlichen Teilen mit einem Finger. Geschützt gegen feste Fremdkörper Ø12,5 mm
3	Geschützt gegen den Zugang zu gefährlichen Teilen mit einem Werkzeug. Geschützt gegen feste Fremdkörper Ø2,5 mm
4	Geschützt gegen den Zugang zu gefährlichen Teilen mit einem Draht. Geschützt gegen feste Fremdkörper Ø1 mm
5	Geschützt gegen den Zugang zu gefährlichen Teilen mit einem Draht. Staubgeschützt. Eindringen von Staub ist nicht vollständig verhindert, aber der Staub darf nicht in einer solchen Menge eindringen, dass das zufriedenstellende Arbeiten des Gerätes oder die Sicherheit beeinträchtigt wird
6	Geschützt gegen den Zugang zu gefährlichen Teilen mit einem Draht. Staubdicht. Kein Eindringen von Staub

2. Ziffer: Wasserschutz*	Bedeutung
0	Nicht geschützt
1	Geschützt gegen Tropfwasser
2	Geschützt gegen Tropfwasser, wenn das Gehäuse bis zu 15° geneigt ist
3	Geschützt gegen Sprühwasser. Wasser, das in einem Winkel bis zu 60° beiderseits der Senkrechten gesprüht wird, darf keine schädliche Wirkung haben
4	Geschützt gegen Spritzwasser. Wasser, das aus jeder Richtung gegen das Gehäuse spritzt, darf keine schädlichen Wirkungen haben
5	Geschützt gegen Strahlwasser.
6	Geschützt gegen starkes Strahlwasser.
7	Geschützt gegen die Wirkungen beim zeitweiligen Untertauchen in Wasser. Wasser darf nicht in einer Menge eintreten, die schädliche Wirkungen verursacht, wenn das Gehäuse für 30 Minuten in 1 m Tiefe in Wasser untergetaucht ist

\*) In diesen Schutzklassen wird nur der Schutz gegen Wasser definiert.

### Chemische Beständigkeit

Die Beständigkeit bezieht sich auf das Gehäuse der Feldbus Box und den verwendeten Metallteilen.

<b>Art</b>	<b>Beständigkeit</b>
Wasserdampf	bei Temperaturen >100°C nicht beständig
Natriumlauge (ph-Wert > 12)	bei Raumtemperatur beständig > 40°C unbeständig
Essigsäure	unbeständig
Argon (technisch rein)	beständig

**Legende**

beständig: Lebensdauer mehrere Monate

bedingt beständig: Lebensdauer mehrere Wochen

unbeständig: Lebensdauer mehrere Stunden bzw. baldige Zersetzung

## **6.6 Zulassungen**

### **Zulassungen**

UL E172151

### **Konformitätskennzeichnung**

CE

### **Schutzart**

IP65/66/67 gemäß EN60529

## **6.7 Prüfnormen für die Geräteprüfung**

### **EMV**

Festigkeit: EN 61000-6-2

Aussendung: EN 61000-6-4

### **Vibrationsfestigkeit**

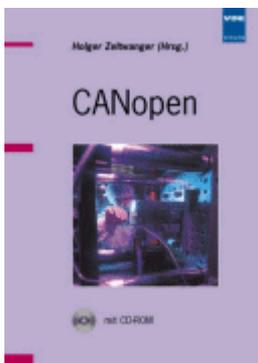
Schwingungsprüfung: EN 60068-2-2, Amplitude 2 g (Norm 1 g)

Schockprüfung: EN 60068-2-27, Schockanzahl 1000 (Norm 2)

## 6.8 Literaturverzeichnis

### Deutsche Bücher

- Holger Zeltwanger (Hrsg.):  
**CANopen**,  
VDE Verlag, 2001. 197 Seiten,  
ISBN 3-800-72448-0



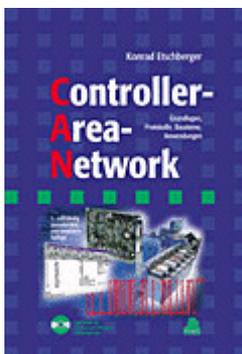
- Konrad Etschberger:  
**Controller Area Network**, Grundlagen, Protokolle, Bausteine, Anwendungen.  
Hanser Verlag, 2000. 431 Seiten.  
ISBN 3-446-19431-2

### Feldbustechnik allgemein

- Gerhard Gruhler (Hrsg.):  
**Feldbusse und Geräte-Kommunikationssysteme**, Praktisches Know-How mit Vergleichsmöglichkeiten.  
Franzis Verlag 2001. 244 Seiten.  
ISBN 3-7723-5745-8

### Englische Bücher

- Konrad Etschberger:  
**Controller Area Network**,  
Ixxat Press, 2001. 440 Seiten.  
ISBN 3-00-007376-0
- M. Farsi, M. Barbosa:  
**CANopen Implementation**,  
RSP 2000. 210 Seiten.  
ISBN 0-86380-247-8



### Standards

- ISO 11898:  
Road Vehicles - Interchange of digital information - Controller Area Network (CAN) for high speed communication.

- CiA DS 301:  
CANopen Application Layer and Communication Profile.  
Erhältlich beim Verband CAN in Automation.
- CiA DS 401:  
CANopen Device Profile for Generic E/A Modules.  
Erhältlich beim Verband CAN in Automation.

## 6.9 Abkürzungsverzeichnis

### **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.

### **NMT**

Network Management. Eines der Dienstelemente der CANopen-Spezifikation. Das Netzwerkmanagement dient zur Netzwerkinitialisierung und zur Knotenüberwachung.

### **PDO**

Process Data Object oder Prozessdatenobjekt. CAN-Telegramm zur Übertragung von Prozessdaten (z.B. E/A-Daten).

### **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.

### **SDO**

Service Data Object oder Servicedatenobjekt. CAN-Telegramm mit Protokoll zur Kommunikation mit Daten des Objektverzeichnisses (typisch Parameterdaten).

### **TxPDO**

Sende-PDO (aus Sicht des CAN-Knotens bezeichnet).

## 6.10 Support und Service

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

### Beckhoff Niederlassungen und Vertretungen

Wenden Sie sich bitte an Ihre Beckhoff Niederlassung oder Ihre Vertretung für den lokalen Support und Service zu Beckhoff Produkten!

Die Adressen der weltweiten Beckhoff Niederlassungen und Vertretungen entnehmen Sie bitte unseren Internetseiten: <https://www.beckhoff.de>

Dort finden Sie auch weitere Dokumentationen zu Beckhoff Komponenten.

### Beckhoff Support

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

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

Hotline: +49(0)5246 963 157  
Fax: +49(0)5246 963 9157  
E-Mail: [support@beckhoff.com](mailto:support@beckhoff.com)

### Beckhoff Service

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

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

Hotline: +49(0)5246 963 460  
Fax: +49(0)5246 963 479  
E-Mail: [service@beckhoff.com](mailto:service@beckhoff.com)

### Beckhoff Firmenzentrale

Beckhoff Automation GmbH & Co. KG

Hülshorstweg 20  
33415 Verl  
Deutschland

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



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