

Programmierhandbuch

Sprache **Deutsch**
Original
Dokument-Nr. 5.07017.03
Artikel-Nr. 413208
Stand 14.08.2014

be in motion

be in motion



CoE

**CANopen over EtherCAT
b maXX[®] Regler**

D	5.07017.03
----------	------------

Vor Beginn aller Arbeiten Betriebsanleitung lesen!

Copyright	<p>Diese Programmierhandbuch darf vom Eigentümer ausschließlich für den internen Gebrauch in beliebiger Anzahl kopiert werden. Für andere Zwecke darf diese Programmierhandbuch auch auszugsweise weder kopiert noch vervielfältigt werden.</p> <p>Verwertung und Mitteilung von Inhalten dieser Programmierhandbuch sind nicht gestattet. Bezeichnungen bzw. Unternehmenskennzeichen in dieser Programmierhandbuch können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.</p>
Vorabinformation	<p>Achtung: Sofern das Ihnen vorliegende Dokument als Vorabinformation gekennzeichnet ist, gilt Folgendes:</p> <p>Bei dieser Version handelt es sich um technische Vorabinformationen, die die Anwender der beschriebenen Geräte und Funktionen frühzeitig erhalten sollen, um sich auf mögliche Änderungen bzw. funktionale Erweiterungen einstellen zu können.</p> <p>Diese Informationen sind als vorläufig zu verstehen, da diese noch nicht dem endgültigen Baumüller internen Review-Prozess unterzogen wurden. Insbesondere unterliegen diese Informationen noch Änderungen, so dass keine rechtliche Verbindlichkeit auf Grund von diesen Vorabinformationen hergeleitet werden kann. Baumüller übernimmt keine Haftung für Schäden, die sich aus dieser unter Umständen fehlerhaften oder unvollständigen Version ergeben können.</p> <p>Sollten Sie inhaltliche und / oder gravierende formale Fehler in dieser Vorabinformation erkennen oder vermuten, so bitten wir Sie, sich an den für Sie zuständigen Betreuer der Firma Baumüller zu wenden und uns über diese Mitarbeiter Ihre Erkenntnisse und Anmerkungen zukommen zu lassen, so dass Ihre Erkenntnisse und Anmerkungen beim Übergang von den Vorabinformationen zu den endgültigen (durch Baumüller gereviewten) Informationen berücksichtigt und ggf. eingepflegt werden können.</p> <p>Die im nachfolgenden Abschnitt unter „Verbindlichkeit“ genannten Bedingungen sind im Falle von Vorabinformationen ungültig.</p>
Verbindlichkeit	<p>Diese Programmierhandbuch ist Teil des Gerätes/der Maschine. Diese Programmierhandbuch muss jederzeit für den Bediener zugänglich und in einem leserlichen Zustand sein. Bei Verkauf/Verlagerung des Gerätes/der Maschine muss diese Programmierhandbuch vom Besitzer zusammen mit dem Gerät/der Maschine weitergegeben werden.</p> <p>Nach Verkauf des Gerätes/der Maschine sind dieses Original und sämtliche Kopien an den Käufer zu übergeben. Nach Entsorgung oder anderem Nutzungsende sind dieses Original und sämtliche Kopien zu vernichten.</p> <p>Mit der Übergabe der vorliegenden Programmierhandbuch werden entsprechende Betriebsanleitungen mit einem früheren Stand außer Kraft gesetzt.</p> <p>Bitte beachten Sie, dass Angaben/Zahlen/Informationen aktuelle Werte zum Druckdatum sind. Zur Ausmessung, Berechnung und Kalkulation sind diese Angaben nicht rechtlich verbindlich.</p> <p>Die Firma Baumüller Nürnberg GmbH behält sich vor, im Rahmen der eigenen Weiterentwicklung der Produkte die technischen Daten und die Handhabung von Baumüller-Produkten zu ändern.</p> <p>Es kann jedoch keine Gewährleistung bezüglich der Fehlerfreiheit dieser Programmierhandbuch, soweit nicht in den Allgemeinen Verkaufs- und Lieferbedingungen anders beschrieben, übernommen werden.</p>

© **Baumüller Nürnberg GmbH**

Ostendstr. 80 - 90
90482 Nürnberg
Deutschland

Tel. +49 9 11 54 32 - 0
Fax: +49 9 11 54 32 - 1 30

E-Mail: mail@baumueller.de
Internet: www.baumueller.de



Inhaltsverzeichnis

1	Einleitung	5
1.1	Allgemeines	5
1.2	Montage und Installation	5
1.3	EDS-Datei	6
1.4	XML-Datei	6
2	Grundlegende Sicherheitshinweise	7
2.1	Gefahrenhinweise und Gebote	7
2.2	Infozeichen	7
3	Grundlagen EtherCAT	9
3.1	Literatur zum Thema EtherCAT und CANopen	9
3.2	Grundlagen EtherCAT	10
3.2.1	Topologie Daten	10
3.2.2	Rahmenaufbau	11
3.2.3	Geräteprofile	12
3.2.4	EtherCAT Frame-Struktur	12
3.2.5	EtherCAT Kommunikationszustände	15
3.2.6	Ethernet over EtherCAT (EoE) - TCP/IP- Tunneling über EtherCAT	18
3.3	Grundlagen CANopen	19
3.3.1	Objektverzeichnis	20
3.3.2	Zustandsmaschine CANopen	21
3.4	Unterstützte Betriebsarten des Geräte Profils DSP 402	21
3.4.1	Kurze Übersicht	21
3.4.2	Betriebsarten und Feldbus-Objekte	22
4	Kommunikation zum b maXX[®] Regler	25
4.1	Kommunikationsablauf	25
4.2	Parametrierung der BACI Kommunikationszeiten	26
4.3	Konfigurationsmöglichkeiten der CoE-Optionskarte in WinBASS II / ProDrive	27
4.3.1	Einstellungen ab Firmware Version FW 03.00 b maXX [®] -Regler	27
4.3.2	EMCY-Errorcode	27
4.4	Allgemeine Bemerkung zur CoE-Optionskarte	28
4.4.1	Applikationsparameter	29
4.4.2	Geschwindigkeitsprofil bei der Positionierung (FBO 6086 _{hex})	29
4.4.3	Einstellbares Verhalten, wenn neues Ziel außerhalb der Software-Endschalter	29
4.4.4	Fehlerrückmeldung bei Fahrt in Hardware-Endschalter	29
4.4.5	Benutzer Einheiten BE	30
4.4.6	Getriebefaktor	31
4.4.7	CANopen Offset	32
4.4.8	Referenzfahrt für Positionierung erforderlich	32
4.4.9	Varianten der Positionierung, Zielposition abhängig vom Positioniermodus (P0601)	33
5	Datenaustausch und Parametrierung	35
5.1	Verzeichnis der Objekte zur Kommunikationssteuerung	35
5.2	Bedarfsdaten (SDO)	40
5.2.1	Telegrammaufbau nach CANopen	41
5.2.2	Arten des SDO-Transfers	42
5.2.3	Fehlerreaktionen	43
5.3	Prozessdaten	44



Inhaltsverzeichnis

5.3.1	PDO-Mapping	44
5.3.2	Synchronisation (SYNC)	47
5.3.3	Beispiel für das PDO-Mapping	49
5.3.4	Eintrag in die BACI	52
5.4	Fehlertelegramm (EMCY)	53
5.4.1	Telegrammaufbau	53
5.4.2	Umsetzung der Fehlermeldungen nach DSP 402 V1.1	54
Anhang A - Abkürzungen		67
Anhang B - Kurzreferenz		69
B.1	4000er Objektnummern (Herstellerspezifische Objekte)	69
B.2	6000er Objektnummern (Device Profile DSP 402)	70
Anhang C - Umsetzungstabellen		75
Anhang D - Technische Daten		93
D.1	CoE-Optionskarte: Technische Merkmale	93
D.2	CANopen Optionskarte: Datenkanäle zum b maXX®-Regler	93
Abbildungsverzeichnis		95
Stichwortverzeichnis		97
Revisionsübersicht		99

1

EINLEITUNG

Das Programmierhandbuch ist ein wichtiger Bestandteil Ihres b maXX[®] 4400 Gerätes. Lesen Sie daher nicht zuletzt im Interesse Ihrer eigenen Sicherheit diese Dokumentation vollständig durch. In dieser Dokumentation erfahren Sie, wie die Fa. Baumüller Nürnberg GmbH die Optionsmodul **CANopen over EtherCAT (CoE)** - Anschaltung auf dem CoE-Slave für die Gerätereihe b maXX[®] 4400 realisiert hat.

Die Einleitung beinhaltet allgemeine Informationen zum Optionsmodul CoE-Slave.

1.1 Allgemeines

Das CoE-Optionsmodul verbindet das b maXX[®] 4400 über den Ethernet-Bus mit anderen CoE-Knoten (z. B. PC, SPS, weitere b maXX, I/O-Module).

Informationen zu Options- und Funktionsmodulen für die Gerätereihe b maXX[®] finden Sie in der Dokumentation 5.01040.

Informationen zur Programmierung des b maXX[®] 4400 Reglers finden Sie im Parameterhandbuch 5.03039.

1.2 Montage und Installation

Die Montage des Optionsmoduls CoE-Slave beschreiben wir in der Dokumentation 5.02014.

1.3 EDS-Datei

Die EDS-Datei ist eine ASCII-Datei und dient der Beschreibung des Funktionsumfangs eines CoE-Gerätes. Es ist ein elektronisches Datenblatt des CoE-Gerätes. Die EDS-Datei wird von CoE-Mastern bzw. Buskonfiguratoren genutzt. Die EDS-Datei beinhaltet Informationen über alle vom Slave unterstützten Objekte und weitere Merkmale.

Der Namensuffix der EDS-Datei ist *.eds.

1.4 XML-Datei

In der XML-Datei sind Informationen enthalten, die ein Master braucht, um z.B. die FMMU und die SyncManager auf dem CoE-Slave zu konfigurieren.

GRUNDLEGENDE SICHERHEITSHINWEISE

In diesem Kapitel beschreiben wir Gefahren, die beim Parametrieren des Baumüller b maXX[®] 4400-Reglerteils auftreten können und wir erklären die Bedeutung des Infozeichens.

2.1 Gefahrenhinweise und Gebote



WARNUNG!

Gefahr durch Änderung von Parametereinstellungen!

Die Gefahr ist: **mechanische und elektrische Einwirkung.**

Die Änderung von Parametern beeinflusst das Verhalten des Baumüller-Geräts und somit das Verhalten der Anlage und ihrer Komponenten. Wenn die Einstellungen der Parameter verändert werden, kann ein gefährliches Verhalten der Anlage und/oder ihrer Komponenten auftreten.

Deshalb:

- Nach jeder Änderung der Parametereinstellungen ist eine Inbetriebnahme unter Beachtung aller Sicherheitshinweise und Sicherheitsvorschriften durchzuführen.

2.2 Infozeichen



HINWEIS!

Der hier angegebene Hinweis ist eine besonders wichtige Information.

GRUNDLAGEN ETHERCAT

3.1 Literatur zum Thema EtherCAT und CANopen

Für grundlegende Informationen zum Thema EtherCAT und CANopen empfehlen wir folgende Literatur:

- [1]
„EtherCAT Technology Group“
<http://www.ethercat.org/>
- [2]
EtherCAT communication specification,
Version 1.0 provided by
the EtherCAT Technology Group
- [3]
EtherCAT Slave Controller ESC10/20 Hardware Data Sheet,
Version 1.1 provided by
the EtherCAT Technology Group
- [4]
Online Encyclopedia Wikipedia <http://www.wikipedia.org/>
- [5]
Zeltwanger
CANopen
Holger Zeltwanger VDE-Verlag
- [6]
www.can-cia.de
CAN in Automation e.V.
Kontumazgarten 3
D-90429 Nürnberg

3.2 Grundlagen EtherCAT

Die Real Time Ethernet Control Automation Technology (EtherCAT) wurde von der Fa. Beckhoff als neuer Feldbus Standard entwickelt. Um EtherCAT als offenen Standard verbreiten zu können, wurde zu diesem Zweck die Ethernet Technology Group ETG gegründet. Die ETG gilt als eine Vereinigung von Interessenten, Herstellern und Anwendern. Diese Vereinigung hatte im Dezember 2006: 421 Mitglieder aus 31 Ländern. Diese Mitglieder tragen in technischen Arbeitskreisen zur Weiterentwicklung der Technologie bei.

3.2.1 Topologie Daten

Es können mehrere Bustopologien genutzt werden, z. B. Linien-, Baum- oder Sterntopologie (► [Abbildung 1](#) ◀ auf Seite 10).

Bis zu 65535 Teilnehmer können erreicht werden, dadurch ist die Netzwerkausdehnung nahezu unbegrenzt (>500 km).

Für die Übertragung reicht ein Standard Ethernet Patch Kabel (CAT5). Es werden die Vollduplex-Eigenschaften von 100 BASE-TX vollständig ausgenutzt, so dass effektive Datenraten von >100 MBit/s (>90 % von 2 x 100 MBit/s) erreichbar sind. Die Leitungslänge zwischen zwei Teilnehmern wird mit bis zu 100 m angegeben.

Alternativ können auch Lichtwellenleiter-Varianten von 50 m bis 2000 m eingesetzt werden.

Vorteilhaft ist auch, dass während des Betriebes Geräte hinzugeführt bzw. entfernt werden können „Hot Connect / Disconnect von Bussegmenten“.

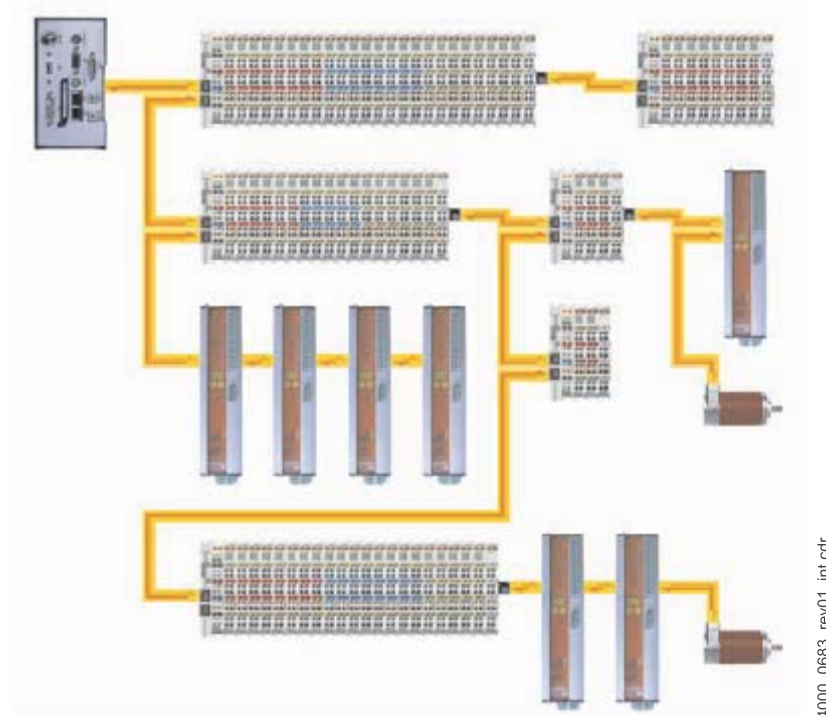


Abbildung 1: Flexible Topologie: Linie, Baum oder Stern [1]

3.2.2 Rahmenaufbau

Das EtherCAT-Protokoll ist speziell für die Prozessdaten optimiert worden. Dies gelingt Dank eines speziellen Ethertypes (88A4h), der direkt in einem Ethernet-Frame transportiert wird. Es kann aus mehreren Subtelegramms bestehen, die jeweils einen Speicherbereich des bis zu 4 Gigabyte grossen logischen Prozessabbildes bedienen. Das Adressieren der Daten kann wahlfrei erfolgen, dabei ist die Reihenfolge der physikalischen Reihenfolge unabhängig von der datentechnischen Reihenfolge der Teilnehmer im Netz.

Das Versenden erfolgt mit einem minimalen Versatz von wenigen Bitzeiten.

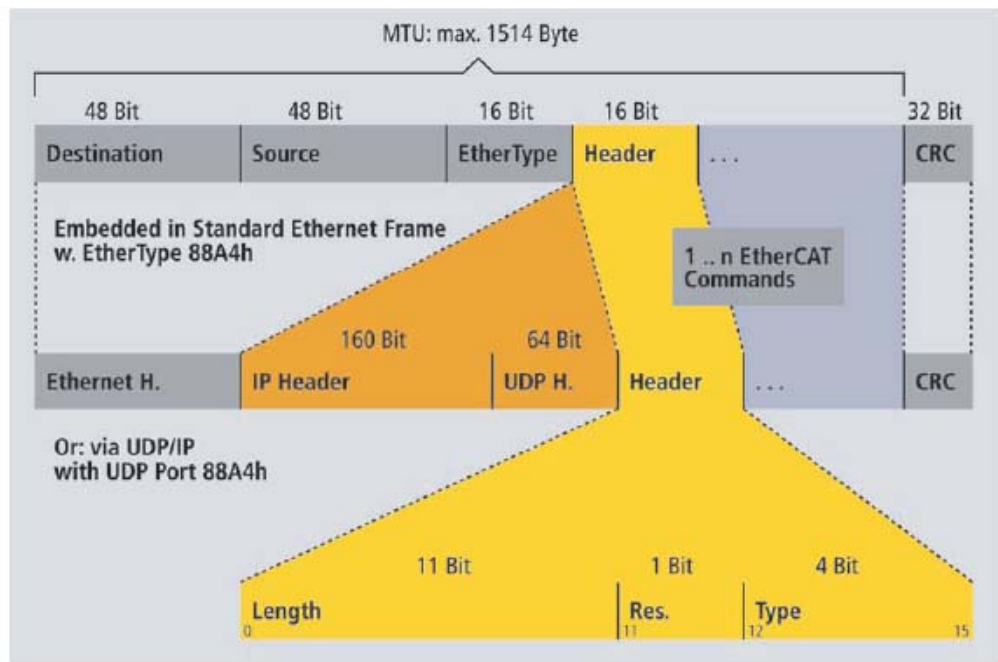


Abbildung 2: EtherCAT: Standard -IEEE 802.3-Frames [1]

3.2 Grundlagen EtherCAT

3.2.3 Geräteprofile

Für verschiedene Aufgaben in der Automatisierung gibt es speziell dafür zugeschnitten Feldbussysteme z.B. CANopen. Die Feldbussysteme sind oft in Standards zusammengefasst. Beim EtherCAT wird darauf verzichtet eigene Profile für schon vorhandene Standards zu entwickeln, vielmehr wird auf schon vorhandenes aufgebaut.

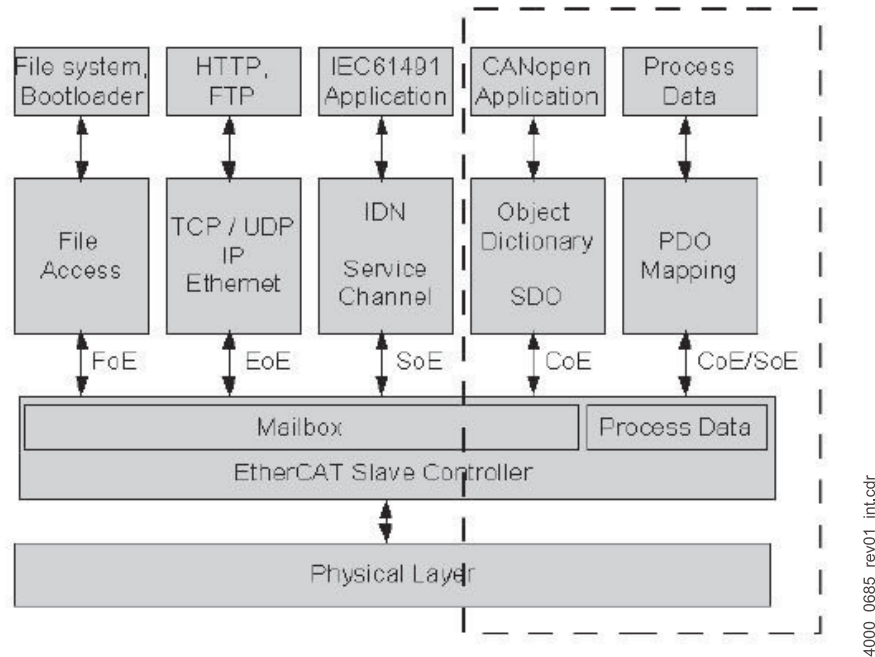


Abbildung 3: Device Profile bei EtherCAT[1]

3.2.4 EtherCAT Frame-Struktur

EtherCAT Telegramme werden, eingebettet in einem Ethernet Telegramm, mit diesen gesendet. Das Telegramm enthält einen Ethernet-Header (a), einen EtherCAT-Header (b) und folgend dann n EtherCAT-Telegramme.

Das EtherCAT-Telegramm (c) seinerseits ist aufgeteilt in einen EtherCAT-Header, Datenbereich und einen Counter Bereich.

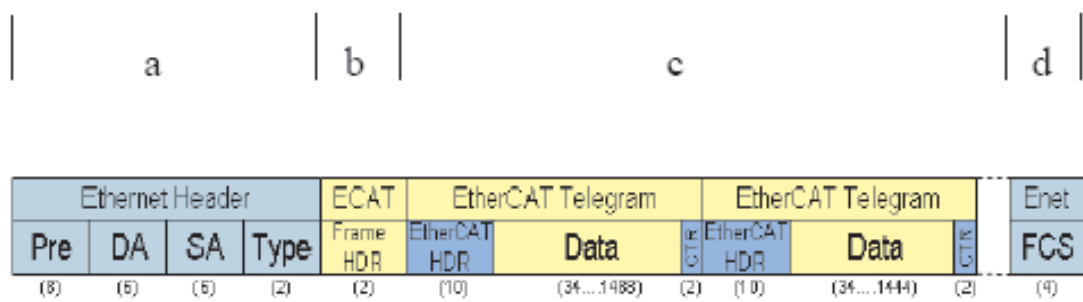


Abbildung 4: EtherCAT - Frame [1]

a) Ethernet Header:

- Pre** Die Präambel dient dem Empfänger zur Synchronisation und zur Lokalisierung, sie besteht aus einer Folge von '10101010' je Byte.
Im Präambel ist das SFD Byte enthalten: SFD: „start of frame delimiter“ kennzeichnet den Frame-Anfang; Bitmuster 10101011.
- DA** Destination MAC address.
- SA** Source MAC address.
Ziel-/Quelladresse: spezifizieren das empfangende (möglicherweise mehrere) und das zu sendende Ethernet Telegramm; innerhalb eines LAN nur einheitliche Längen erlaubt (16 oder 48 Bit)
- Type** Definiert den EtherType. Der EtherType zeigt an welches Protokoll der nächsthöheren Schicht* innerhalb der Nutzdaten benutzt wird. 88A4_{hex} definiert den EtherCAT Type.

* ISO-OSI-Schichtmodell

b) EtherCAT Frame Header:

Das EtherCAT Frame Header hat eine Länge von 2 Byte. Hier sind die Information über die Datenlänge und den Datentype des folgenden Telegramms enthalten.

c) EtherCAT-Telegramm:

Das EtherCAT -Telegram unterteilt sich in ein Telegramm-Header, den zu übertragenden Daten und dem Working Counter. Der Working Counter wird von jedem bearbeitendem Slave inkrementiert.

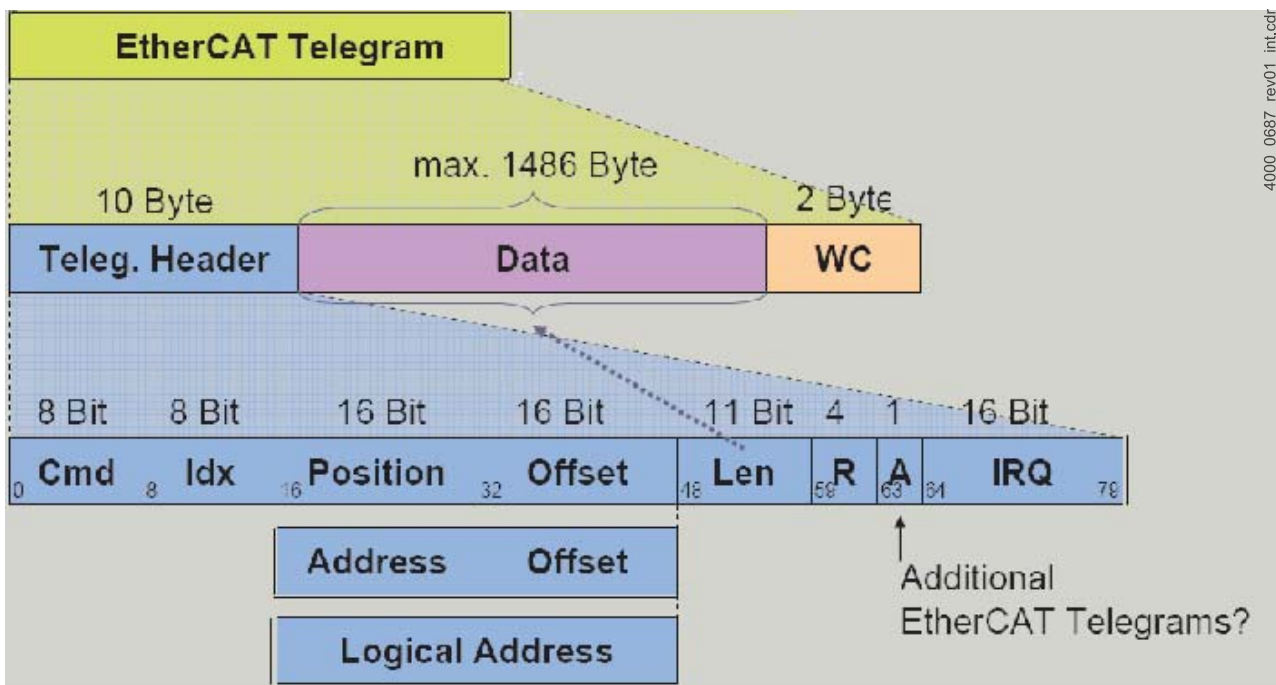


Abbildung 5: EtherCAT Telegramm [1]

Der „EtherCAT Telegram Header“ hat eine Länge von 10 Byte. Er enthält Informationen über die folgenden Daten.

- **CMD**, 1 Byte. Kodiert das vom Master abgesetzte EtherCAT Kommando, das entweder schreibend oder lesend gekennzeichnet sein kann.
- **IDX**, 1 Byte. Index des Frames. Wird vom Slave unverändert weitergegeben, damit kann der Master das Telegramm beim Empfang wieder leicht zuordnen.
- Die **Position**, zeigt die Adresse oder die physikalische Position des Slaves an. Zusätzlich wird ein Offset mit angegeben.

Aufgeteilt in:	ADP (2 Byte) Adress-Page	abhängig vom verwendeten Kommando
	ADO (2 Byte) Adress-Offset	abhängig vom verwendeten Kommando
	INT Interrupt-Feld	
- **LEN**, 2 Bytes.

In den Bits 0 bis 10 wird die Länge des nachfolgenden Datenblocks gespeichert. Die Bits 11 bis 15 dienen für verschiedene Zwecke als Flags.
Bit 63(A) zeigt an ob ein zusätzliches EtherCAT Telegramm nachfolgend gesendet wird.

Der Datenbereich ist maximal 1486 Byte groß. Im Datenbereich eines Ethernet-Frames können mehrere EtherCAT-Frames und somit mehrere Kommandos an verschiedene Slaves enthalten sein. Die physikalische Reihenfolge der Slaves im Strang muss dabei in der Regel nicht beachtet werden. Durch die Eigenschaft, dass mehrere EtherCAT-Kommandos in einem Ethernet-Frame Platz finden und durch ein Memory Mapping in den Slaves, welches den Zugriff auf die Speicherbereiche mehrerer Slaves mit einem EtherCAT-Kommando erlaubt, wird die Nutzdatenrate erheblich erhöht. Dadurch wird das Problem des hohen Overheads von Ethernet bei geringem jedoch wiederholten Datenaufkommen gelöst.

Das EtherCAT-Telegramm endet mit einem 2 Byte großen Working Counter. Jeder Slave, der eine Nachricht erfolgreich empfangen hat inkrementiert den Counter. Der Master hat damit die Möglichkeit Fehler zu erkennen.

d) Frame Check Sequence (FCS):

Das FCS Feld stellt eine 32-Bit-CRC-Prüfsumme dar. Ist die Prüfsumme nach Verwendung der FCS ungleich Null, war die Übertragung fehlerhaft.

3.2.5 EtherCAT Kommunikationszustände

Das AL Management in EtherCAT beschreibt das Handling der EtherCAT State Machine (ESM). Es sind der Zustand und die Zustandsänderung des jeweiligen Slaves in einer Applikation beschrieben. Der aktuelle Zustand des ECT Slaves wird im Statusregister aufgezeigt und Zustandsänderungen werden im Control Register, vom Master initiiert, angezeigt.

EtherCAT definiert vier Kommunikationszustände. Die Kommunikationszustände (State) und deren Übergänge (Transitionen) siehe ▶Abbildung 6◀.

State Änderungen werden vom Master angefragt. Der Slave antwortet korrekt wenn er den Wechsel vollzogen hat oder setzt eine Fehlermeldung ab wenn der Wechsel nicht vollzogen werden konnte.

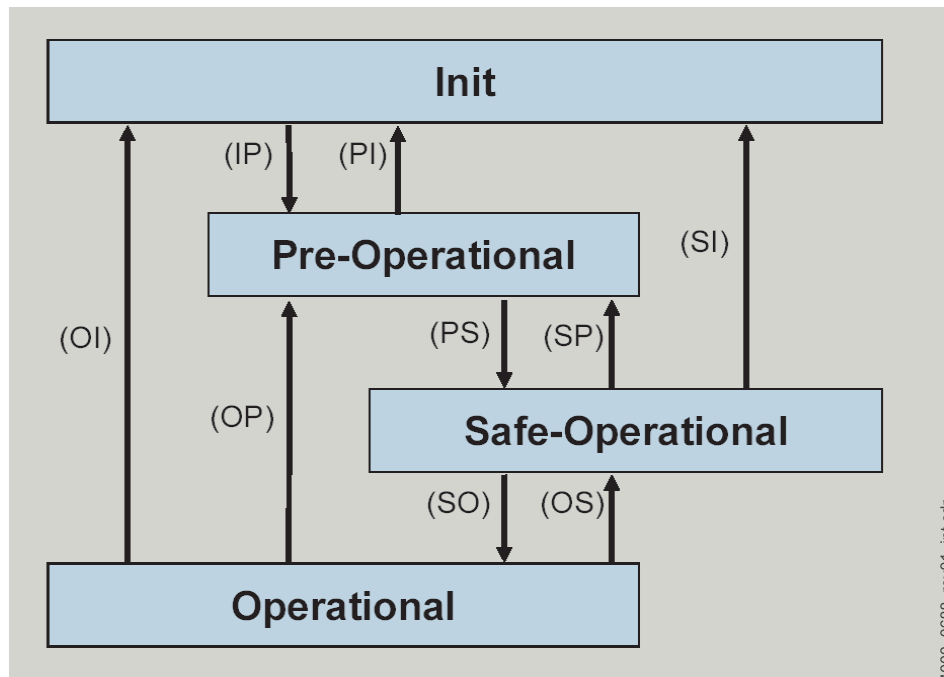


Abbildung 6: EtherCAT Kommunikationsübergänge [1]

Zustände:

- **Init:**
Initialisierung des Slaves. In der Init Phase ist keine direkte Kommunikation auf der Applikationsebene möglich
- **Pre-Operational:**
In diesem Zustand kann eine Mailbox für eine Service Daten Kommunikation konfiguriert sein (wenn der Slave diese unterstützt). Service Daten Kommunikation ist dann möglich, aber noch keine Prozessdatenkommunikation.
- **Safe-Operational:**
In diesem Zustand ist weiterhin die Service Daten Kommunikation möglich. Außerdem werden nur vom Slave ausgehende Daten, TX-Data gesendet. RX-Daten vom Master werden noch ignoriert. Mailbox ist weiterhin möglich.

- **Operational:**

Mailbox und Zyklische Kommunikation in beide Richtungen (TxPDO und RxPDO) sind nun möglich. Mailbox ist weiterhin möglich.

Die Transitionen werden in der folgenden Tabelle aufgezeigt.

State transition	Local management service
IP	Start Mailbox Communication
PI	Stop Mailbox Communication
PS	Start Input Update
SP	Stop Input Update
SO	Start Output Update
OS	Stop Output Update
OP	Stop Output Update, Stop Input Update
Si	Stop Input Update, Stop Mailbox Communication
OI	Stop Output Update, Stop Input Update, Stop Mailbox Communication

Übergänge:

Können die Anforderung des Masters zu einen Zustandswechsel durch den Slave nicht erfüllt werden, weil z.B. ein falsches Mapping vorliegt, so hat der Slave die Möglichkeit durch Senden einer Fehlernachricht dies dem Master mitzuteilen. Diese Nachricht ist ähnlich wie die der Gerätesteuerung unterteilt.

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
A0 _{hex}	00 _{hex}	08 _{hex}	04 _{hex}	00 _{hex}	00 _{hex}	00 _{hex}	00 _{hex}

Byte 0 und **Byte 1** beinhalten den Emergency Error Code.

Vom CoE Standard sind zwei Einträge definiert.

A000_{hex}: Übergang von PRE-OPERATIONAL nach SAFE-OPERATIONAL war nicht erfolgreich

A001_{hex}: Übergang von SAFE-OPERATIONAL nach OPERATIONAL war nicht erfolgreich

Byte 2:

Anhand der folgenden Tabelle ist zu erkennen welche Meldung abgesetzt wird, wenn eine falsche Parametrierung der SyncManager erfolgt.

SyncManager2 (Prozessdaten out RxPDO)	08 _{hex}	SyncManager falsch parametrier
	09 _{hex}	PDO Länge stimmt nicht mit der Mappinglänge der Objekte überein
	0A _{hex}	SyncManager-Einstellungen an unzulässiger Adresse
SyncManager3 (Prozessdaten in TxPDO)	0C _{hex}	SyncManager falsch parametrier
	0D _{hex}	PDO Länge stimmt nicht mit der Mappinglänge der Objekte überein
	0E _{hex}	SyncManager-Einstellungen an unzulässiger Adresse

Für die SyncManager0 und SyncManager1 kann keine Meldung abgesetzt werden, weil damit die Mailboxen beschrieben werden. Sind die Mailboxen falsch konfiguriert, bleibt der Slave im Zustand Init. Der nicht erfolgte Wechsel nach PRE-OPERATIONAL wird in dem Fall nur über den AI-Status dem Master mitgeteilt.

Bei falschen Syncmanager-Einstellungen wird zuerst die EMCY für SyncManager2 abgesetzt, egal ob SyncManager3 auch falsch konfiguriert ist. Erst wenn der erste Fehler beseitigt ist, wird dann die nächste Emergency gesendet.

Byte 3:

Gibt die Anzahl der folgenden Bytes an, entweder 4 Byte (oder 2 Byte EMCY-Codes bei Fehler des Gerätes).

Byte 4-7:

	Byte 4	Byte 5	Byte 6	Byte 7
SM2 Adressen Fehler	0	0	0	0
SM2 falsche Länge	High-Byte	Low-Byte	High-Byte	Low-Byte
	Minimale Länge der Syncmanager		Maximale Länge der Syncmanager	
SM2 falsch parametriert	High-Byte	Low-Byte	High-Byte	Low-Byte
	Kleinste zulässige Adresse		Größte zulässige Adresse	
SM3 Adressen Fehler	0	0	0	0
SM3 falsche Länge	High-Byte	Low-Byte	High-Byte	Low-Byte
	Minimale Länge der Syncmanager		Maximale Länge der Syncmanager	
SM3 falsch parametriert	High-Byte	Low-Byte	High-Byte	Low-Byte
	Kleinste zulässige Adresse		Größte zulässige Adresse	

Herstellerspezifischer Fehlercode:

Byte 0 und Byte 1:

A0A0_{hex}:

der Fehlercode erscheint, wenn der Antrieb synchron laufen soll, aber nach einer gewissen Zeit (in Abhängigkeit vom b maXX[®]-Gerät und vom Gerätezustand, von 100 ms bis 30 s) immer noch nicht synchron läuft.

Byte 2 beinhaltet FF_{hex} und Byte 4-7 hat den Wert Null.

Synchronisierung

Die exakte Synchronisierung von Teilnehmern bei EtherCAT erfolgt nach dem Prinzip der verteilten Uhren, wie im neuen Standard IEEE 1588 beschrieben. Jeder Slave hat eine unabhängig laufende Uhr implementiert. Dabei wird die Uhrzeit der Haupt-Uhr via EtherCAT zu den Neben-Uhren übertragen. Um die Laufzeit des Synchronisations-Telegramms zu berücksichtigen wird eine Laufzeitmessung durchgeführt. Dafür sendet der Master ein Broadcast-Telegramm aus, in das alle Slaves den Empfangszeitpunkt dieses Broadcast-Telegramms bezüglich ihrer Uhr notieren. Dadurch sind die Laufzeiten bestimmt und können vom Master entsprechend berücksichtigt werden. Bei EtherCAT wird die Haupt-Uhr in einem Slave-Gerät konfiguriert, sodass auch hierfür keine spezielle Hardware im Master erforderlich ist. Die Synchronisationsgenauigkeit ist dabei deutlich unter einer µs, bei 300 Teilnehmern und 120 m Leitungslänge wurden Abweichungen von +/- 20 ns erzielt [1].

Die notwendigen Einstellungen des Slaves durch den Master bzw. Einstellungen im Datensatz werden in [►Synchronisation \(SYNC\)◄](#) auf Seite 47 beschrieben.

3.2.6 Ethernet over EtherCAT (EoE) - TCP/IP- Tunneling über EtherCAT

Für die Ethernet-Kommunikation zu EtherCAT-Slaves (z. B. zum b maXX[®]-Regler mit EtherCAT-Slave, hier speziell für das Bedientool ProDrive) werden die TCP-Pakete innerhalb der EtherCAT-Pakete übertragen (Tunneling). In diesem Fall muss für jeden EtherCAT-Slave eine eigene IP-Adresse eingestellt werden. Über diese IP-Adresse wird der EtherCAT-Slave als Ethernet-Teilnehmer angesprochen.

Einstellmöglichkeiten der IP-Adresse:

1 Einstellung über DIP Schalter

Der Wert der DIP Schalter wird zur Basis-Adresse 192.168.1.1 addiert.

192.168.1.XXX

192.168.1 wird fest vorgegeben.

XXX bedeutet Einstellung der DIP Schalter (SW13100 bzw. S12100/S12200 auf der HW) + 1

z. B. wenn die Adresse 192.168.1.5 gewünscht wird, muss auf dem DIP Schalter 4 eingestellt werden.

2 Einstellung über ProDrive

In ProDrive auf der Seite Optionsmodule G, H-Konfiguration kann eine beliebige IP-Adresse konfiguriert werden. Dafür muss das Bit 13 in Konfigurationsparameter 1 (P0830/P0840) gesetzt werden. Die IP-Adresse kann in Konfigurationsparameter 4 und 5 (P0833/P0843 und P0834/P0844) eingetragen werden. Konfigurationsparameter 5 beinhaltet die ersten beiden Zahlen der IP-Adresse und Konfigurationsparameter 4 die letzten beiden Zahlen. Die Konfiguration sollte im Datensatz des Reglers abgespeichert werden.

The screenshot displays the 'Optionsmodule G, H-Konfiguration' page in the ProDrive software. It is divided into two main sections: 'Optionsmodul G' and 'Optionsmodul H'. Each section contains eight configuration rows, each with a text input field. In the 'Optionsmodul G' section, the values are: 0x2000, 0x0000, 0x0000, 0x0101, 0xC08A, 0x0000, 0x0000, and 0x0000. In the 'Optionsmodul H' section, all values are 0x0000. A bracket on the right side of each section groups rows 4 and 5, with a button labeled 'IP-Adresse eingeben'. Below the configuration area is an 'Information' box with a warning: 'Achtung: Eingestellte Werte für die benutzerdefinierten Einheiten haben teilweise Auswirkung auf Optionsmodulberechnungen (Siehe Handbuch zum Optionsmodul) und werden deshalb aus informativen Zwecken hier angezeigt. Die Einstellungen können auf der Seite "Umnormierung" geändert werden.' At the bottom of the information box, there is a calculation: 'Lagenormierung in Benutzereinheiten' (65536) equals 'Lagenormierung Anzahl der Umdrehungen' (1).

Abbildung 7: ProDrive Seite Optionsmodule G, H-Konfiguration

3 Einstellung über EtherCAT Master

Es besteht auch die Möglichkeit, die IP-Adresse durch den EtherCAT Master vorzugeben (falls dies vom Master unterstützt wird).

Dabei kann eine beliebige IP-Adresse gewählt werden.

Die Portnummer für die EoE Kommunikation ist 5043_{hex} (= 20547_{dez}).

Weil die EoE Kommunikation über die Mailboxen des EtherCAT erfolgt, sollten die Mailboxen mit weniger als 10ms abgefragt werden.

3.3 Grundlagen CANopen

CANopen ist ein offenes und damit herstellernerutrales Feldbussystem, welches auf den Layern 1 und 2 - Definitionen des CAN-Standards aufsetzt.

CAL-Spezifikation Das CANopen-Protokoll basiert auf der CAL-Spezifikation (Schicht 7-Protokoll). Bei CANopen werden Profile unterschieden. Das Kommunikationsprofil (DS 301) definiert die Art und Weise des Datenaustausches und allgemeine, für alle Geräte geltende Festlegungen.

Geräteprofil In den Geräteprofilen werden die anwendungs- und gerätespezifische Festlegungen, die inhaltliche Bedeutung der Daten und die Gerätefunktionalität beschrieben. Geräteprofile existieren u. a. für Antriebe, I/O-Module, Geber oder programmierbare Geräte. Das Optionsmodul CANopen-Slave für den bmaXX[®] 4400 Regler ist nach dem Geräteprofil DSP402 (Drives and Motion Control) implementiert.

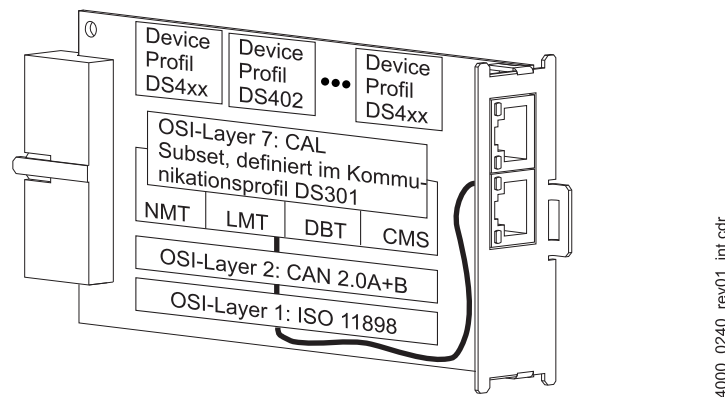


Abbildung 8: Profilstruktur von CANopen

3.3.1 Objektverzeichnis

Das zentrale Element eines jeden CANopen-Gerätes ist das Objektverzeichnis des CANopen-Gerätes.

Index (hex)	Objekt
0000	Nicht benutzt
0001 _{hex} - 001F _{hex}	Statische Datentypen
0020 _{hex} - 003F _{hex}	Komplexe Datentypen
0040 _{hex} - 005F _{hex}	Herstellerspezifische Datentypen
0060 _{hex} - 007F _{hex}	Geräteprofilsspezifische Statische Datentypen
0080 _{hex} - 009F _{hex}	Geräteprofilsspezifische Dynamische Datentypen
00A0 _{hex} - 0FFF _{hex}	Reserviert
1000 _{hex} - 1FFF _{hex}	Bereich für das Kommunikationsprofil
2000 _{hex} - 5FFF _{hex}	Bereich für herstellerepezifische Objekte
6000 _{hex} - 9FFF _{hex}	Bereich für das Geräteprofil
A000 _{hex} - AFFF _{hex}	Steuerungs-Objekte nach IEC 61131-3 programmierbare Geräte (DSP 405)

Die Objekte werden immer über einen Index (16 Bit) und zusätzlich über einen Subindex (8 Bit) adressiert.

CANopen unterscheidet vier Arten von Nachrichten:

- Administrative Nachrichten (z.B. Netzwerk-Management NMT, Layer-Management LMT)
- Servicedaten (SDO)
- Prozessdaten (PDO)
- Vordefinierte Nachrichten (z.B. Synchronisation, Time Stamp, Emergency)

NMT Mittels NMT-Diensten (Netzwerk-Management) werden die Kommunikationszustände des Gerätes gesteuert und überwacht.

SDO SDOs dienen der Übertragung größerer Datenmengen mit niedriger Priorität (Bedarfsdaten). Ein Datenblock mit mehr als vier Byte Nutzdaten wird dazu durch das CANopen-Protokoll segmentiert und auf mehrere SDOs verteilt (SDO Segmented Transfer). Datengrößen von maximal vier Byte werden mit einer SDO übertragen (SDO Expedited Transfer). Typischerweise werden SDOs zur Gerätekonfigurierung benutzt. SDOs werden asynchron übertragen und vom Empfänger bestätigt. Mittels SDO kann auf alle Einträge im Objektverzeichnis zugegriffen werden.

PDO PDOs dienen dem Austausch von Prozessdaten (Daten mit hoher Priorität). PDOs können sowohl synchron als auch asynchron übertragen werden. Sie haben Broadcast-Charakter und werden vom Empfänger nicht bestätigt.

Synchron bedeutet, dass die Übertragung vom Synchronisations-Objekt abhängt. Der Inhalt der PDOs muss vom Anwender über SDOs festgelegt werden (variables PDO-Mapping). Dieses Mapping muss vor Beginn der Prozessdatenkommunikation abgeschlossen sein. In den Geräteprofilen sind Default-Mappings angegeben.

3.3.2 Zustandsmaschine CANopen

CANopen definiert ein Boot Up des Netzwerkes. Das einfache Boot Up enthält vier Kommunikationszustände:

- INITIALISIERUNG
- PRE-OPERATIONAL
- STOPPED
- OPERATIONAL

Die einzelnen Zustandsübergänge werden durch NMT-Kommandos ausgelöst. Das Optionsmodul CANopen-Slave wechselt nach dem Initialisieren automatisch in den Zustand PRE-OPERATIONAL.

Im Vergleich zum CoE Standard (siehe [EtherCAT Kommunikationszustände](#) auf Seite 15) ist im CANopen ein Zustand, Safe-Operational, weniger definiert. Zudem werden bei CoE die State Übergänge vom Slave nach erfolgtem Wechsel bestätigt oder mit einer Fehlermeldung bei nicht erfolgtem Wechsel versehen.

3.4 Unterstützte Betriebsarten des Geräte Profils DSP 402

3.4.1 Kurze Übersicht

Folgende Betriebsarten werden unterstützt, d.h. alle mandatory Objekte sind auf dem Optionsmodul CANopen-Slave vorhanden	
Device Control	optionale Objekte vollständig vorhanden
Objekte des Homing	optionale Objekte vollständig vorhanden
Objekte des Profile Position Mode	optionale Objekte nicht vollständig vorhanden
Position Control Funktion	optionale Objekte nicht vollständig vorhanden
Objekte des Velocity Mode	optionale Objekte nicht vollständig vorhanden
Objekte des Profile Velocity Mode	optionale Objekte nicht vollständig vorhanden
Common entries in the Object Dictionary (keine mandatory-Objekte vorhanden)	optionale Objekte nicht vollständig vorhanden

Folgende Betriebsarten werden nicht unterstützt, d.h. mindestens ein mandatory Objekt ist nicht vorhanden, es können auch optionale Objekte vorhanden sein.	
Interpolated Position Mode	keine Objekte
Profile Torque Mode	ein Objekt

3.4 Unterstützte Betriebsarten des Geräte Profils DSP 402

3.4.2 Betriebsarten und Feldbus-Objekte

Betriebsart	
Feld-Bus-Objekte	Feld-Bus-Name

Objekte des Homing Mode alle mandatory Objekte und alle optionalen Objekte werden unterstützt (Referenzfahrt)		
6098 _{hex}	mandatory	homing_method
6099 _{hex}	mandatory SIX 0 = 2	homing_speed
607C _{hex}	optional	home_offset
609A _{hex}	optional	homing_acceleration

Device Control alle mandatory Objekte und alle optionalen Objekte werden unterstützt (Gerätekontrolle)		
6040 _{hex}	mandatory	controlword
6041 _{hex}	mandatory	statusword
6060 _{hex}	mandatory	modes_of_operation
6061 _{hex}	mandatory	modes_of_operation_display
605A _{hex}	optional	quick_stop_option_code
605B _{hex}	optional	shutdown_option_code
605C _{hex}	optional	disable_operation_option_code
605D _{hex}	optional	halt_reaction_option_code
605E _{hex}	optional	fault_reaction_option_code

Objekte des Profile Torque Mode ein optionales Objekt wird unterstützt (Momentenregelung)		
6072 _{hex}	optional	max_torque

Objekte des Profile Position Mode alle mandatory Objekte, zum Teil optionale Objekte werden unterstützt (Positionierung)		
607A _{hex}	mandatory	target_position
607D _{hex}	optional SIX 0 = 2	software_position_limit
607F _{hex}	optional	max_profile_velocity
6080 _{hex}	optional	max_motor_speed
6081 _{hex}	mandatory	profile_velocity
6083 _{hex}	mandatory	profile_acceleration
6084 _{hex}	mandatory	profile_deceleration
6085 _{hex}	optional	quick_stop_deceleration
6086 _{hex}	mandatory	motion_profile_type

Objekte des Profile Velocity Mode alle mandatory Objekte, zum Teil optionale Objekte werden unterstützt (Drehzahlregelung)		
606A _{hex}	mandatory	sensor_selection_code
6069 _{hex}	mandatory	velocity_sensor_actual_value
606B _{hex}	mandatory	velocity_demand_value
606C _{hex}	mandatory	velocity_actual_value
606F _{hex}	optional	velocity_threshold
60FF _{hex}	mandatory	target_velocity
60F8 _{hex}	optional	max_slippage

Objekte der Position Control Funktion alle mandatory Objekte, zum Teil optionale Objekte werden unterstützt (Positionierkontrolle)		
6067 _{hex}	optional	position_window
6068 _{hex}	optional	position_window_time
6064 _{hex}	mandatory	position_actual_value
6063 _{hex}	optional	position_actual_value*
6062 _{hex}	optional	position_demand_value
6066 _{hex}	optional	following_error_time_out
60FB _{hex}	optional SIX 0 = 28	position_control_parameter_set

Objekte des Velocity Mode alle mandatory Objekte, zum Teil optionale Objekte werden unterstützt (Drehzahlreglung)			
6042 _{hex}	mandatory		vl_target_velocity
6043 _{hex}	mandatory		vl_velocity_demand
6044 _{hex}	mandatory		vl_control_effort
6045 _{hex}	optional		vl_manipulated_velocity
6048 _{hex}	mandatory	SIX 0 = 2	vl_velocity_acceleration
6049 _{hex}	mandatory	SIX 0 = 2	vl_velocity_deceleration
6046 _{hex}	mandatory	SIX 0 = 2	vl_velocity_min_max_amount
604C _{hex}	optional	SIX 0 = 2	vl_manipulated_velocity
604D _{hex}	optional		vl_pole_number
604F _{hex}	optional		vl_ramp_function_time
6050 _{hex}	optional		vl_slow_down_time
6051 _{hex}	optional		vl_quick_stop_time

Common Entries im Object Dictionary Keine mandatory Objekte sind vorhanden, zum Teil werden optionale Objekte unterstützt (allgemeine Eingänge im Objekt Verzeichnis)			
60FD _{hex}	optional		digits_inputs
6510 _{hex}	optional	SIX 0 = 08	drive_date

Factor Group Keine mandatory Objekte sind vorhanden, zum Teil werden optionale Objekte unterstützt (Gruppe der Benutzer Einheiten)			
6092 _{hex}	optional	SIX 0 = 2	feed_constant

Objekte des Profile Torque Mode ein optionales Objekt wird unterstützt (Drehmoment)			
6072 _{hex}	optional	SIX 0 = 2	max_torque

4

KOMMUNIKATION ZUM b maXX[®] REGLER

In diesem Kapitel wird die Datenkommunikation zwischen dem b maXX[®] 4400-Gerät und dem CoE-Slave-Optionsmodul beschrieben.

4.1 Kommunikationsablauf

Das CoE-Optionsmodul tauscht über ein FPGA Daten mit dem b maXX[®] 4400 Regler aus. Dieser Datenaustausch erfolgt in einem bestimmten Zeitraster über die BACI-Schnittstelle (Baumüller-Bus).

Dabei stößt das CoE-Optionsmodul die Kommunikation mit dem b maXX[®] 4400 Regler an. Bei der Kommunikation werden zwei unterschiedliche Typen von Daten übertragen:

- Prozessdaten
- Bedarfsdaten

Die Prozessdaten werden immer zyklisch übertragen. In der Restzeit eines Zyklus werden die Bedarfsdaten übertragen. Die Prozessdaten Übertragung findet in einem einstellbaren Zeitraster statt.



HINWEIS!

Die zyklische Kommunikation (RxPDO) ist nur im CoE-Kommunikationszustand OPERATIONAL aktiv. TxPDO können auch in SAFE-OPERATIONAL gesendet werden.

4.2 Parametrierung der BACI Kommunikationszeiten

Zwischen dem Optionsmodul CoE-Slave und dem b maXX[®] Regler können 8 Sollwerte und 8 Istwerte als Prozessdaten in einem Kommunikationszyklus ausgetauscht werden. Welche Sollwerte und Istwerte ausgetauscht werden, wird in den Mapping-Objekten auf dem Optionsmodul CoE-Slave festgelegt (Einstellung über SDO durch den Master oder Default-Einstellung, siehe [►Prozessdaten◄](#) ab Seite 44). Die Parametrierung der Kommunikation wird in diesem Kapitel festgelegt.

Die Einstellung der Kommunikationszeiten zwischen Optionsmodul CoE-Slave und b maXX[®] Regler werden vom CoE Slave automatisch eingestellt und können vom Anwender nicht geändert werden. Dabei werden die BACI-Zeiten der Zykluszeit angepasst. Das Einstellen der Zykluszeit wird in [►Datenaustausch und Parametrierung◄](#) ab Seite 35 erklärt.

Auf der WinBASS II / ProDrive Seite „BACI“ (Optionsmodul1) ist es möglich die Kommunikationszykluszeit (Rate Sollwerte, Istwerte), den Zyklus-Offset der Sollwerte und den Zyklus-Offset der Istwerte abzulesen.

Der b maXX[®] Regler ruft alle 125 µs eine Kommunikationszeitscheibe auf, in der Prozessdatensollwerte oder Prozessdatenistwerte übertragen werden.

Die Kommunikationszykluszeit ist ein Vielfaches des Aufrufs der Kommunikationszeitscheibe des Reglers (alle 125 µs). In der EditBox „Rate Sollwerte, Istwerte“ wird nur der Faktor angegeben, d. h. der Wert in der EditBox „Rate Sollwerte, Istwerte“ berechnet sich wie folgt:

$$\text{Zykluszeit Sollwerte, Istwerte} = \frac{\text{Kommunikationszykluszeit (in } \mu\text{s)}}{125 \mu\text{s}}$$

Beispiel:

$$\text{Kommunikationszykluszeit} = 500 \mu\text{s} \Rightarrow \text{Sollwerte, Istwerte} = 4$$



HINWEIS!

Beachten Sie bei der Festlegung von BM_u_Baci1M1Period: auf die BACI kann nur alle 250 µs zugegriffen werden.

Ebenfalls auf der WinBASS II / ProDrive Seite „BACI“ (Optionsmodul 1) finden sie die Parameternummern der Sollwerte und der Istwerte. Diese dienen nur der Anzeige, da die Einstellung der Parameternummern für den Prozessdatenaustausch in den Mapping-Objekten auf dem Optionsmodul CoE-Slave festgelegt wird. Das Mapping wird erst beim Übergang PRE-OPERATIONAL nach SAFE- OPERATIONAL eingetragen.



HINWEIS!

Wird die zyklische Kommunikation unterbrochen, z.B. beim Übergang von OPERATIONAL nach PRE-OPERATIONAL kann der Fehler/Warnung Alive Counter oder der Fehler zyklische Kommunikation auftreten.

4.3 Konfigurationsmöglichkeiten der CoE-Optionskarte in WinBASS II / ProDrive

WinBASS II / ProDrive „Optionsmodul G/H - Konfiguration 1“.

**HINWEIS!**

Einstellungen führen zu einem geänderten Verhalten!

4.3.1 Einstellungen ab Firmware Version FW 03.00 b maXX[®]-Regler

(Änderungen einiger Normierungsfunktionen

z. B. Einheiten 1/10 U/min oder 1/100° Grad Auflösung)

Bit 2 ⇒ 0: altes Verhalten wie z.B. FW 02.08 (LC1): z.B.1 U/min

Bit 2 ⇒ 1: neue Funktionen, Neuerungen werden im weiteren Verlauf aufgeführt (z.B.1/10 U/min)

4.3.2 EMCY-Errorcode

Im DSP 402 nicht definierte Reglerfehler werden zu dem herstellerspezifischen Errorcode FF00_{hex} dazu addiert,

z.B. Reglerfehler Nummer 167 (Bremsen öffnet nicht), wird dann mit

FF00_{hex} + 00A7_{hex} (Nr. 167) = FFA7_{hex} ausgegeben.

Bit 9 ⇒ 0: neues Verhalten wie oben beschrieben Standard z.B.: FFA7_{hex}

Bit 9 ⇒ 1: altes Verhalten z.B.: FF00_{hex}

4.4 Allgemeine Bemerkung zur CoE-Optionskarte

Wichtig: Änderungen die über WinBASS II / ProDrive erfolgen, werden nicht automatisch auf der CoE Optionskarte aktualisiert oder wahrgenommen. Der Zugriff auf den Regler sollte, wenn im DSP402 vorhanden, mit FBO über CoE stattfinden.

Änderungen über WinBASS II / ProDrive bei der Umschaltung zwischen relativen und absoluten Positioniermodi werden auf der CoE Optionskarte während des Positionierbetriebes nicht wahrgenommen. Dazu zählen z. B. auch Änderungen der Betriebsart über WinBASS II / ProDrive. Die Umschaltung/Änderung darf nur über den CoE stattfinden.

Müssen die Änderungen doch über WinBASS II / ProDrive eingegeben werden, besteht die Möglichkeit, die Parameter beim Übergang der CoE Zustandsmaschine OPERATIONAL nach PRE-OPERATIONAL oder nach INIT zu aktualisieren. Weiterhin findet nach einem Abspeichern im Datensatz und einem Neu booten des Reglers eine Aktualisierung statt.

Auf folgende Parameter/FBOs ist es möglich, sowohl über WinBASS II / ProDrive als auch über die Optionskarte zuzugreifen:

P0830	kein FBO nach DSP 402, Zugriff nur über herstellerspezifisches Objekt möglich
P0304	(FBO 6060 _{hex}),
P1031	(FBO6080 _{hex}),
P3050	(FBO 6092 _{hex} SIX1),
P3051	(FBO 6092 _{hex} SIX2),
P0601	(interne Umschaltung auf der CANopen Optionskarte durch das Steuerwort Bit 6, Betriebsart Positionierung, relative und absolute Modi),
P1190	(FBO 6086 _{hex}).

Folgende Parameter können nur über Feldbus eingegeben werden:

P1172	(FBO 6048 _{hex} SIX1, SIX2) für die Ermittlung der Beschleunigung
P1173	(FBO 6049 _{hex} SIX1, SIX2) für die Ermittlung der Verzögerung
P3314	(FBO 604C _{hex} SIX1),
P3315	(FBO 604C _{hex} SIX2)

Weiterhin auch die FBOs, die nicht auf einen Parameter im Regler abgebildet werden. Siehe [▶Anhang C - Umsetzungstabellen◀](#) ab Seite 75, z.B. Mapping Parameter.

Zudem können die Normierungen der FBO von denen von WinBASS II / ProDrive abweichen.

z.B.: Eingabe der Positioniergeschwindigkeit über den FB in [m/s] und Eingabe über WinBASS II / ProDrive in [INC/ms] entspricht einem Unterschied von Faktor 1000.

4.4.1 Applikationsparameter

Es werden **keine** Applikationsparameter verwendet (anders als auf der CANopen Optionkarte, die ebenfalls von Baumüller angeboten wird).

4.4.2 Geschwindigkeitsprofil bei der Positionierung (FBO 6086_{hex})

Das Geschwindigkeitsprofil kann über das FBO 6086_{hex} eingestellt werden, auch während der Positionierung. Dabei wird der aktuelle Fahrbefehl zu Ende gefahren und dann der neue Fahrbefehl mit dem neuen Profil gestartet.

4.4.3 Einstellbares Verhalten, wenn neues Ziel außerhalb der Software-Endschalter

In WinBASS II / ProDrive unter „Antriebsmanager 2 Warnung aktivieren“ einstellbar. Im Datensatz abspeicherbar.

Wenn neues Ziel außerhalb ⇒ keine Bewegung;
Es wird eine CAN Emergency Meldung Code 8600_{hex} Positioning Controller (Regler Fehler Nr. 196 SW Endschalter 1, Regler Fehler Nr. 197 SW Endschalter 2) abgesetzt. Verhalten des Antriebs über 605A_{hex} einstellbar. Der Fehler muss quittiert werden und ein neuer Fahrsatz kann anschließend ausgeführt werden.

Wenn die aktuelle Position bereits außerhalb und das neue Ziel auch außerhalb ist ⇒ keine Bewegung;
Es wird eine CAN Emergency Meldung Code 8600_{hex} Positioning Controller (Regler Fehler Nr. 196 SW Endschalter 1, Regler Fehler Nr. 197 SW Endschalter 2) abgesetzt. Verhalten des Antriebs über 605A_{hex} einstellbar. Der Fehler muss quittiert werden und ein neuer Fahrsatz kann anschließend ausgeführt werden.

4.4.4 Fehlerauslösung bei Fahrt in Hardware-Endschalter

Die HW-Endschalterüberwachung ist in WinBASS II / ProDrive unter Antriebsmanager 2 Warnung aktivieren einstellbar.

Es wird eine CAN Emergency Meldung Code 8600_{hex} Positioning Controller (Regler Fehler Nr. 198 negativer HW Endschalter, Regler Fehler Nr. 199 positiver HW Endschalter) abgesetzt. Der erzeugte Fehler führt nicht zur Impulssperre. Er muss quittiert werden, bevor der Start eines neuen Fahrauftrages ausgeführt wird.

4.4 Allgemeine Bemerkung zur CoE-Optionskarte

4.4.5 Benutzer Einheiten BE

Die Benutzereinheiten können jetzt auch über WinBASS II / ProDrive \Rightarrow unter „Umnormierung“ eingegeben werden. Danach muss der Datensatz abgespeichert werden (Neustart des Reglers erforderlich).

Wichtig: Im Default Datensatz für die Benutzereinheiten ist 1 BE = 1 INC eingestellt.

Sind die gewünschten BE eingestellt sollten diese auch bei nachfolgenden Updates des Reglers erhalten bleiben. Vorsichtshalber nochmals in WinBASS II / ProDrive überprüfen.

Um eine gewünschte Eingabe des z.B. Positions-Sollwertes in Benutzer Einheiten BE eingeben zu können, wurde das FBO 6092_{hex} dementsprechend angepasst.

6092_{hex}: feed constant = feed / driving shaft revolutions

„driving shaft revolutions“ wird intern auf der CANopen-Optionskarte mit 65536 multipliziert.

Maximale Eingabe für „feed“ (BE) ist $0 \dots 2^{24} - 1$.

SIX1 = feed

[in Benutzer Einheiten z. B. 360.00 Grad, 1/100 Grad Auflösung]

Auf P3050 im b maXX[®] abgebildet im Datensatz abspeicherbar.

SIX2 = driving shaft revolutions

[1 Umdrehung wird intern auf der CANopen-Optionskarte mit 65536 [INC] multipliziert].

Auf P3051 im b maXX[®] abgebildet im Datensatz abspeicherbar.

Die Anzahl der Umdrehungen ist auf 255 begrenzt.

z. B.

Die Eingabe über den Feldbus 360,00 Grad wird auf der Optionskarte auf die notwendige Einheit für den Regler von 65536 Inkrementen für eine Umdrehung umgerechnet.

Beispiel: Positionssollwert in BE = 36000; entspricht 360,00 Grad.

Die Umrechnung auf der CoE-Optionskarte sieht folgendermaßen aus:

Positions Sollwert [INC] im b maXX[®]

= FBO [BE] * driving shaft revolutions * 65536 [INC] / feed [BE]

= 36000 * 1 * 65536 / 36000 [BE * INC / BE]

= 65536 [INC]



HINWEIS!

Werden die BE in WinBASS II / ProDrive verstellt, rechnet die CoE Optionskarte, nach einem Neustart, mit den neuen Werten.

**HINWEIS!**

Die Berechnung der Benutzereinheiten ist Zeit intensiv und sollte nach Möglichkeit nicht bei Zykluszeiten unter 0,5 ms genutzt werden. Die Berechnung entfällt bei der Einstellung der Benutzereinheiten auf 1:1.

Auf folgende FBOs wirken sich die BE aus:

6062_{hex}, 6063_{hex}, 6064_{hex}, 6067_{hex}, 607A_{hex}, 607C_{hex}, 607D_{hex} Sub1/2, 6081_{hex},
6083_{hex}, 6084_{hex}, 6085_{hex}, 6099_{hex} Sub1/2, 609A_{hex}

4.4.6 Getriebefaktor

Zusätzlich zu den Benutzereinheiten ist ein Getriebefaktor eingeführt worden, der mit dem Feldbus-Objekt 604C_{hex} eingestellt wird. Mit dem Getriebefaktor ist es nun möglich z. B. die Getriebeübersetzung oder andere Skalierungen zu berücksichtigen, woraus dann die notwendige Geschwindigkeit des Antriebes errechnet wird.

604C_{hex}:

vl_dimension_factor =

vl_dimension_factor_numerator / vl_dimension_factor_denominator

SIX1 = vl_dimension_factor_numerator

INT32 (-33000 ... 33000)

SIX2 = vl_dimension_factor_denominator

INT32 (-33000 ... 33000)

Die Umrechnung im Regler sieht z. B. folgendermaßen aus:

Sollwert Drehzahl Motor im b maXX[®]:

Für vl_dimension_factor_numerator = 10

und vl_dimension_factor_denominator = 5

$$\begin{aligned} \text{Sollwert Drehzahl Motor} &= \text{FBO[U/min]} * \text{vl_dimension_factor} \\ &= 100 * 10 / 5 \text{ [U/min]} \\ &= 200 \text{ [U/min]} \end{aligned}$$
**HINWEIS!**

Die Berechnung des Getriebefaktors ist sehr zeitintensiv und sollte nach Möglichkeit nicht bei Zykluszeiten unter 0,5 ms genutzt werden. Wird der Getriebefaktor auf 1:1 gesetzt entfällt die Berechnung.

Auf folgende FBOs wirkt sich der Getriebefaktor aus:

6042_{hex}, 6043_{hex}, 6048_{hex} Sub01/02, 6049_{hex} Sub01/02, 606B_{hex}, 606C_{hex}, 60FF_{hex},
6044_{hex}, 6045_{hex}

4.4.7 CANopen Offset

Abbildung der Zahlenskala USIGN32 auf INT32 (CANopen) in der Betriebsart Positionierung. Beim Schreiben/Lesen auf einige FB Objekte wird intern auf der CoE Optionskarte je nach Richtung ein Offset von 2^{31} dazu addiert bzw. subtrahiert, dies wird vorallen bei der absoluten Positionierung berücksichtigt.

Sollen die Positionswerte und die Zielposition auch in WinBASS II / ProDrive in der INT32 Zahlenskala dargestellt werden, kann auf der Seite „Umnormierung“ eine Checkbox für den Offset aktiviert werden.

Auf folgende FBOs wirkt sich der CANopen Offset aus:

$$(6062_{\text{hex}}, 6064_{\text{hex}}, 607A_{\text{hex}}, 607C_{\text{hex}}, 607D_{\text{hex}} \text{ Sub } \frac{1}{2}) - 2^{31}$$
$$(607A_{\text{hex}}, 607C_{\text{hex}}, 607D_{\text{hex}} \text{ Sub } \frac{1}{2}) + 2^{31}$$

4.4.8 Referenzfahrt für Positionierung erforderlich

Damit während der Positionierung die Abbildung der Zahlenskala USIGN32 auf INT32 ordnungsgemäß durchgeführt wird, sollte vor der Positionierung eine Referenzfahrt erfolgen.

In WinBASS II / ProDrive auf der Seite „Referenzfahrt“ mit der dafür vorgesehenen Checkbox kann die Aktivierung dafür erfolgen, ob der Antrieb eine Positionierung zulässt, wenn keine erstmalige Referenzfahrt erfolgt ist.

Deaktiviert:

Zum Betrieb in der Betriebsart Positionieren wird ohne Referenzierung auch keine Fehlermeldung abgegeben. Es ist zu beachten, dass im Positioniermodus „CANopen“ durch die Zahlenskalawandlung auf eine falsche Referenzfahrt referenziert wird.

Wichtig:

Die Referenzierung sollte nach Möglichkeit nur über den Feldbus erfolgen. Ist dies nicht möglich oder vorgesehen, sollte unbedingt die Checkbox für den Offset in WinBASS II / ProDrive aktiviert werden (siehe [▶CANopen Offset◀](#) auf Seite 32).

Aktiviert: Wenn der Antrieb in Betriebsart Positionieren freigegeben wird, ohne dass vorher eine Referenzfahrt erfolgt ist, wird eine Fehlermeldung (EMYC-Telegramm 8600_{hex} ⇒ Regler Fehler Nr. 200) ausgegeben und der Antrieb bleibt lagegeregelt auf der aktuellen Position stehen. Positionieraufträge werden nicht ausgeführt. Erst nachdem eine Referenzfahrt durchgeführt wurde (einmalig nach dem Einschalten), werden Positionieraufträge ausgeführt. Die Fehlermeldung kann nur quittiert werden, wenn eine Referenzfahrt durchgeführt wurde. Nach der Referenzierung kann eine Positionierung gestartet werden.

4.4.9 Varianten der Positionierung, Zielposition abhängig vom Positioniermodus (P0601)



HINWEIS!

Es ist darauf zu achten, das in WinBASS II / ProDrive unter Positionierung 0 auch der Positionier-Datensatz 0 eingestellt ist, ansonsten wird die Positionierung über den CoE nicht korrekt durchgeführt. Das Umschalten zwischen den Positioniermodi „relativ (negativ/positiv)“ und „absolut“ findet nur über das Steuerwort statt. Der Positionierung sollte immer eine Referenzierung vorausgehen.

Auf der CoE Optionskarte wird der CANopen Mode (Wert 9 im Parameter P0601) default eingestellt. Hierbei wird die Zielposition in INT32 berechnet. Die Umschaltung zwischen absolut und relativ findet nur über das Steuerwort statt. Der CANopen Mode sollte bei der CoE Optionskarte bevorzugt werden.

Bei Gebrauch eines anderen Positioniermodus ist zu beachten, dass der Zahlenbereich sich bei der absoluten Positionierung verschiebt (siehe auch [CANopen Offset](#) auf Seite 32).

Positioniermodi	Beschreibung
Absolut/relativ CoE: Default (Wert 9)	<ul style="list-style-type: none"> • Ziel steht in P0607 (INT32) • Umschaltung „absolut/relativ“ findet nur über das Steuerwort statt
Relativ/positiv/negativ (Wert 4)	<ul style="list-style-type: none"> • Ziel steht in P0607 (INT32) • keine Umschaltung „absolut/relativ“ über das Steuerwort • Datentyp INT32
Absolut/relativ (Wert 10)	<ul style="list-style-type: none"> • Ziel steht in P0600 • Absolut = USIGN32 • Relative = INT32 • Umschaltung „absolut/relativ“ findet nur über das Steuerwort statt
Modulo-Positionierung	In Richtung des kürzeren Weges (wegoptimiert) <ul style="list-style-type: none"> • Ziel steht in P0607 (INT32) • Umschaltung „absolut/relativ“ findet nur über das Steuerwort statt
Alle anderen Modis	<ul style="list-style-type: none"> • Ziel steht in P0600 (USIGN32) • keine Umschaltung „absolut/relativ“ über das Steuerwort • Datentyp INT32

Umschaltung „absolut/relativ“. Über das Steuerwort Bit 6

Steuerwort Bit 6 = 0 ⇒ absolut

Steuerwort Bit 6 = 1 ⇒ relativ

Umrechnung von INT32 nach UINT32 soll heißen, ein Offset von 2^{31} wird je nach Richtung dazu addiert oder abgezogen [CANopen Offset](#) auf Seite 32.

4.4 Allgemeine Bemerkung zur CoE-Optionskarte

In **P1190** mit Bit 9 ist es möglich das automatische Einstellen des Modus „Absolut/ Relativ CoE“ während der Init Phase der CoE-Optionskarte zu deaktivieren.

P1190 Bit 9 = 0 \Rightarrow aktiviert

P1190 Bit 9 = 1 \Rightarrow deaktiviert.

Dabei muss beachtet werden, dass zum Start der Positionierung auch Bit 11 notwendig ist, siehe auch Parameterhandbuch b maXX[®] 4400, Parameter **P1190**.

DATENAUSTAUSCH UND PARAMETRIERUNG

Der Zugriff auf Daten oder Parameter findet bei CoE über Objekte statt.

Entsprechend der Profilstruktur werden Objekte zur Kommunikationssteuerung (Indizes $1XXX_{\text{hex}}$) und anwendungs- oder gerätespezifische Objekte unterschieden. Die letzteren gliedern sich in Objekte nach Profil DSP 402 (Indizes $6XXX_{\text{hex}}$) und herstellerspezifische Objekte (Indizes $4XXX_{\text{hex}}$). Eine Auflistung der $6XXX_{\text{hex}}$ -er und $4XXX_{\text{hex}}$ -er Objekte finden Sie in [►Anhang B - Kurzreferenz◀](#) ab Seite 69.

Wichtig:

Bei den herstellerspezifischen Objekten ($4XXX_{\text{hex}}$) ergibt sich der Objektindex aus $4000_{\text{hex}} + b \text{ maXX}^{\text{®}}$ 4400-Parameternummer in Hexadezimal,

z. B. ist das Objekt $412C_{\text{hex}}$ auf den $b \text{ maXX}^{\text{®}}$ 4400-Parameter **P0300**, dem Controlword umgesetzt. Diese Objekte besitzen nur den Subindex 00_{hex} .

5.1 Verzeichnis der Objekte zur Kommunikationssteuerung

In diesem Abschnitte finden Sie alle von dem Baumüller CANopen-Optionsmodul unterstützten Objekte des kommunikationsspezifischen Bereiches des Objektverzeichnis nach DS301.

Name	Index	Subindex	Datentyp	Defaultwert
Device Type	1000_{hex}	00_{hex}	U32	$XX020192_{\text{hex}}$

Dieses Objekt kann nur gelesen werden und enthält die Information, um welches Gerät es sich handelt (Drive nach DSP 402), nur lesbar.

Bit 31 .. 24 Herstellerspezifische Objekte:

Bit 25	Bit 24	Optionskarte für:
1	0	$b \text{ maXX}^{\text{®}}$ 4400

5.1 Verzeichnis der Objekte zur Kommunikationssteuerung

Name	Index	Subindex	Datentyp	Defaultwert
Manufacturer Device Name	1008 _{hex}	00 _{hex}	VString	-

Dieses Objekt ist nur lesbar. Es enthält die Zeichenfolge: „b maXX 4400“.

Name	Index	Subindex	Datentyp	Defaultwert
Manufacturer Hardware Version	1009 _{hex}	00 _{hex}	VString	-

Dieses Objekt ist nur lesbar. Es enthält die aktuelle Hardware-Version des Optionsmoduls, z. B. die Zeichenfolge: „HV01.00“.

Name	Index	Subindex	Datentyp	Defaultwert
Manufacturer Software Version	100A _{hex}	00 _{hex}	VString	-

Dieses Objekt ist nur lesbar. Es enthält die aktuelle Software-Version des Optionsmoduls, z. B. die Zeichenfolge: „SV01.00“.

Name	Index	Subindex	Datentyp	Defaultwert
Identity Object	1018 _{hex}	00 _{hex}	U8	03 _{hex}
Vendor ID		01 _{hex}	U32	15 _{hex}
Product Code		02 _{hex}	U32	0
Revision Number		03 _{hex}	U32	Siehe unten
Serial Number		04 _{hex}	U32	0

In diesem Objekt sind Informationen über das Gerät enthalten.
Die Revision Number enthält den aktuellen Stand der Firmware z.B. 00030002 für FW 03.02.

Name	Index	Subindex	Datentyp	Defaultwert
EtherCAT-Adresse	1100 _{hex}	00 _{hex}	U16	-

In diesem Objekt wird die EtherCAT-Adresse, die vom Master vergeben wurde angezeigt.

Name	Index	Subindex	Datentyp	Defaultwert
1. Receive PDO Mapping	1600 _{hex}	00 _{hex}	U8	1
		01 _{hex}	U32	60400010 _{hex}
		:	:	
		n _{hex}	U32	

Dieses Objekt enthält den Inhalt der Receive-PDO1. In Subindex 00_{hex} steht die Gesamtzahl der nachfolgenden Einträge. In Subindex 01_{hex} ist defaultmäßig das Steuerwort (Objekt 6040_{hex} Subindex 00_{hex} Länge 10_{hex}) eingetragen. Es sind max. 8 Einträge zu je 4 Byte möglich (bei einer Zykluszeit von 250 µs), siehe auch [▷PDO-Mapping◁](#) ab Seite 44.

Name	Index	Subindex	Datentyp	Defaultwert
1. Transmit PDO Mapping	1A00 _{hex}	00 _{hex}	U8	01 _{hex}
		01 _{hex}	U32	60410010 _{hex}
		:	:	
		n _{hex}		

Dieses Objekt enthält den Inhalt der Transmit-PDO1. In Subindex 00_{hex} steht die Gesamtzahl der nachfolgenden Einträge. In Subindex 01_{hex} ist defaultmäßig das Statuswort (Objekt 6041_{hex} Subindex 00_{hex} Länge 10_{hex}) eingetragen. Es sind max. 8 Einträge zu je 4 Byte möglich (bei einer Zykluszeit von 250 µs), siehe auch [▷PDO-Mapping◁](#) ab Seite 44.

Name	Index	Subindex	Datentyp	Defaultwert
Sync Manager Communication Type	1C00 _{hex}	00 _{hex}	U8	04 _{hex}
Communication Type Manager 0 Mailbox receive (Master to Slave)		01 _{hex}	U8	1 (SM0)
Communication Type Manager 1 Mailbox transmit (Slave to Master)		02 _{hex}	U8	2 (SM1)
Communication Type Manager 2 Rx-PDO (Master to Slave)		03 _{hex}	U8	3 (SM2)
Communication Type Manager 3 Tx-PDO (Slave to Master)		04 _{hex}	U8	4 (SM3)

In diesem Objekt sind Informationen über die Sync Manager Einstellungen enthalten. Das Objekt ist nur lesbar.

5.1 Verzeichnis der Objekte zur Kommunikationssteuerung

Name	Index	Subindex	Datentyp	Defaultwert
Sync Manager Communication Type Channel 0	1C10 _{hex}	00 _{hex}	U8	0

Das Objekt beinhaltet Informationen über den Kommunikationstyp des Sync Manager Kanal 0 (Mailbox received), speziell die Anzahl der PDOs die dafür konfiguriert sind. Bei der CoE-Optionskarte sind das Null PDOs auf diesem Kanal. Das Objekt ist nur lesbar.

Name	Index	Subindex	Datentyp	Defaultwert
Sync Manager Communication Type Channel 1	1C11 _{hex}	00 _{hex}	U8	0

Das Objekt beinhaltet Informationen über den Kommunikationstyp des Sync Manager Kanal 1 (Mailbox transmit), speziell die Anzahl der PDOs die dafür konfiguriert sind. Bei der CoE-Optionskarte sind das Null PDOs auf diesem Kanal. Das Objekt ist nur lesbar.

Name	Index	Subindex	Datentyp	Defaultwert
Number of assigned RxPDOs	1C12 _{hex}	00 _{hex}	U8	0-255
PDO Mapping Object Index des zugehörigen RxPDO Sync Manager Communication Type Channel 2		01 _{hex}	U16	1600 _{hex} : RX-PDO1

Das Objekt beinhaltet Informationen über den Kommunikationstyp des Sync Manager Kanal 2 (Process data output). Es wird angezeigt wie viele und welche RxPDOs vom Slave unterstützt werden. Bei der CoE-Optionskarte ist das eine RxPDO.

Name	Index	Subindex	Datentyp	Defaultwert
Number of assigned RxPDOs	1C13 _{hex}	00 _{hex}	U8	0-255
PDO Mapping Object Index des zugehörigen RxPDO Sync Manager Communication Type Channel 3		01 _{hex}	U16	1A00 _{hex} : TX-PDO1

Das Objekt beinhaltet Informationen über den Kommunikationstyp des Sync Manager Kanal 3 (Process data input). Es wird angezeigt wie viele und welche RxPDOs vom Slave unterstützt werden. Bei der CoE-Optionskarte ist das eine TxPDO.

Name	Index	Subindex	Datentyp	Defaultwert
Sync Manager Synchronization	1C32 _{hex}	00 _{hex}	U8	02 _{hex}
Synchronization type		01 _{hex}	U16	siehe unten
Cycle time		02 _{hex}	U32	Hier wird die Zykluszeit eingetragen. Angabe in ns.

Synchronisation type

0	keine Synchronisierung des Reglers
02 _{hex} (2 _{dez})	DC Sync0 Synchronisierung auf AL Event Sync0
22 _{hex} (34 _{dez})	SyncSM2 Synchronisierung auf AL Event Sync Manager2 (RxPDO vom Master)

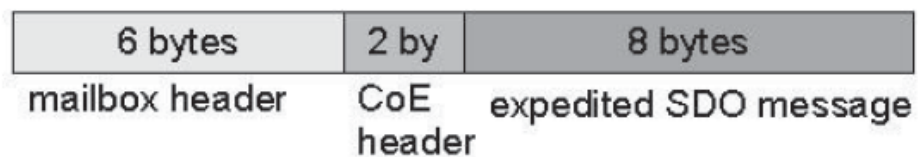
In diesem Objekt sind Informationen über die Synchronisierungsart des Sync Managers enthalten.

Die Zykluszeit wird in ns angegeben, z.B. 1 ms $\hat{=}$ 1 000 000 ns.

5.2 Bedarfsdaten (SDO)

Service Daten Objekte (SDO) dienen dem Austausch von Nachrichten ohne Echtzeitanforderungen. Die SDOs werden zur Parametrierung der Slaves und zur Einstellung der Kommunikationsbeziehungen für die PDOs verwendet. Der Datenzugriff erfolgt ausschließlich über die Objektliste. SDOs sind immer bestätigte Daten, d.h. der Sender erhält eine Quittung vom Empfänger. Für die Übertragung der SDO werden in ECT die Mailbox Dienste in Anspruch genommen.

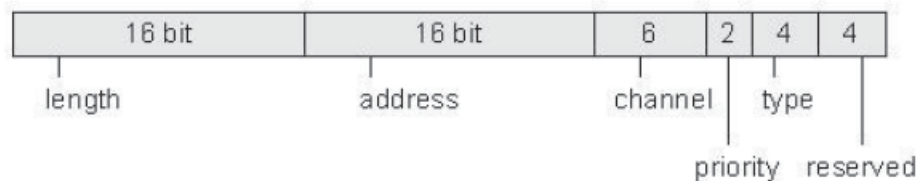
Die Mailbox ist aufgeteilt in einen Telegramm-Header und den Mailbox Datenbytes. In [Abbildung 9](#) auf Seite 40 ist der Mailbox -Aufbau schematisch dargestellt.



4000_0689_rev01_int.cdr

Abbildung 9: Aufbau der Mailbox

Der Aufbau des Mailbox-Headers ist weiter unterteilt:



4000_0690_rev01_int.cdr

Abbildung 10: Mailbox-Header

Length	Anzahl der Bytes der Mailbox, die anschließend dem Header folgen
Address	ECT-Adresse des jeweiligen Slaves
Type	Type des benutzten Protokolls der Mailbox, z.B. 3. CoE (CANopen over EtherCAT)

Der CoE-Header unterteilt sich wie folgt:



4000_0691_rev01_int.cdr

Abbildung 11: CoE-Header

PDO number	Mit der Mailbox ist es auch möglich PDOs zu übertragen. Hier wird angegeben, ob die Mailbox für PDO Übertragung konfiguriert wurde.
Type	0: Reserved 1: Emergency Message 2: SDO Request 3: SDO Response 4: TxPDO 5: RxPDO 6: Remote Übertragung der TxPDO 7: Remote Übertragung der RxPDO 8: SDO Information 9 - 15: Reserved

5.2.1 Telegrammaufbau nach CANopen

Der Telegrammaufbau bei ECT ist in den Datenbytes nach dem CANopen-Standard definiert. Allerdings kann die Grenze von 8 Byte aufgebrochen werden, je nachdem ob ein Slave dieses unterstützt oder nicht.

Das Datenfeld des Datentelegramms (8 Byte) für eine SDO gliedert sich in drei Teile, einem Command Specifier CS (1 Byte), einem Multiplexor M (3 Byte) und dem eigentlichen Nutzdatenbereich D0 - D3 (4 Byte).

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
CS	M	M	M	D0	D1	D2	D3

Der Multiplexor M besteht aus dem 16 Bit Index eines Objektes und dem dazugehörigen acht Bit breiten Subindex.

Der Command Specifier CS für eine Schreibanforderung im Expedited transfer lautet für die verschiedenen Längen:

Datenlängen in D0 - D3	Command Specifier CS
1 Byte	2F _{hex}
2 Byte	2B _{hex}
4 Byte	23 _{hex}

Der CS für die Antwort auf eine Schreibanforderung beträgt CS = 60_{hex} oder im Fehlerfall CS = 80_{hex}.

Der Command Specifier CS für eine Leseanforderung im Expedited transfer lautet CS = 40_{hex}.

Die Antwort für die verschiedenen Längen ist dann:

Datenlängen in D0 - D3	Command Specifier CS
1 Byte	4F _{hex}
2 Byte	4B _{hex}
4 Byte	43 _{hex}

5.2.2 Arten des SDO-Transfers

Die Baumüller Anschaltung unterstützt den Expedited Transfer und den Segmented Transfer, wobei letzterer nur für die Objekte 1008_{hex}, 1009_{hex} und 100A_{hex} Manufacturer Device Name genutzt wird.

Expedited Transfer

Es können Objekte geschrieben oder gelesen werden, deren Daten maximal 4 Byte umfassen. Es sind nur zwei Telegramme erforderlich, eine Anforderung und eine Antwort. Alle Objekte mit den Indizes 1XXX_{hex}, 4XXX_{hex}, 6XXX_{hex} sind über Expedited SDOs ansprechbar mit Ausnahme der Objekte 1008_{hex}, 1009_{hex} und 100A_{hex}.

Segmented Transfer

Für Objekte mit Daten größer als 4 Byte ist der Segmented Transfer erforderlich. Dabei wird die 8 Byte Grenze für die Nutzdaten aufgebrochen. Dies ist nur beim Lesen der Objekte 1008_{hex}, 1009_{hex} und 100A_{hex} möglich.

5.2.3 Fehlerreaktionen

Fehlerhafte SDO-Zugriffe werden mit Abort Codes abgewiesen. Der Aufbau dieser Abort-Telegramme ist identisch zu dem in [▶Abbildung 5.2.1◀](#) auf Seite 41 dargestellten SDO-Telegramm. Das Datenfeld enthält einen Abort Code mit 4 Bytes.

Bei fehlerhaften Zugriffen auf kommunikationsspezifische Objekte (1XXX_{hex}) werden folgende Meldungen unterschieden:

Abort Code	Bedeutung
05 _{hex} 03 _{hex} 00 _{hex} 00 _{hex}	Parameter inkonsistent (Toggle Bit hat nicht gewechselt)
05 _{hex} 04 _{hex} 00 _{hex} 00 _{hex}	SDO Protokoll Zeitüberschreitung
05 _{hex} 04 _{hex} 00 _{hex} 01 _{hex}	Client/Server Kommando Specific CS nicht gültig oder unbekannt.
05 _{hex} 04 _{hex} 00 _{hex} 05 _{hex}	Speicherbereich überschritten
06 _{hex} 01 _{hex} 00 _{hex} 00 _{hex}	Fehler im Datenformat
06 _{hex} 01 _{hex} 00 _{hex} 01 _{hex}	Lesen auf ein nur schreibbares Objekt
06 _{hex} 01 _{hex} 00 _{hex} 02 _{hex}	Schreiben auf ein schreibgeschütztes Objekt
06 _{hex} 02 _{hex} 00 _{hex} 00 _{hex}	Objekt existiert nicht im Objektverzeichnis
06 _{hex} 04 _{hex} 00 _{hex} 41 _{hex}	Daten können nicht gemappt werden (z. B. falsche Längenangabe)
06 _{hex} 04 _{hex} 00 _{hex} 42 _{hex}	Die Objektnummer und die Länge des Objekts welches gemappt werden soll liegt außerhalb der PDO Länge
06 _{hex} 04 _{hex} 00 _{hex} 43 _{hex}	Allgemeine Parameterinkompatibilität
06 _{hex} 06 _{hex} 00 _{hex} 00 _{hex}	Zugriffsfehler auf die Hardware (speichern/laden vom Flash)
06 _{hex} 07 _{hex} 00 _{hex} 10 _{hex}	Inkorrekte Datenwert Länge
06 _{hex} 09 _{hex} 00 _{hex} 11 _{hex}	Subindex existiert nicht
06 _{hex} 09 _{hex} 00 _{hex} 30 _{hex}	Wertebereich überschritten (bei Schreibzugriffen)
06 _{hex} 09 _{hex} 00 _{hex} 31 _{hex}	Wert zu groß (bei Schreibzugriffen)
06 _{hex} 09 _{hex} 00 _{hex} 32 _{hex}	Wert zu klein (bei Schreibzugriffen)
08 _{hex} 00 _{hex} 00 _{hex} 00 _{hex}	Allgemeiner Fehler
08 _{hex} 00 _{hex} 00 _{hex} 20 _{hex}	Daten können nicht zur Applikation übertragen oder abgespeichert werden
08 _{hex} 00 _{hex} 00 _{hex} 22 _{hex}	Daten können aufgrund des derzeitigen Kommunikationszustandes nicht gemappt werden (z. B. Mapping ändern im Zustand OPERATIONAL)

5.3 Prozessdaten

Prozess-Daten-Objekte (PDO) sind auf den Austausch von Daten mit Echtzeitanforderungen optimiert. In den PDOs auf der CoE-Optionskarte können maximal 32 Byte je Kommunikationsrichtung für die Nutzdatenübertragung/zyklische Kommunikation verwendet werden. Für den Datenaustausch über die PDOs muss die exakte Lage der Objekte im EtherCAT-Frame vor Beginn der Kommunikation zwischen Sender und Empfänger definiert werden. Die „Fieldbus Memory Management Unit FMMU“ ordnet den logischen Adressraum des EtherCAT Busses den physikalischen Adressraum des Slaves zu. Die Konfiguration findet üblicherweise in der Init-Phase durch den Master statt.

Die Prozessdaten des EtherCAT Slaves werden durch die SyncManager Channels beschrieben. Jeder SyncManager beschreibt einen zusammenhängenden Speicherbereich der zyklischen Daten. Mit den SyncManager wird auch die Mailbox beschrieben. Die EtherCAT Optionskarte unterstützt 4 SyncManager, 2 für die Mailbox einen je Richtung und 2 SyncManager als RxPDO bzw. TxPDO. Die Konfiguration der SyncManager erfolgt wie auch bei den FMMUs durch den Master. Bitweise Adressierung ist im CoE-Standard vorgesehen, bei der CoE-Optionskarte aber nicht möglich (nur byteweise Adressierung wird unterstützt).

Für die Übertragung der zyklischen Daten und die Synchronisierung des Reglers sind bei der CoE-Optionskarte drei Synchronisierungsarten möglich. Synchronisierung deaktiviert (der Betrieb ist nur bis zum Zustand PRE-OPERATIONAL, ab FW-Version V01.01 bis zum Zustand OPERATIONAL möglich), Synchronisierung auf SyncManager2 (RxPDO1) und Synchronisierung auf Distributed Clocks DC.



HINWEIS!

Alle in den PDOs konfigurierten Objekte werden zwischen der CoE-Optionskarte und dem b maXX[®]-Regler als zyklische Daten übertragen (siehe auch [►Kommunikationsablauf◄](#) ab Seite 25). Da die zyklische Datenübertragung (speziell die RxPDOs) nur im Zustand OPERATIONAL stattfindet, sollte auch nur in diesem Zustand die Kommunikationsüberwachung in WinBASS II / ProDrive BACI aktiviert werden, weil in den anderen Zuständen (z.B. PRE-OPERATIONAL) sonst eine Fehlermeldung generiert wird (Timeout für zyklische Kommunikation **P0836** BACI). Diese muss dann nach Übergang zu OPERATIONAL quittiert werden.

5.3.1 PDO-Mapping

Mapping ist ein Verfahren zur Zuordnung von Variablen/Objekten an PDOs. Mit den PDOs werden diese Variablen/Objekte über den Bus transportiert. Durch das Mapping wird der zyklische Datenaustausch projiziert. Für diese Parametrierung werden SDOs genutzt. Das Mapping wird über im Objektverzeichnis adressierbare Objekte eingestellt. Für jede PDO existiert ein solches Objekt.

Prozess-Daten-Objekt	Objekt für Inhalt
TX-PDO1	1A00 _{hex}
RX-PDO1	1600 _{hex}



HINWEIS!

Das Mapping kann nicht im Zustand OPERATIONAL/SAFE-OP geändert werden. Ein neues Mapping wird erst beim Übergang nach SAFE-OPERATIONAL/OPERATIONAL aktiviert.

Durch das Mapping wird der logische Inhalt der PDOs festgelegt. Für diese Festlegung werden bestimmte Angaben über das zu mappende Objekt benötigt: Objektindex, Subindex und Länge des Datums. Aus dem Objektverzeichnis werden die entsprechenden Objekte in das Mapping-Objekt eingetragen. Die Reihenfolge dieses Eintrages, festgelegt durch den Subindex des Mapping-Objektes, bestimmt die Reihenfolge der Daten im EtherCAT-Telegramm. In den Mapping-Objekten (1600_{hex}, 1A00_{hex}) werden die zu mappenden Objekte an die entsprechenden Subindizes (mit 01_{hex} beginnend) geschrieben, z.B. wird auf Objekt 1600_{hex} Subindex 01_{hex} der Wert 60400010_{hex} eingetragen. Dies bedeutet, die ersten beiden Bytes der in RX-PDO1 empfangenen Daten werden auf das Steuerwort (Objekt 6040_{hex} Subindex 00_{hex}) geschrieben. Das Objekt 6040_{hex} ist auf den b maXX[®]-Parameter **P0300** Steuerwort umgesetzt (siehe auch [►Anhang C - Umsetzungstabellen◄](#) ab Seite 75). Damit wird das erste Wort des in RX-PDO1 empfangenen Telegramms auf das Steuerwort des b maXX[®] geschrieben. In Subindex 00_{hex} muss die Anzahl der zu mappenden Objekte (Anzahl der mit gültigen Objekten belegten Subindizes) eingetragen werden. Ein Beispiel für das Mapping ist im [►Beispiel für das PDO-Mapping◄](#) ab Seite 49 erläutert.

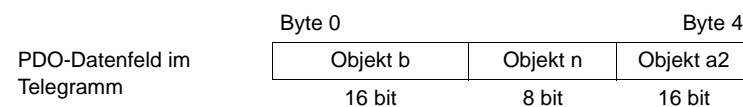
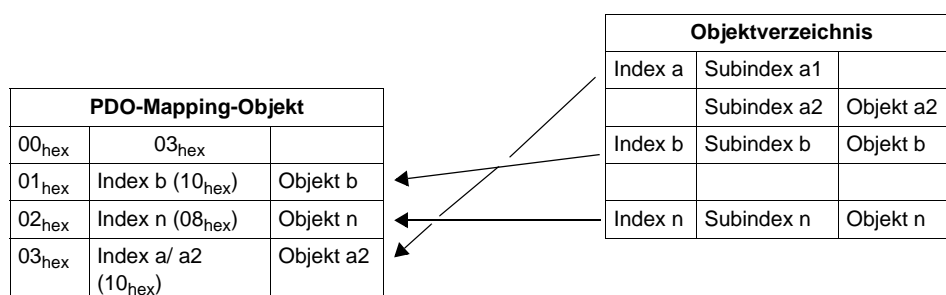


Abbildung 12: Mapping

Das Default-Mapping ist in [►Verzeichnis der Objekte zur Kommunikationssteuerung◄](#) ab Seite 35 beschrieben.

Um ein bestehendes Mapping zu löschen, können die Werte in den Subindizes mit neuen Objekten überschrieben oder zu Null gesetzt werden. Mit dem Schreiben des Wertes „0“ auf den Subindex 00_{hex} der entsprechenden PDO (1600_{hex}, 1A00_{hex}) deaktiviert man die PDO.



HINWEIS!

Bei der Einstellung des Mappings in den (1600_{hex}, 1A00_{hex}) ist jeweils der Subindex 00_{hex} mit der richtigen Anzahl der gemappten Objekte zuletzt zu beschreiben.

Sollwerte

Die zulässigen zyklische Sollwerte sind in einer Tabelle mit der Spalte „PDO-Mapping“ als „RX“ gekennzeichnet. Die Tabelle ist im Anhang B.2 (für die 6000'er Objektnummern) zu finden. Bei den herstellerspezifischen Parameter (4000'er Objekte) muss im Parameterhandbuch b maXX[®] 4400 Grundgerät (5.02017), Kapitel 6.1.4 Attribute, für das b maXX[®] 4400 nachgeschaut werden.

Istwerte

Die zulässigen zyklische Istwerte sind in einer Tabelle mit der Spalte „PDO-Mapping“ als „TX“ gekennzeichnet. Die Tabelle ist im Anhang B.2 (für die 6000'er Objektnummern) zu finden. Bei den herstellerspezifischen Parametern (4000'er Objekte) muss im Parameterhandbuch b maXX[®] 4400 Grundgerät (5.02017) für das b maXX[®] 4400 nachgeschlagen werden. Eine detaillierte Beschreibung der b maXX[®]-Parameter finden Sie ebenfalls im Parameterhandbuch b maXX[®].

Fehlerhafte Mapping-Konfigurationen (unzulässige Objekte in 1600_{hex}, 1A00_{hex}) werden durch Abort Codes über SDO gemeldet.

Die zyklischen Soll-/Istwerte werden lückenfrei in der BACI-Konfiguration initialisiert, d. h. der erste Sollwert von PDO1 steht an erster Stelle in der BACI, der zweite Sollwert von PDO1 an zweiter Stelle usw. Anschließend folgen die Sollwerte der PDO2. Analog gilt für die Istwert-Initialisierung der erste Istwert von PDO1 steht an erster Stelle in der BACI, der zweite Istwert von PDO1 an zweiter Stelle usw.

Dummy-Mapping

Das Optionsmodul CoE-Slave stellt 2 Dummy-Objekte bereit: ein 1 Byte Dummy-Objekt und ein 2 Byte Dummy-Objekt, welches ebenfalls in eine PDO gemappt werden kann. Diese Objekte haben die Indices 0005_{hex} (1 Byte Dummy) und 0006_{hex} (2 Byte Dummy). Das Dummy-Objekt dient als Platzhalter, um nur bestimmte Objekte innerhalb eines Telegramms zu verwenden (siehe auch [►Beispiel für das PDO-Mapping◄](#) ab Seite 49).



HINWEIS!

Das aktuell eingestellte Mapping geht nach dem Ausschalten verloren. Danach stellt sich das Default-Mapping ein, wenn kein neues Mapping beschrieben wird.

5.3.2 Synchronisation (SYNC)

Zum Synchronisieren des Reglers kann auf zwei Synchronisiermechanismen zurück gegriffen werden. Erstens Synchronisieren auf SM2 (RxPDO) und zweitens mit den Distributed Clocks (DC). Die DC sind im [►Grundlagen EtherCAT◄](#) ab Seite 10 kurz vorgestellt worden.

Beide Arten lösen auf dem Optionsmodul CoE einen Interrupt aus, der an den b maXX[®] Regler weitergeleitet wird. Damit kann dieses Signal zur Synchronisation des b maXX[®] Reglers verwendet werden.

Einstellung der Zykluszeit

Die Einstellung der Zykluszeit sollte vorzugsweise über das FBO 1C32 SIX2 erfolgen. Dabei wird die Zykluszeit in ns angegeben. Eingabe 1 000 000 entspricht z.B. dann einer Zykluszeit von 1 ms. Die Eingabe über das FBO wird von der Optionskarte vorgezogen und kann die im Datensatz abgespeicherte Zykluszeit ändern. Wird eine Zykluszeit ungleich den zulässigen Zykluszeiten 8 ms, 4 ms, 2 ms, 1 ms, 500 µs und 250 µs geschrieben, wird die Synchronisation abgeschaltet (in WinBASS II / ProDrive unter „Synchronisieren“ zu erkennen).

Es gibt noch die Möglichkeit die Zykluszeit über das Visualisierungstool WinBASS II / ProDrive des Reglers einzustellen. Auf der Seite „Synchronisierung“ wird die zulässige (siehe oben) Zykluszeit eingestellt. Zusätzlich sollte als SyncQuelle „Sync 1 Signal von der BACI“ eingestellt werden. Nach der Einstellung muss der Datensatz abgespeichert und der Regler neu gebootet werden. Default wird auf SM2 synchronisiert, wenn die DC (Distributed Clocks) nicht aktiviert wurden. Beim Synchronisieren auf SM2 ist im FPGA der Optionskarte eine Funktion aktiviert, die den Jitter der RxPDO vom Master ausgleicht (PLL). Bei Verwendung der DC wird diese Funktion deaktiviert.

Wird keine der beiden Möglichkeiten genutzt, wird das Synchronisieren des Reglers deaktiviert. **Dabei ist zu beachten, dass die Optionskarte nicht mehr nach SAFE-OPERATIONAL geschaltet werden kann.** Erst ab FW-Version V01.01 ist ein Zustandswechsel nach OPERATIONAL möglich.

Einstellungen für die BACI werden von dem Optionsmodul vorgenommen.



HINWEIS!

Wird im laufenden Betrieb die Synchronisationsart gewechselt, muss der Regler neu gebootet werden.

Master-Einstellungen für die Verwendung von Distributed Clocks (DC)

Damit die DC im ECT-FPGA (ASIC) überhaupt aktiviert werden kann, muss die Register Adresse des ECT ASIC 981_{hex} (siehe hierzu [3]) vom Master folgend beschrieben werden:

Bit 0 ⇒ 1 „Activate cyclic operation“

Bit 1 ⇒ 1 „Activate Sync0“

Die Überprüfung im Slave findet beim Übergang PRE-OPERATIONAL nach SAFE-OPERATIONAL der ECT Zustandsmaschine statt.

Über FBO 1C32: SIX1 wird eingestellt, welche Synchronisierungsart gewünscht ist:

Wert 0_{hex} \Rightarrow freerun, nicht synchronisiert

Wert 2_{hex} \Rightarrow DC Sync0, synchronisiert mit DC IRQ Sync0

Wert 22_{hex} \Rightarrow SyncSM2, synchronisiert mit SyncManager IRQ des SM2 (SyncManager2 RxPDO)

Alle anderen Synchronisierungsarten werden nicht unterstützt. Wird trotzdem versucht darauf zu schreiben wird eine Fehlermeldung (06010000_{hex} = Fehler im Datenformat) generiert.

Die Zykluszeit muss über FBO $1C32_{\text{hex}}$:SIX2 oder über Register $9A0_{\text{hex}}$ im ECT eingestellt werden (DWORD in ns). Der Master muss dafür sorgen, dass die Sollwert-Telegramme bei eingestellten DC 200 μs bis 50 μs vor dem Sync-Event nicht gesendet werden. Mit Hilfe des Sync-Offsets in der WinBASS II / ProDrive Seite „Synchronisieren“ ist es möglich diesen „verbotenen Bereich“ zu verschieben, wenn der Master keine Möglichkeit hat die Sollwert-Telegramme aus dem „verbotenen Bereich“ zu schieben.

Wird in beiden Fällen keine Zykluszeit eingestellt, wird die Zykluszeit entsprechend der Seite Synchronisierung in WinBASS II / ProDrive übernommen.



HINWEIS!

Es kann eine andere Zykluszeit im Regler (Datensatz) eingestellt sein als gewünscht.

Wenn die DC aktiviert werden, wird durch die Optionskarte (im FPGA) die „PLL“ deaktiviert und das Sync0 Signal direkt von den DC zum Regler weitergegeben. Zusätzlich wird überwacht, ob Sollwertausfälle eintreten. Wenn in einem Zyklus kein Sollwert empfangen wird, wird eine EMCY abgesetzt (8100_{hex} XX 02_{hex} AA_{hex} 00_{hex} 00_{hex}).

Beim Übergang nach SAFE-OPERATIONAL wird überprüft ob im Register 981_{hex} die DC aktiviert wurden. Wenn nicht wird FBO $1C32_{\text{hex}}$: SIX1 auf den Wert 22_{hex} (Synchronisierung auf SM2) zurückgesetzt.

Notwendig für die DC ist eine FPGA Version größer B20, in der ESD Datei ETC03_121.eds ist das FBO $1C321_{\text{hex}}$ enthalten.

5.3.3 Beispiel für das PDO-Mapping

Das Optionsmodul CoE-Slave mit der Knotenadresse 1 empfängt vom Master einen Drehzahl Sollwert in RX-PDO1. Dieser Drehzahl Sollwert soll auf den Hochlaufgeber-Eingang geschrieben werden. Weiterhin erhält Knoten 1 in seiner RX-PDO1 das Steuerwort vom Master. Als Istwerte sendet der Knoten 1 seinen Drehzahl-Istwert und das Statuswort. Die Zykluszeit soll 1 ms betragen und auf DC oder auf SM2 synchronisiert werden. Der Slave wird mit Hilfe des Masters aus der Init Phase in den Zustand PRE-OPERATIONAL gebracht. Dies erfolgt in CoE mit den definierten AL Event Mechanismus.

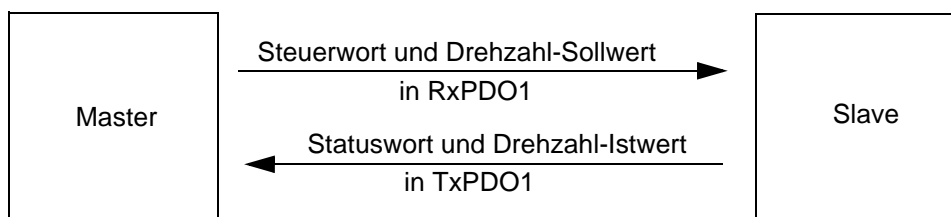


Abbildung 13: Beispiel-Mapping mit einem b maXX®

1. Schritt: Bestimmen der notwendigen Objekte

Ermitteln Sie aus der Objektliste (siehe [▶Anhang C - Umsetzungstabellen◀](#) ab Seite 75 und [▶Verzeichnis der Objekte zur Kommunikationssteuerung◀](#) ab Seite 35) die entsprechende Objekte des Objektverzeichnisses.

Für die Geräte sind folgende Parameter, die mit den angegebenen Objekten korrespondieren, relevant:

Parameternummer, z.B. beim b maXX® 4400	CANopen Feldbus-Objekt
P0301 Statuswort	↔ 6041 _{hex} Statuswort
P0300 Steuerwort	↔ 6040 _{hex} Steuerwort
P1171 Sollwertauswahl HLG Eingang	↔ 6042 _{hex} Drehzahl Sollwert am HLG
P0353 Drehzahl-Istwert	↔ 6044 _{hex} Control Effort

2. Schritt: Mapping konfigurieren

Schreiben des ersten zu mappenden Objektes mit Index (6041_{hex}), Subindex (00_{hex}) und Länge (10_{hex}) auf 1A00_{hex} Subindex 01_{hex} (TxPDO 1).

Schreiben des zweiten zu mappenden Objektes mit Index (6044_{hex}), Subindex (00_{hex}) und Länge (10_{hex}) auf 1A00_{hex} Subindex 02_{hex} (TxPDO 1).

Schreiben der Anzahl der gemappten Objekte (02_{hex}) auf 1A00_{hex} Subindex 00_{hex} (TxPDO 1).

Der Inhalt von Objekt 1A00_{hex} sieht wie folgt aus:

1A00 _{hex}	00 _{hex}	02 _{hex}
	01 _{hex}	60410010 _{hex}
	02 _{hex}	60440010 _{hex}

Schreiben des ersten zu mappenden Objektes mit Index (6040_{hex}), Subindex (00_{hex}) und Länge (10_{hex}) auf 1600_{hex} Subindex 01_{hex} (RxPDO 1).

Schreiben des zweiten zu mappenden Objektes mit Index (6042_{hex}), Subindex (00_{hex}) und Länge (10_{hex}) auf 1600_{hex} Subindex 02_{hex} (RxPDO 1).

Schreiben der Anzahl der gemappten Objekte (02_{hex}) auf 1600_{hex} Subindex 00_{hex} (RxPDO 1).

Der Inhalt von Objekt 1600_{hex} sieht wie folgt aus:

1600 _{hex}	00 _{hex}	02 _{hex}
	01 _{hex}	60400010 _{hex}
	02 _{hex}	60420010 _{hex}

Jetzt muss noch die Zykluszeit von 1 ms eingestellt werden, sowie der Synchronisiermodus (SM2 oder Sync0). Dies geschieht über SDOs mit dem FBO 1C32_{hex} SIX1/2. Zusätzlich müssen beim Synchronisieren auf DC noch Einstellungen im EtherCAT ASIC vorgenommen werden. Siehe dazu [►Synchronisation \(SYNC\)◄](#) ab Seite 47.

3. Schritt: Synchronisierung

- Synchronisieren auf DC

FBO 1C32_{hex} SIX1 Type der Synchronisation:

Request	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
	2B _{hex}	32 _{hex}	1C _{hex}	01 _{hex}	02 _{hex}			

CS= 2B_{hex} für 1 Byte;
 32_{hex}1C_{hex} wird gekreuzt und setzt sich zusammen aus 1C32_{hex};
 SIX1=02_{hex} für den Sync Type DC.

Die Antwort darauf lautet:

Response	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
	60 _{hex}	32 _{hex}	1C _{hex}	01 _{hex}	02 _{hex}			

- Synchronisieren auf SM2

FBO 1C32_{hex} SIX1 Type der Synchronisation:

Request	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
	2B _{hex}	32 _{hex}	1C _{hex}	01 _{hex}	22 _{hex}			

CS= 2B_{hex} für 1 Byte;
 32_{hex}1C_{hex} wird gekreuzt und setzt sich zusammen aus 1C32_{hex};
 SIX1=22_{hex} für den Sync Type SyncManager2.

Die Antwort darauf lautet:

Response	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
	60 _{hex}	32 _{hex}	1C _{hex}	01 _{hex}	22 _{hex}			

Alternativ kann die Zykluszeit auch im Datensatz abgespeichert werden. Defaultmäßig wird dann das Synchronisieren auf SM2 eingestellt, das FBO 1C32_{hex} SIX1/2 wird dabei nicht benötigt.

5.3.4 Eintrag in die BACI

Es können maximal acht zyklische Soll- und acht zyklische Istwerte zwischen dem Optionsmodul CoE-Slave und dem b maXX[®]-Regler gleichzeitig ausgetauscht werden. Alle Werte werden in einem Zyklus aktualisiert. Die Soll-/Istwerte können bei CoE auf je eine PDO verteilt sein. Es können 8 mal 4 Byte je Richtung zyklisch übertragen werden.

Die Aktualisierungszeit für die Verarbeitung der PDOs im Regler ist abhängig von der im b maXX[®]-Regler eingestellten Kommunikationszeit (siehe Kommunikation zum b maXX[®]-Regler). Die Einträge erfolgen lückenfrei, mit dem 1. Objekt von PDO1 beginnend werden die Inhalte abwechselnd auf ihre Gültigkeit für die BACI-Konfiguration (kein Dummy) abgefragt. Ist das Objekt gültig, dann wird dieses an die nächste freie Stelle der BACI-Konfiguration eingetragen. Ist das PDO-Mapping fehlerhaft (falsche Parameternummern o.ä.), wird keine zyklische Kommunikation zwischen Optionskarte und b maXX[®]-Regler gestartet.



HINWEIS!

Werden in dem vorhandenen PDO gleicher Richtung mehrmals identische Objektnummern gemappt, so erscheint das Objekt nur einmal in der BACI-Konfiguration. Hierbei muss beachtet werden, dass sich die Objekte im Mapping unter Umständen gegenseitig beeinflussen.



HINWEIS!

Das Dummy-Objekt wird in der BACI-Initialisierung nicht berücksichtigt.

5.4 Fehlertelegramm (EMCY)

Emergency-Telegramme bei den zyklischen Daten dienen der Anzeige von b maXX[®]-Fehlern. Sobald der b maXX[®]-Regler einen internen Fehler erkannt hat wird ein EMCY-Telegramm gesendet. Bei jedem neu hinzukommenden Fehler, wenn die Fehlernummer kleiner ist als der vorher angezeigte Wert, wird einmalig ein EMCY-Telegramm gesendet. Eine Telegrammwiederholung erfolgt nicht. Das EMCY-Telegramm wird in CoE über den Mailboxmechanismus gesendet. Der Master fragt dabei z.B. alle 20 ms nach (vom Master abhängig), ob ein EMCY vom Slave in die Sende Mailbox eingetragen wurde.

5.4.1 Telegrammaufbau

Der Nutzdatenbereich des Emergency-Telegramms gliedert sich in drei Teile:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Emergency Error Code		Error Register	Herstellerspezifisches Fehlerfeld				

Der Emergency Error Code (Byte 0, 1) ist in CANopen DSP 402 definiert. Die Umsetzung auf b maXX[®]-Fehlernummern wird in [►Umsetzung der Fehlermeldungen nach DSP 402 V1.1◀](#) ab Seite 54 dargestellt.

Das Error Register ist wie folgt definiert.

Bit	Bedeutung
0	Fehler aufgetreten, allgemeiner Fehler
1	Stromfehler
2	Spannungsfehler
3	Temperaturfehler
4	Kommunikationsfehler
5	Gerätespezifischer Fehler
6	Nicht verwendet
7	Herstellerspezifischer Fehler

Byte 3: enthält die Anzahl der Bytes des folgenden Emergency Codes, bei den gerätespezifischen Fehler ist das die Länge 2 Byte.

Byte 4 und 5 des herstellerspezifischen Fehlerfeldes enthalten die b maXX[®]-Fehlernummer.

Beispiel

Slave 5 hat einen Geberfehler an Geber 1 erkannt (Leitungsbruch Geber 1). Das EMCY-Telegramm hat dann folgende Form:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
00 _{hex}	73 _{hex}	81 _{hex}	2 _{hex}	73 _{hex}	00 _{hex}	00 _{hex}	00 _{hex}

Stehen mehrere Fehler an und wird ein Fehler gelöscht, sendet das Optionsmodul CoE-Slave das EMCY-Telegramm mit der nächsten Fehlernummer. Sind alle Fehler quittiert, wird vom Slave das Telegramm „Error Reset / No Error“ gesendet. Dabei sind alle Bytes des Telegramms mit 0 belegt.

5.4 Fehlertelegramm (EMCY)

5.4.2 Umsetzung der Fehlermeldungen nach DSP 402 V1.1

Die Beschreibung der Regler-Fehlermeldungen und Hinweise zur Beseitigung der Störungen können Sie der Betriebsanleitung b maXX[®] entnehmen. Die folgende Tabelle zeigt die Umsetzung der Regler-Fehlermeldungen auf CANopen-Fehlermeldungen.

Regler Fehlercode	Bezeichnung (vom b maXX [®] -Regler)	CANopen Fehlercode
0000 _{hex}	Reserve	--
0001 _{hex}	Watchdog-Fehler	7000 _{hex}
0002 _{hex}	Falscher oder unerwarteter Interrupt ist aufgetreten	7000 _{hex}
0003 _{hex}	NMI-Interrupt ist aufgetreten - falscher Buszugriff	7000 _{hex}
0010 _{hex}	Fehler beim Booten des Systems	5000 _{hex}
0011 _{hex}	Softwarefehler (u.a. switch)	5000 _{hex}
0012 _{hex}	Fehler beim Konfigurieren des Zeitscheiben-Betriebssystems	7000 _{hex}
0013 _{hex}	Zeitscheiben-Zeitverletzung	7000 _{hex}
0014 _{hex}	kein Speicher mehr frei	7000 _{hex}
0015 _{hex}	Software-Fehler: ungültiger Fehlercode	7000 _{hex}
0016 _{hex}	Software-Fehler: ungültiger Warnungscode	7000 _{hex}
0017 _{hex}	FPGA-Version ist nicht kompatibel zur Firmware	7000 _{hex}
0020 _{hex}	Timeout ProProg-Protokoll	8100 _{hex}
0021 _{hex}	Protokollfehler	7000 _{hex}
0022 _{hex}	Falscher Modultyp	7000 _{hex}
0023 _{hex}	Zu viele Daten in Liste bzw. Telegramm	7000 _{hex}
0024 _{hex}	Zu wenig Daten in Liste bzw. Telegramm	7000 _{hex}
0025 _{hex}	Ungültiger Operand	7000 _{hex}
0026 _{hex}	Gerät unterstützt nur VARSTAT_MEMORY	7400 _{hex}
0027 _{hex}	Ungültige Operandenadresse (log. Adresse)	7000 _{hex}
0028 _{hex}	Wert kleiner als Minimalwert	7000 _{hex}
0029 _{hex}	Wert größer als Maximalwert	7000 _{hex}
002A _{hex}	Parameter ist read-only	7000 _{hex}
002B _{hex}	Parameter kann aufgrund Betriebszustand nicht geändert werden	7000 _{hex}
002C _{hex}	Parameterwert ist ungültig	7000 _{hex}
002D _{hex}	WinBASS II / ProDrive ist nicht mehr angeschlossen oder reagiert nicht	7000 _{hex}

Regler Fehlercode	Bezeichnung (vom b maXX [®] -Regler)	CANopen Fehlercode
0030 _{hex}	Fehler im SmallModule_A (um die genaue Fehlernummer zu erfahren, muss der Parameter P0240 im b maXX [®] gelesen werden, die Fehlerbezeichnung des P0240 ist ab ▶Seite 61◀ beschrieben)	FF00 _{hex}
0031 _{hex}	Fehler im SmallModule_B (um die genaue Fehlernummer zu erfahren, muss der Parameter P0240 im b maXX [®] gelesen werden, die Fehlerbezeichnung des P0241 ist ab ▶Seite 61◀ beschrieben)	FF00 _{hex}
0032 _{hex}	Fehler im SmallModule_C (um die genaue Fehlernummer zu erfahren, muss der Parameter P0242 im b maXX [®] gelesen werden, die Fehlerbezeichnung des P0240 ist ab ▶Seite 61◀ beschrieben)	FF00 _{hex}
0033 _{hex}	Fehler im SmallModule_D (um die genaue Fehlernummer zu erfahren, muss der Parameter P0243 im b maXX [®] gelesen werden, die Fehlerbezeichnung des P0240 ist ab ▶Seite 61◀ beschrieben)	FF00 _{hex}
0034 _{hex}	Fehler im SmallModule_F (um die genaue Fehlernummer zu erfahren, muss der Parameter P0244 im b maXX [®] gelesen werden, die Fehlerbezeichnung des P0240 ist ab ▶Seite 61◀ beschrieben)	FF00 _{hex}
0035 _{hex}	Fehler im BigModule_G (um die genaue Fehlernummer zu erfahren, muss der Parameter P0245 im b maXX [®] gelesen werden, die Fehlerbezeichnung des P0240 ist ab ▶Seite 61◀ beschrieben)	FF00 _{hex}
0036 _{hex}	Fehler im BigModule_H (um die genaue Fehlernummer zu erfahren, muss der Parameter P0246 im b maXX [®] gelesen werden, die Fehlerbezeichnung des P0240 ist ab ▶Seite 61◀ beschrieben)	FF00 _{hex}
0037 _{hex}	Fehler im BigModule_J (um die genaue Fehlernummer zu erfahren, muss der Parameter P0247 im b maXX [®] gelesen werden, die Fehlerbezeichnung des P0240 ist ab ▶Seite 61◀ beschrieben)	FF00 _{hex}
0038 _{hex}	Fehler im BigModule_K (um die genaue Fehlernummer zu erfahren, muss der Parameter P0248 im b maXX [®] gelesen werden, die Fehlerbezeichnung des P0240 ist ab ▶Seite 61◀ beschrieben)	FF00 _{hex}
0039 _{hex}	Fehler im BigModule_L (um die genaue Fehlernummer zu erfahren, muss der Parameter P0249 im b maXX [®] gelesen werden, die Fehlerbezeichnung des P0240 ist ab ▶Seite 61◀ beschrieben)	FF00 _{hex}
003A _{hex}	Fehler im BigModule_M (um die genaue Fehlernummer zu erfahren, muss der Parameter P0250 im b maXX [®] gelesen werden, die Fehlerbezeichnung des P0240 ist ab ▶Seite 61◀ beschrieben)	FF00 _{hex}
003B _{hex}	Timeout beim Systemhochlauf	7000 _{hex}

Regler Fehlercode	Bezeichnung (vom b maXX [®] -Regler)	CANopen Fehlercode
003C _{hex}	CRC-Fehler in SPI-Übertragung Modul ⇒ Regler	7000 _{hex}
003D _{hex}	CRC-Fehler in SPI-Übertragung Regler ⇒ Modul	7000 _{hex}
0040 _{hex}	Netzausfall	3100 _{hex}
0041 _{hex}	Phasenausfall	3100 _{hex}
0042 _{hex}	Unterspannung Netz	3100 _{hex}
0043 _{hex}	Überspannung Netz	3100 _{hex}
0044 _{hex}	Unterspannung 24 V	3100 _{hex}
0045 _{hex}	Drehfeldererkennungsfehler	3100 _{hex}
0046 _{hex}	Frequenzänderungsfehler	3100 _{hex}
0047 _{hex}	Frequenzbereichfehler	5000 _{hex}
0048 _{hex}	Störung/Defekt am Schütz (Rückmeldung obwohl nicht angesteuert)	5000 _{hex}
0049 _{hex}	Keine Rückmeldung vom Schütz	3100 _{hex}
004A _{hex}	Störung bei Uzk-Aufbau	3100 _{hex}
004B _{hex}	Unterspannung Uzk	3100 _{hex}
004C _{hex}	Netzanschlussfehler	3100 _{hex}
004D _{hex}	Stromgrenze erreicht	3100 _{hex}
004E _{hex}	Synchronisationsfehler	3100 _{hex}
0050 _{hex}	Kommunikationsfehler nach Hiperface-Spezifikation (um die dazugehörige Fehlernummer zu erfahren muss der Parameter P0233 im b maXX [®] gelesen werden, die Fehler-Bezeichnung ist im Anschluss ab ▶Seite 61◀ beschrieben)	FF00 _{hex}
0051 _{hex}	Temperaturschwelle des Kühlkörpers ist überschritten	4200 _{hex}
0052 _{hex}	Überspannung Uzk	3200 _{hex}
0053 _{hex}	Überstrom Leistungsteil	2300 _{hex}
0054 _{hex}	Erdstrom	2200 _{hex}
0055 _{hex}	Temperaturschwelle der Innenluft ist überschritten	4200 _{hex}
0056 _{hex}	Ixt Grenzwert	FF00 _{hex}
0057 _{hex}	Sicherheitsrelais aus (bzw. Sicherheitsrelais defekt)	5000 _{hex}
0058 _{hex}	Sicherheitsrelais aus (Sicherheitsrelais O.K., aber Spannung nicht vorhanden)	5000 _{hex}
0059 _{hex}	Leistungsteil nicht betriebsbereit	5000 _{hex}
005A _{hex}	Phasenausfall	3100 _{hex}

Diese Fehler werden **nur** vom b maXX[®] 4100 unterstützt.

Regler Fehlercode	Bezeichnung (vom b maXX [®] -Regler)	CANopen Fehlercode
005B _{hex}	Netzausfall	3100 _{hex}
005C _{hex}	Unterspannung Netz	3100 _{hex}
005D _{hex}	Überspannung Netz	3100 _{hex}
005E _{hex}	Unterspannung Uzk	3100 _{hex}
0060 _{hex}	Temperaturfühler des Motors kurzgeschlossen (T _M ≤ -30 °C)	4300 _{hex}
0061 _{hex}	Temperaturfühler des Motors nicht angeschlossen (T _M > +300 °C)	4300 _{hex}
0062 _{hex}	Fehler Motortemperatur - Abschaltschwelle überschritten	4300 _{hex}
0063 _{hex}	Fehler I ² t > 100 % im Motor	7000 _{hex}
0070 _{hex}	Kommunikationsfehler nach Hiperface-Spezifikation (um die dazugehörige Fehlernummer zu erfahren muss der Parameter P0234/P0235 im b maXX [®] gelesen werden, die Fehler-Bezeichnung ist im Anschluss ab ▶Seite 61◀ beschrieben)	FF00 _{hex}
0071 _{hex}	Modulcode ungültig	7000 _{hex}
0072 _{hex}	Fehler beim Schreiben der Geberlage	7000 _{hex}
0073 _{hex}	Leitungsbruch Geber 1	7000 _{hex}
0074 _{hex}	Überdrehzahl Geber 1	7000 _{hex}
0075 _{hex}	Amplitudengrenze überschritten	7000 _{hex}
0076 _{hex}	Gebertyp unbekannt	7000 _{hex}
0077 _{hex}	Datenfeld für Motordaten ungültig	7000 _{hex}
0078 _{hex}	Motordaten ungültig	7000 _{hex}
0079 _{hex}	Fehler beim Speichern der Motordaten	7000 _{hex}
007A _{hex}	Motordaten schreibgeschützt (nicht BM-Motoren)	7000 _{hex}
007B _{hex}	Feldwinkel Fehler	7000 _{hex}
007C _{hex}	Geber ohne Temperaturerfassung	7000 _{hex}
0080 _{hex}	Kommunikationsfehler nach Hiperface-Spezifikation (um die dazugehörige Fehlernummer zu erfahren muss der Parameter P0235 im b maXX [®] gelesen werden, die Fehler-Bezeichnung ist im Anschluss ab ▶Seite 61◀ beschrieben)	FF00 _{hex}
0081 _{hex}	Modulcode ungültig	7000 _{hex}
0082 _{hex}	Fehler beim Schreiben der Geberlage	7000 _{hex}
0083 _{hex}	Leitungsbruch Geber 2	7000 _{hex}
0084 _{hex}	Überdrehzahl Geber 2	7000 _{hex}
0085 _{hex}	Amplitudengrenze überschritten	7000 _{hex}

Regler Fehlercode	Bezeichnung (vom b maXX [®] -Regler)	CANopen Fehlercode
0086 _{hex}	Gebertyp unbekannt	7000 _{hex}
0087 _{hex}	Datenfeld für Motordaten ungültig	7000 _{hex}
0088 _{hex}	Motordaten ungültig	7000 _{hex}
0089 _{hex}	Fehler beim Speichern der Motordaten	7000 _{hex}
008A _{hex}	Motordaten schreibgeschützt (nicht BM-Motoren)	7000 _{hex}
008B _{hex}	Feldwinkel Fehler	7000 _{hex}
008C _{hex}	Geber ohne Temperaturerfassung	7000 _{hex}
0090 _{hex}	Absolutlage des Geber nicht bekannt	7000 _{hex}
0091 _{hex}	Absolutlage des Geber 2 nicht bekannt	7000 _{hex}
0092 _{hex}	Geber Modul 1 fehlt	7000 _{hex}
0093 _{hex}	Geber Modul 2 fehlt	7000 _{hex}
0094 _{hex}	Geber Modul fehlt für Messwertspeicherung	7000 _{hex}
0095 _{hex}	Bei Resolver keine Messwertspeicherung möglich	7000 _{hex}
0096 _{hex}	Triggerung auf Nullimpuls und Geber ist kein Inkrementalgeber	7000 _{hex}
0097 _{hex}	Digital I/O-Modul wird benötigt und fehlt	7000 _{hex}
0098 _{hex}	Inkrementalgeber-Nachbildungsmodul wird benötigt und fehlt	7000 _{hex}
0099 _{hex}	Gebermodul 1 wird für Inkrementalgeber-Nachbildung benötigt und fehlt	7000 _{hex}
009A _{hex}	Gebermodul 2 wird für Inkrementalgeber-Nachbildung benötigt und fehlt	7000 _{hex}
009B _{hex}	Fehler beim Initialisieren des Inkrementalgeber-Nachbildungsmoduls	7000 _{hex}
009C _{hex}	Inkrementalgeber-Nachbildungsmodul meldet Fehler	7000 _{hex}
009D _{hex}	Inkrementalgeber-Nachbildung: Option „Start nach erstem Nullpuls“ für Nicht-Inkrementalgeber gewählt	7000 _{hex}
009E _{hex}	SSI-Geber-Nachbildung Modul fehlt	7000 _{hex}
009F _{hex}	Fehler in Sollwertquelle Geber 1 bzw. Geber 2	7000 _{hex}
00A0 _{hex}	Zeitüberwachung Proprog Kommunikation	8100 _{hex}
00A1 _{hex}	Zeitüberwachung BACI Kommunikation	8100 _{hex}
00A2 _{hex}	Zeitüberwachung zyklische Kommunikation	8110 _{hex}
00A3 _{hex}	Zeitüberwachung Bedarfsdaten Übertragung	8100 _{hex}
00A4 _{hex}	Feldbus Fehler	8100 _{hex}
00A5 _{hex}	Regler nicht synchron zu externem Signal	8100 _{hex}

Regler Fehlercode	Bezeichnung (vom b maXX [®] -Regler)	CANopen Fehlercode
00A6 _{hex}	Fehler bei der Bremsansteuerung	8100 _{hex}
00B0 _{hex}	Kopierfehler EEPROM	5000 _{hex}
00B1 _{hex}	Zeitüberschreitung beim EEPROM- Beschreiben	5000 _{hex}
00B2 _{hex}	Checksummenfehler im EEPROM	5000 _{hex}
00B3 _{hex}	Kein Bootdatensatz	5000 _{hex}
00B4 _{hex}	Inkompatibel SW	5000 _{hex}
00B5 _{hex}	Datensatzumschaltung: DS nicht vorhanden	5530 _{hex}
00B6 _{hex}	Checksummenfehler im PSI	5000 _{hex}
00B7 _{hex}	PSI ist gelöscht	5000 _{hex}
00B8 _{hex}	PSI-Daten sind ungültig	5000 _{hex}
00B9 _{hex}	Selbstoptimierungstabellen ungültig - Selbstoptimierung erneut durchführen	5000 _{hex}
00BA _{hex}	A/D-Korrekturtabelle ungültig	5000 _{hex}
00C0 _{hex}	Schleppfehler dynamisch	8000 _{hex}
00C1 _{hex}	Schleppfehler statisch	8000 _{hex}
00C2 _{hex}	Geber 1 für Lageregelung verwendet, aber nicht aktiv	7300 _{hex}
00C3 _{hex}	Geber 2 für Lageregelung verwendet, aber nicht aktiv	7300 _{hex}
00C4 _{hex}	Software-Endschalterüberwachung 1 aktiv	8600 _{hex}
00C5 _{hex}	Software-Endschalterüberwachung 2 aktiv	8600 _{hex}
00C6 _{hex}	Hardware-Endschalterüberwachung 1 aktiv	8600 _{hex}
00C7 _{hex}	Hardware-Endschalterüberwachung 2 aktiv	8600 _{hex}
00C8 _{hex}	Positionierung ohne Referenzfahrt gestartet	8600 _{hex}
00C9 _{hex}	Sollwert im Modus Set-of-setpoints nicht rechtzeitig angekommen	8600 _{hex}
00CA _{hex}	Überwachung der Modulo-Position aktiv: Zielposition > Modulo-Position	8600 _{hex}
00CB _{hex}	Spindelpositionierung: Fehler bei Initialisierung des Triggers	8600 _{hex}
00CC _{hex}	Spindelpositionierung: Timeout beim Triggersignal (Nullimpuls/Schalteingang)	8600 _{hex}
00D0 _{hex}	Antrieb blockiert	7000 _{hex}
00D1 _{hex}	Maximaldrehzahl erreicht	7000 _{hex}

5.4 Fehlertelegramm (EMCY)

Regler Fehlercode	Bezeichnung (vom b maXX [®] -Regler)	CANopen Fehlercode
00D2 _{hex}	Geber 2 für Motorführung verwendet aber nicht aktiv	7000 _{hex}
00D3 _{hex}	Überdrehzahl Open Loop	7000 _{hex}
<p>Die herstellerspezifischen Error Codes 0030_{hex} bis 003B_{hex}, 0050_{hex}, 0070_{hex} und 0080_{hex} werden zusammenfassend über ein EMY- Telegramm mit dem CANopen Error Code FF00_{hex} ausgegeben. Die genaue Bezeichnung dafür kann in den folgenden Parameter P0233, P0234, P0235 und P0240 bis P0250 abgelesen werden und die Beschreibung zu den ausgelesenen Fehlernummern ist nachfolgend aufgezeichnet.</p>		

Regler Fehlercode	Bezeichnung (vom b maXX [®] -Regler)	CANopen Fehlercode
<p>Die nachfolgenden b maXX[®] Error Codes werden n i c h t über ein EMY- Telegramm ausgegeben.</p> <p>Error Code (0050_{hex}) ⇒ P0233 Kommunikationsfehler nach Hiperface-Spezifikation (AmpHiperfaceError)</p>		
06 _{hex}	Datenüberlauf	
07 _{hex}	Bitrahmen- Fehler	
08 _{hex}	Ungültiger Kommandozustand	
09 _{hex}	Parity- Fehler	
0A _{hex}	Check-Summe der übertragenen Daten ist falsch	
0B _{hex}	Unbekannte Befehlscode	
0C _{hex}	Anzahl der übertragenen Daten falsch	
0D _{hex}	Unzulässiges Argument	
0E _{hex}	Datenfeld ist nicht beschreibbar	
0F _{hex}	Falscher Zugriffcode	
10 _{hex}	Datenfeld in seiner Größe nicht veränderbar	
11 _{hex}	Angegebene Wortadresse außerhalb Datenfeld	
12 _{hex}	Zugriff auf nicht existierendes Datenfeld	
24 _{hex}	Falsche LT-Daten Checksumme	
25 _{hex}	keine Antwort vom LT	
42 _{hex}	Ungültige Antwort	
<p>Error Code (0070_{hex}) ⇒ P0234 Kommunikationsfehler nach Hiperface-Spezifikation (Enc1HiperfaceError)</p> <p>Error Code (0080_{hex}) ⇒ P0235 Kommunikationsfehler nach Hiperface-Spezifikation (Enc2HiperfaceError)</p>		
01 _{hex}	Analogsignale außerhalb Spezifikation	
02 _{hex}	Interner Winkeloffset fehlerhaft	
03 _{hex}	Tabelle über Datenfeldpartitionierung zerstört	
04 _{hex}	Analoge Grenzwerte nicht verfügbar	
05 _{hex}	Interner I2C-Bus nicht funktionsfähig	
06 _{hex}	Interner Checksummenfehler	
07 _{hex}	Interner Watchdog-Fehler - Geber-Reset	
09 _{hex}	Parity- Fehler	
0A _{hex}	Checksumme der übertragenen Daten ist falsch	
0B _{hex}	Unbekannte Befehlscode	
0C _{hex}	Anzahl der übertragenen Daten falsch	

5.4 Fehlertelegramm (EMCY)

Regler Fehlercode	Bezeichnung (vom b maXX [®] -Regler)	CANopen Fehlercode
0D _{hex}	Unzulässiges Argument	
0E _{hex}	Datenfeld ist nicht beschreibbar	
0F _{hex}	Falscher Zugriffscode	
10 _{hex}	Datenfeld in seiner Größe nicht veränderbar	
11 _{hex}	Angegebene Wortadresse außerhalb Datenfeld	
12 _{hex}	Zugriff auf nicht existierendes Datenfeld	
1C _{hex}	Betragsüberwachung der Analogsignale	
1D _{hex}	Sendestrom kritisch	
1E _{hex}	Gebertemperatur kritisch	
1F _{hex}	Drehzahl zu hoch - keine Positionsbildung möglich	
20 _{hex}	Position Singleturn unzulässig	
21 _{hex}	Positionsfehler Multiturn	
22 _{hex}	Positionsfehler Multiturn	
23 _{hex}	Positionsfehler Multiturn	
24 _{hex}	Falsche MT-Daten Checksumme	
40 _{hex}	Keine Antwort vom Hiperface-Geber	
41 _{hex}	Keine Antwort vom EnDatGeber	
42 _{hex}	unbrauchbare Antwort auf Geberkommando	
50 _{hex}	CRC hat Fehler festgestellt	
51 _{hex}	Ungültiges Kommando	
52 _{hex}	Adresse bzw. MRS-Code im Antwort-Telegramm falsch	
53 _{hex}	Alarmbit des Gebers ist gesetzt	
54 _{hex}	Speicher im Geber ist belegt	
55 _{hex}	Checksummenfehler beim Lesen der Motordaten	
56 _{hex}	Motordaten-Datenlänge und/oder Datenversion von Geber und Reglerfirmware sind nicht identisch	
57 _{hex}	Einschalttest hat keine EnDat-Schnittstelle am Geber festgestellt	
58 _{hex}	Überschreitung des auswertbaren Übertragungsformates	
59 _{hex}	Überschreitung der auswertbaren Messschrittlänge	
5A _{hex}	Signalperiodenlänge < Messschrittlänge	
60 _{hex}	Fehler Beleuchtung	
61 _{hex}	Fehler Signalamplitude	
62 _{hex}	Fehler Positionswert	

Regler Fehlercode	Bezeichnung (vom b maXX [®] -Regler)	CANopen Fehlercode
63 _{hex}	Fehler Überspannung	
64 _{hex}	Fehler Unterspannung	
65 _{hex}	Fehler Überstrom	
66 _{hex}	Fehler Batterie	
Error Code (0030 _{hex} 0034 _{hex}) ⇒ P0240...P0244 Fehler im SmallModule 1 bis 5		
01 _{hex}	Modul nicht erkannt	
02 _{hex}	Erkannter Modul am unzulässigen Platz	
03 _{hex}	Digitaler Ausgang kurzgeschlossen	
04 _{hex}	Falscher Zielparameterwert durch digitalen Eingang	
05 _{hex}	Direkter PLC-IO-Zugriff für dieses Modul nicht erlaubt	
07 _{hex}	Modul im Regler nicht erlaubt	
Error Code (0035 _{hex} 0040 _{hex}) ⇒ P0245...P0250 Fehler im BigModule 1 bis 6		
1000 _{hex}	Falsche Parameter-Nr. bei Sollwert Parameter 1	
1001 _{hex}	Falsche Parameter-Nr. bei Sollwert Parameter 2	
1002 _{hex}	Falsche Parameter-Nr. bei Sollwert Parameter 3	
1003 _{hex}	Falsche Parameter-Nr. bei Sollwert Parameter 4	
1004 _{hex}	Falsche Parameter-Nr. bei Sollwert Parameter 5	
1005 _{hex}	Falsche Parameter-Nr. bei Sollwert Parameter 6	
1006 _{hex}	Falsche Parameter-Nr. bei Sollwert Parameter 7	
1007 _{hex}	Falsche Parameter-Nr. bei Sollwert Parameter 8	
1008 _{hex}	Falsche Parameter-Nr. bei Sollwert Parameter 9	
1009 _{hex}	Falsche Parameter-Nr. bei Sollwert Parameter 10	
100A _{hex}	Falsche Parameter-Nr. bei Sollwert Parameter 11	
100B _{hex}	Falsche Parameter-Nr. bei Sollwert Parameter 12	
100C _{hex}	Falsche Parameter-Nr. bei Sollwert Parameter 13	
100D _{hex}	Falsche Parameter-Nr. bei Sollwert Parameter 14	
100E _{hex}	Falsche Parameter-Nr. bei Sollwert Parameter 15	
100F _{hex}	Falsche Parameter-Nr. bei Sollwert Parameter 16	
1010 _{hex}	Falsche Parameter-Nr. bei Istwert Parameter 1	
1011 _{hex}	Falsche Parameter-Nr. bei Istwert Parameter 2	
1012 _{hex}	Falsche Parameter-Nr. bei Istwert Parameter 3	
1013 _{hex}	Falsche Parameter-Nr. bei Istwert Parameter 4	
1014 _{hex}	Falsche Parameter-Nr. bei Istwert Parameter 5	

5.4 Fehlertelegramm (EMCY)

Regler Fehlercode	Bezeichnung (vom b maXX [®] -Regler)	CANopen Fehlercode
1015 _{hex}	Falsche Parameter-Nr. bei Istwert Parameter 6	
1016 _{hex}	Falsche Parameter-Nr. bei Istwert Parameter 7	
1017 _{hex}	Falsche Parameter-Nr. bei Istwert Parameter 8	
1018 _{hex}	Falsche Parameter-Nr. bei Istwert Parameter 9	
1019 _{hex}	Falsche Parameter-Nr. bei Istwert Parameter 10	
101A _{hex}	Falsche Parameter-Nr. bei Istwert Parameter 11	
101B _{hex}	Falsche Parameter-Nr. bei Istwert Parameter 12	
101C _{hex}	Falsche Parameter-Nr. bei Istwert Parameter 13	
101D _{hex}	Falsche Parameter-Nr. bei Istwert Parameter 14	
101E _{hex}	Falsche Parameter-Nr. bei Istwert Parameter 15	
101F _{hex}	Falsche Parameter-Nr. bei Istwert Parameter 16	
1020 _{hex}	Ungültiger Wert bei Sollwert-Parameter-Nr. 1	
1021 _{hex}	Ungültiger Wert bei Sollwert-Parameter-Nr. 2	
1022 _{hex}	Ungültiger Wert bei Sollwert-Parameter-Nr. 3	
1023 _{hex}	Ungültiger Wert bei Sollwert-Parameter-Nr. 4	
1024 _{hex}	Ungültiger Wert bei Sollwert-Parameter-Nr. 5	
1025 _{hex}	Ungültiger Wert bei Sollwert-Parameter-Nr. 6	
1026 _{hex}	Ungültiger Wert bei Sollwert-Parameter-Nr. 7	
1027 _{hex}	Ungültiger Wert bei Sollwert-Parameter-Nr. 8	
1028 _{hex}	Ungültiger Wert bei Sollwert-Parameter-Nr. 9	
1029 _{hex}	Ungültiger Wert bei Sollwert-Parameter-Nr. 10	
102A _{hex}	Ungültiger Wert bei Sollwert-Parameter-Nr. 11	
102B _{hex}	Ungültiger Wert bei Sollwert-Parameter-Nr. 12	
102C _{hex}	Ungültiger Wert bei Sollwert-Parameter-Nr. 13	
102D _{hex}	Ungültiger Wert bei Sollwert-Parameter-Nr. 14	
102E _{hex}	Ungültiger Wert bei Sollwert-Parameter-Nr. 15	
102F _{hex}	Ungültiger Wert bei Sollwert-Parameter-Nr. 16	
1030 _{hex}	Ungültiger Wert für Sollwert-Periode	
1031 _{hex}	Ungültiger Wert für Istwert-Periode	
1032 _{hex}	Falscher Wert für Cycle-Offset Sollwerte	
1033 _{hex}	Falscher Wert für Cycle-Offset Istwerte	
1034 _{hex}	BACI-Timeout bei zyklischen Daten	
1035 _{hex}	BACI-Timeout bei Bedarfsdaten	
1036 _{hex}	Überprüfung ergab fehlerhafte Checksumme	

Regler Fehlercode	Bezeichnung (vom b maXX [®] -Regler)	CANopen Fehlercode
1037 _{hex}	Hochlauf: Timeout beim Warten auf Slave-Type bzw. beim Warten auf Rücksetzen von Config-Pending-Flag	
1038 _{hex}	Falscher Datentransfer-Struktur-Typ	
1039 _{hex}	Interner Fehler: Falscher BACI-Zustand	
103A _{hex}	Zugriffskonflikt mit Slave bei zyklischer Kommunikation	
103B _{hex}	Fehler zyklische Kommunikation: Parameterwert falsch	
103C _{hex}	Fehler zyklische Kommunikation: Alive-Counter Konflikt	
103D _{hex}	Cmd-Interface: Kanalnummer falsch (0 oder > 6)	
103E _{hex}	Cmd-Interface: Angegebener Kanal existiert nicht	
103F _{hex}	Cmd-Interface: Interner Fehler - Falscher Pointer	
1040 _{hex}	Cmd-Interface: Interner Fehler - Falscher Zustand	
1041 _{hex}	Cmd-Interface: Falsche Paketnummer	
1042 _{hex}	Cmd-Interface: Falsche Kommandonummer	
1043 _{hex}	Cmd-Interface: Falscher Zustand bei Pakethandling	
1044 _{hex}	Cmd-Interface: Timeout bei Kommandobearbeitung	
1045 _{hex}	Cmd-Interface: Falsche Paketlänge	
1046 _{hex}	Cmd-Interface: Kein Deskriptor mehr verfügbar (zu wenig Speicher)	
1047 _{hex}	Cmd-Interface: Falscher Paktetyp	
1048 _{hex}	Cmd-Interface: Checksummenfehler	
1049 _{hex}	Modulkennung: PCI-Fehler beim Lesen	
104A _{hex}	Modulkennung: PCI-Fehler beim Schreiben	
104B _{hex}	Modulkennung: allgemeiner Fehler beim Lesen	
104C _{hex}	Modulkennung: allgemeiner Fehler beim Schreiben	
104D _{hex}	Interner Fehler	
104E _{hex}	Konfiguration zyklische Dienste: Parameter nicht bzw. nicht cykl. beschreibbar	
104F _{hex}	Konfiguration zyklische Dienste: Ungültige Parameternummer	
1050 _{hex}	Falscher Optionsmodule Fehlercode (einstellbar mit P1007)	
2000 _{hex}	Fehler CANopen-Timeout auf CAN-Bus (Node Guarding)	



ANHANG A - ABKÜRZUNGEN

BACI	Baumüller Antriebe serielle Schnittstelle
DC	Distributed Clocks
EMCY	Fehlertelegramm
FMMU	Fieldbus Memory Management Unit
HD	Hamming Distanz
ID	Ident-Nummer
LMT	Layer-Management
M	Multiplexer
NMT	Netzwerk-Management
PC	Personal Computer
PDO	Prozessdaten
SDO	Servicedaten
SIX	Subindex
SYNC	Synchronisation



ANHANG B - KURZREFERENZ

Die folgende Kurzreferenz zeigt den Zusammenhang zwischen CANopen-Objektnummer und dem b maXX[®] Regler-Parameternummern (siehe Betriebsanleitung b maXX[®] 5.02017).

B.1 4000er Objektnummern (Herstellerspezifische Objekte)

Herstellerspezifische Objekte ergeben sich aus
 $4000_{\text{hex}} + \text{Parameternummer}_{\text{hex}}$
Der Subindex für alle 4000er-Parameter ist immer 00_{hex} .

Beispiel

Parameter **P0053** \Rightarrow Objektindex 4035_{hex} Subindex 00_{hex}

Die Informationen in kursiver Schrift sind ein Hinweis, dass dieser Parameter auch von einem 6000er-Parameter oder mehreren 6000er-Parametern beeinflusst werden kann. Weitere Hinweise finden Sie in ([>B.2 6000er Objektnummern \(Device Profile DSP 402\)](#)< ab Seite 70).

B.2 6000er Objektnummern (Device Profile DSP 402)

Auf manche Parameter des Reglers ist es möglich sowohl über ein oder mehrere 6000er als auch über 4000er Objekte zuzugreifen (siehe *kursiven* Text in [►B.1 4000er Objektnummern \(Herstellerspezifische Objekte\)](#) ab Seite 69).

Auf einige Parameter ist der Zugriff ausschließlich mit einem 6000er-Parameter möglich (606A_{hex}, 6048_{hex} SIX1, 6049_{hex} SIX1 und 604C_{hex} SIX1/2).

Es ist zu beachten, dass die Normierungen zwischen den herstellerspezifischen Objekten und den Device Profile Objekten in der Regel unterschiedlich sind. Werden beide Feldbusobjekte (über herstellerspezifisches Objekt und über Device Profile Objekt) für ein und denselben Parameter im b maXX[®]-Regler gemappt, so werden sich beide Feldbusobjekte gegenseitig beeinflussen. Dies sollte in jeden Fall vermieden werden, deshalb sollte nach Möglichkeit ein Zugriff auf die Objekte, die im Device Profile vorhanden sind nicht über herstellerspezifische Objekte erfolgen. Es gibt weiterhin FBOs, die auf zwei Regler-Parametern abgebildet werden.

TX: Transmit; RX: Receive; r: read; w: write; ro: read only; wo: write only

CANopen-Objekt-Nummer		Parameternummer	PDO-Mapping	Access - Typ	Betriebsart nach DSP 402
Index	Subindex				
6040 _{hex}	00 _{hex}	P0300	TX / RX	rw	Device Control
6041 _{hex}	00 _{hex}	P0301	TX	ro	Device Control
6042 _{hex}	00 _{hex}	P1171	TX / RX	rw	Velocity Mode
6043 _{hex}	00 _{hex}	P0351	TX	ro	Velocity Mode
6044 _{hex}	00 _{hex}	P0353	TX	ro	Velocity Mode
6046 _{hex}	01 _{hex}	P1041	TX	ro	Velocity Mode
6046 _{hex}	02 _{hex}	P1041, P1042	TX / RX	rw	Velocity Mode
6048 _{hex}	01 _{hex}		TX / RX	rw	Velocity Mode
6048 _{hex}	02 _{hex}	P1172	TX / RX	rw	Velocity Mode
6049 _{hex}	01 _{hex}		TX / RX	rw	Velocity Mode
6049 _{hex}	02 _{hex}	P1173	TX / RX	rw	Velocity Mode
604C _{hex}	01 _{hex}		TX / RX	rw	Velocity Mode
604C _{hex}	02 _{hex}		TX / RX	rw	Velocity Mode
604D _{hex}	00 _{hex}	P0065	TX	rw	Velocity Mode
605E _{hex}	00 _{hex}	P1007	TX		Device Control
604F _{hex}	00 _{hex}	P1172	TX / RX	rw	Velocity Mode
6050 _{hex}	00 _{hex}	P1173	TX / RX	rw	Velocity Mode
6051 _{hex}	00 _{hex}	P1174	TX / RX	rw	Velocity Mode

CANopen-Objekt-Nummer		Parameternummer	PDO-Mapping	Access - Typ	Betriebsart nach DSP 402
Index	Subindex				
605A _{hex}	00 _{hex}	P1004	TX	rw	Device Control
605B _{hex}	00 _{hex}	P1005	TX	rw	Device Control
605C _{hex}	00 _{hex}	P1006	TX	rw	Device Control
605D _{hex}	00 _{hex}	P1003	TX	rw	Device Control
6060 _{hex}	00 _{hex}	P1000	- / RX	wo	Device Control
6061 _{hex}	00 _{hex}	P0304	TX	ro	Device Control
6062 _{hex}	00 _{hex}	P0463	TX	ro	Position Control Function
6063 _{hex}	00 _{hex}	P0362	TX	ro	Position Control Function
6064 _{hex}	00 _{hex}	P0462	TX	ro	Position Control Function
6066 _{hex}	00 _{hex}	P1056	TX	rw	Position Control Function
6067 _{hex}	00 _{hex}	P1194	TX / RX	rw	Position Control Function
6068 _{hex}	00 _{hex}	P1195	TX	rw	Position Control Function
6069 _{hex}	00 _{hex}	P0362	TX / RX	rw	Profile Velocity Mode
606A _{hex}	00 _{hex}	-	-	ro	Profile Velocity Mode
606B _{hex}	00 _{hex}	P0352	TX	ro	Profile Velocity Mode
606C _{hex}	00 _{hex}	P0353	TX	ro	Profile Velocity Mode
606F _{hex}	00 _{hex}	P1073	TX / RX	rw	Profile Velocity Mode
6072 _{hex}	00 _{hex}	P0357	TX / RX	rw	Profile Torque Mode
607A _{hex}	00 _{hex}	P0600	TX / RX	rw	Profile Position Mode
607C _{hex}	00 _{hex}	P1200	TX / RX	rw	Homing Mode
607D _{hex}	01 _{hex}	P1196	TX	rw	Profile Position Mode
607D _{hex}	02 _{hex}	P1197	TX	rw	Profile Position Mode
607F _{hex}	00 _{hex}	P0057	TX	rw	Profile Position Mode
6080 _{hex}	00 _{hex}	P1031	TX	rw	Profile Position Mode
6081 _{hex}	00 _{hex}	P0602	TX	rw	Profile Position Mode
6083 _{hex}	00 _{hex}	P0603	TX	rw	Profile Position Mode
6084 _{hex}	00 _{hex}	P0604	TX	rw	Profile Position Mode
6085 _{hex}	00 _{hex}	P1213	TX	rw	Profile Position Mode
6086 _{hex}	00 _{hex}	P1190	TX	rw	Profile Position Mode
6092 _{hex}	01 _{hex}	P1193	TX	rw	Factor Group
6092 _{hex}	02 _{hex}	P3050	TX	rw	Factor Group

CANopen-Objekt-Nummer		Parameter-Nummer	PDO-Mapping	Access - Typ	Betriebsart nach DSP 402
Index	Subindex				
6098 _{hex}	00 _{hex}	P3051	TX	rw	Homing Mode
6099 _{hex}	01 _{hex}	P1201	TX / RX	rw	Homing Mode
6099 _{hex}	02 _{hex}	P1202	TX / RX	rw	Homing Mode
609A _{hex}	00 _{hex}	P1203	TX / RX	rw	Homing Mode
60F8 _{hex}	00 _{hex}	P1054	TX / RX	rw	Profile Velocity Mode
60FB _{hex}	01 _{hex}	P0360	TX	ro	Position Control Function
60FB _{hex}	02 _{hex}	P1050	TX	rw	Position Control Function
60FB _{hex}	03 _{hex}	P1051	TX	rw	Position Control Function
60FB _{hex}	04 _{hex}	P0364	TX / RX	rw	Position Control Function
60FB _{hex}	05 _{hex}	P0363	TX / RX	rw	Position Control Function
60FB _{hex}	06 _{hex}	P1053	TX	rw	Position Control Function
60FB _{hex}	07 _{hex}	P0367	TX	ro	Position Control Function
60FB _{hex}	08 _{hex}	P0362	TX / RX	rw	Position Control Function
60FB _{hex}	09 _{hex}	P0392	TX	ro	Position Control Function
60FB _{hex}	0A _{hex}	P0391	TX	ro	Position Control Function
60FB _{hex}	0B _{hex}	P0365	TX	ro	Position Control Function
60FB _{hex}	0C _{hex}	P0460	TX	ro	Position Control Function
60FB _{hex}	0D _{hex}	P1191	TX / RX	rw	Position Control Function
60FB _{hex}	0E _{hex}	P1190	TX	rw	Position Control Function
60FB _{hex}	0F _{hex}	P1200	TX	rw	Position Control Function
60FB _{hex}	10 _{hex}	P1208	TX	rw	Position Control Function
60FB _{hex}	11 _{hex}	P0464	TX	ro	Position Control Function
60FB _{hex}	12 _{hex}	P0605	TX / RX	rw	Position Control Function
60FB _{hex}	13 _{hex}	P1198	TX / RX	rw	Position Control Function
60FB _{hex}	14 _{hex}	P1199	TX / RX	rw	Position Control Function
60FB _{hex}	15 _{hex}	P0601	TX / RX	rw	Position Control Function
60FB _{hex}	16 _{hex}	P0608	TX / RX	rw	Position Control Function
60FB _{hex}	17 _{hex}	P0370	TX / RX	rw	Position Control Function
60FB _{hex}	18 _{hex}	P1209	TX / RX	rw	Position Control Function
60FB _{hex}	19 _{hex}	P1204	TX / RX	rw	Position Control Function
60FB _{hex}	1A _{hex}	P0353	TX	ro	Position Control Function

CANopen-Objekt-Nummer		Parameternummer	PDO-Mapping	Access - Typ	Betriebsart nach DSP 402
Index	Subindex				
60FB _{hex}	1B _{hex}	P0262 P0263	TX	ro	Position Control Function
60FD _{hex}	00 _{hex}	P0410	TX	ro	Common entries
60FF _{hex}	00 _{hex}	P1171	TX / RX	rw	Profile Velocity Mode
6510 _{hex}	01 _{hex}	P0001	TX	ro	Info
6510 _{hex}	02 _{hex}	P0002	TX	ro	Info
6510 _{hex}	03 _{hex}	P0003	TX	ro	Info
6510 _{hex}	04 _{hex}	P0004	TX	ro	Info
6510 _{hex}	05 _{hex}	P0005	TX	ro	Info
6510 _{hex}	06 _{hex}	P0009	TX	ro	Info
6510 _{hex}	07 _{hex}	P0555	TX	ro	Info
6510 _{hex}	08 _{hex}	P0556	TX	ro	Info



ANHANG C - UMSETZUNGSTABELLEN

Dieses Kapitel beinhaltet die Tabellen, welche die Umsetzung der CANopen-Kommunikationsobjekte in b maXX[®]-Regler-Kommunikationsparameter und umgekehrt spezifizieren. Die Umsetzung erfolgt unter Angabe der Wertebereiche ($x = x_{\min} \dots x_{\max}$) und der Abbildungsfunktion $x = f(x)$ (im einfachsten Fall wird der Wert nur durchgereicht: $y = x$).

Die Tabellen enthalten folgende Einträge:

CANopen-Objekt:	Bezeichnung des CANopen-Objektes aus DS402
Index ▶ P-Nr.:	Abbildung der CANopen-Objektindizes auf b maXX [®] -Regler-Parameter
Regler-Parameter:	Bezeichnung des Regler-Parameters
P-Nr. ▶ Index:	Umsetzung der b maXX [®] -Regler-Parameter auf CANopen-Objektindizes



CANopen-Objekt	Index Wertebereich	P-Nr. Normierung	Regler-Parameter	P-Nr. Wertebereich	Index Rück-Normierung	Kommentar
Controlword	6040 _{hex}	▶ P0300	Steuerwort	P0300	▶ 6040 _{hex}	Bit 6 im Steuerwort wird nun unterstützt; Bit 6 = 0 : Positionier- Modus „absolut“ Bit 6 = 1 : Positionier- Modus „relativ, negativ positiv“ Es wird über CoE und dem Steuerwort keine anderer Positionsmodus unterstützt.
	x = 0 .. FFFF _{hex}	▶ y = x		x = 0 .. FFFF _{hex}	▶ y = x	
Switch On	Bit 0	▶ unverändert	Einschalten	Bit 0	▶ unverändert	
Disable Voltage	Bit 1	▶ unverändert	Spannung sperren	Bit 1	▶ unverändert	
Quick Stop	Bit 2	▶ unverändert	Schnellhalt	Bit 2	▶ unverändert	
Enable Op.	Bit 3	▶ unverändert	Betrieb freigeben	Bit 3	▶ unverändert	
Operation mode specific	Bit 4	▶ unverändert	betriebsartabhängig	Bit 4	▶ unverändert	
Operation mode specific	Bit 5	▶ unverändert	betriebsartabhängig	Bit 5	▶ unverändert	
Operation mode specific	Bit 6	▶ unverändert	betriebsartabhängig	Bit 6	▶ unverändert	
Reset Fault	Bit 7	▶ unverändert	Reset Störung	Bit 7	▶ unverändert	
Operation mode specific	Bit 8	▶ unverändert	betriebsartabhängig	Bit 8	▶ unverändert	
reserved	Bit 9	▶ unverändert	Reserve (immer 0)	Bit 9	▶ unverändert	
reserved	Bit 10	▶ unverändert	Reserve (immer 0)	Bit 10	▶ unverändert	
Manufacturer specific	Bit 11	▶ unverändert	betriebsartabhängig	Bit 11	▶ unverändert	
Manufacturer specific	Bit 12	▶ unverändert	betriebsartabhängig	Bit 12	▶ unverändert	
Manufacturer specific	Bit 13	▶ unverändert	betriebsartabhängig	Bit 13	▶ unverändert	
Manufacturer specific	Bit 14	▶ unverändert	betriebsartabhängig	Bit 14	▶ unverändert	
Manufacturer specific	Bit 15	▶ unverändert	Schreibschutz	Bit 15	▶ unverändert	
Statusword	6041 _{hex} /ro		Statuswort	P0301	▶ 6041 _{hex}	
	x = 0 .. FFFF _{hex}			x = 0 .. FFFF _{hex}	▶ y = x	
Ready To Switch On			Einschaltbereit	Bit 0	▶ unverändert	
Switched On			Eingeschaltet	Bit 1	▶ unverändert	
Operation Enabled			Betrieb freigegeben	Bit 2	▶ unverändert	
Fault			Störung	Bit 3	▶ unverändert	
Voltage Disabled			Spannung gesperrt	Bit 4	▶ unverändert	
Quick Stop			Schnell-Halt	Bit 5	▶ unverändert	
Switched On Enabled			Einschaltsperr	Bit 6	▶ unverändert	
Warning			Warning	Bit 7	▶ unverändert	

CANopen-Objekt	Index Wertebereich	P-Nr. Normierung	Regler-Parameter	P-Nr. Wertebereich	Index Rück-Normierung	Kommentar
Man. specific			betriebsartabhängig	Bit 8	▶ unverändert	
Remote			Remote	Bit 9	▶ unverändert	
Targeted Reached			Sollwert erreicht	Bit 10	▶ unverändert	
Internal Limit Active			betriebsartabhängig	Bit 11	▶ unverändert	In WinBASS über Antriebsmanager einstellbar
Operation mode specific			betriebsartabhängig	Bit 12	▶ unverändert	
Operation mode specific			betriebsartabhängig	Bit 13	▶ unverändert	
Manufacturer .specific			konf. Statusbits	Bit 14	▶ unverändert	
Manufacturer specific			konf. Statusbits	Bit 15	▶ unverändert	
vl_target_velocity	6042 _{hex} x = -32768 .. 32767	▶ P1171 y = x * 4000 _{hex} / MotorMax-Speed	RFG1Input	P1171 x = -32768 .. 32767	▶ 6042 _{hex} y = x * MotorMaxSpeed / 4000 _{hex}	Die benutzerdefinierte Einheit (speed units) wird im b maXX [®] -Regler als U/min interpretiert. Die Skalierung der Getriebeübersetzung ist im FBO 604C _{hex} abgelegt.
vl_velocity_demand	6043 _{hex} /ro		RFG Output	P0351 x = 8000 _{hex} .. 7FFF _{hex}	▶ 6043 _{hex} y = x * MotorMaxSpeed / 4000 _{hex}	Die benutzerdefinierte Einheit (speed units) wird im b maXX [®] -Regler als U/min interpretiert. Die Skalierung der Getriebeübersetzung ist im FBO 604C _{hex} abgelegt.
vl_control_effort	6044 _{hex} /ro		SpeedActValue	P0353 x = 8000 _{hex} .. 7FFF _{hex}	▶ 6044 _{hex} y = x * MotorMaxSpeed / 4000 _{hex}	Die benutzerdefinierte Einheit (speed units) wird im b maXX [®] -Regler als U/min interpretiert. Die Skalierung der Getriebeübersetzung ist im FBO 604C _{hex} abgelegt.
vl_control_effort	6045 _{hex} /ro		SpeedActValue	P0352 x = 8000 _{hex} .. 7FFF _{hex}	▶ 6045 _{hex} y = x * MotorMaxSpeed / 4000 _{hex}	Die benutzerdefinierte Einheit (speed units) wird im b maXX [®] -Regler als U/min interpretiert. Die Skalierung der Getriebeübersetzung ist im FBO 604C _{hex} abgelegt.
vl_velocity_min_max_amount	6046 _{hex}					
vl_velocity_min_amount	Sub. 01 _{hex}	„kein“	SpeedSet_Ulim	„kein“ x = 0	▶ Sub. 01 _{hex} ▶ y = x	Sub. 1 ist immer Null, die min. Grenze ist auf Null festgelegt
vl_velocity_max_amount	Sub. 02 _{hex} x = 0 .. FFFFFFFF _{hex}	▶ P1042 / P1041 y = x * 40000000 _{hex} / MotorMax-Speed	SpeedSet_Llim P1041: x = 0 .. 40000000 _{hex} P1042: x = C0000000 _{hex} .. 0	P1042 / P1041	▶ Sub. 02 _{hex} ▶ y = x * MotorMaxSpeed / 40000000 _{hex}	Die max. Grenze wirkt sich im b maXX [®] auf beide Dreh-Richtungen symmetrisch aus. Die benutzerdefinierte Einheit (speed units) wird im b maXX [®] -Regler als U/min interpretiert



CANopen-Objekt	Index Wertebereich	P-Nr. Normierung	Regler-Parameter	P-Nr. Wertebereich	Index Rück-Normierung	Kommentar
vl_velocity_acceleration	6048 _{hex}	$a = \left(\frac{dv}{dt}\right)$; Da im Regler für die Beschleunigung kein Parameter vorhanden ist, wird durch variieren der Hochlaufzeit P1172 die gewünschte Beschleunigung erreicht. Sie wird auf die maximale Drehzahl des Reglers normiert. Die Berechnung für die gewünschte Beschleunigung erfolgt erst nach Eingabe von dv in SIX1 und dann dt in SIX2. Erst dann wird die korrekt berechnete Zeit auf die Hochlaufzeit geschrieben. Die Rekonstruktion der eingestellten Beschleunigung $\left(\frac{dv}{dt}\right)$ ist nach einem Neubooten nicht mehr möglich.				
vl_delta_speed	Sub. 01 _{hex} x = 0 .. FFFFFFFF _{hex}	y = Δv		x = 8000 .. 7FFF _{hex}	Sub. 01 _{hex} y = Δv	Die Skalierung der Getriebeübersetzung ist im FBO 604C _{hex} abgelegt.
vl_delta_time	Sub. 02 _{hex} x = 0 .. FFFFFFFF _{hex}	P1172 y = Δt*Motor-MaxSpeed / Δv*100	RFG1RampUpTime	P1172 x = 0 .. 65000	Sub. 02 _{hex} y = Δt	delta_time wird in Sekunden angegeben; entspricht der Hochlaufgeber-Hochlaufzeit
vl_velocity_deceleration	6049 _{hex}	$a = \left(\frac{dv}{dt}\right)$; Da im Regler für die Verzögerung kein Parameter vorhanden ist, wird durch variieren der Rücklaufzeit P1173 die gewünschte Verzögerung erreicht. Sie wird auf die maximale Drehzahl des Reglers normiert. Die Berechnung für die gewünschte Verzögerung erfolgt erst nach Eingabe von dv in SIX1 und dann dt in SIX2. Erst dann wird die korrekt berechnete Zeit auf die Rücklaufzeit geschrieben. Die Rekonstruktion der eingestellten Verzögerung $\left(\frac{dv}{dt}\right)$ ist nach einem Neubooten nicht mehr möglich.				
vl_delta_speed	Sub. 01 _{hex} x = 0 .. FFFFFFFF _{hex}	y = Δv		x = 8000 .. 7FFF _{hex}	Sub. 01 _{hex} y = Δv	Die Skalierung der Getriebeübersetzung ist im FBO 604C _{hex} abgelegt.
vl_delta_time	Sub. 02 _{hex} x = 0 .. FFFFFFFF _{hex}	P1173 y = Δt*Motor-MaxSpeed / - Δv*100	RFG1RampDownTime	P1173 x = 0 .. 65000	Sub. 02 _{hex} y = Δt	delta_time wird in Sekunden angegeben; entspricht der Hochlaufgeber-Rücklaufzeit
vl_dimension_factor	604C _{hex}				604C _{hex}	Die Umrechnung im Regler sieht z. B. folgendermaßen aus: Sollwert Drehzahl Motor im b maXX®: Für vl_dimension_factor_numerator = 10 und vl_dimension_factor_denominator = 5 Sollwert Drehzahl Motor = FBO [U/min]*vl_dimension_factor = 100*10 / 5 [U/min] = 200 [U/min]
vl_dimension_factor_numerator	Sub. 01 _{hex} X=-2 ³¹ .. 2 ³¹ -1	y=x		x=-33000 .. 33000	Sub. 01 _{hex} y=x	
vl_dimension_factor_denominator	Sub. 02 _{hex} X=-2 ³¹ .. 2 ³¹ -1	y=x		x=-33000 .. 33000	Sub. 02 _{hex} y=x	
vl_pole_number	604D _{hex} x = 0 .. 255	P0065 y = x / 2	PolPaarzahl	P0065 x = 1..120	604D _{hex} y = x*2	



CANopen-Objekt	Index Wertebereich	P-Nr. Normierung	Regler-Parameter	P-Nr. Wertebereich	Index Rück-Normierung	Kommentar
vl_ramp_function_time	604F _{hex}	▶ P1172	RFG1RampUpTime	P1172	▶ 604F _{hex}	Hochlaufgeber Hochlaufzeit (1 = 1/1000 s ⇒ 1s = 1000). Die Auflösung beträgt 10 ms
	x = 0 .. FFFFFFFF _{hex}	▶ y = x		x = 0 .. 65000	▶ y = x	
vl_slow_down_time	6050 _{hex}	▶ P1173	RFG1RampDownTime	P1173	▶ 6050 _{hex}	Hochlaufgeber Hochlaufzeit (1 = 1/1000 s, 1s = 1000). Die Auflösung beträgt 10 ms
	x = 0 .. FFFFFFFF _{hex}	▶ y = x		x = 0..65000	▶ y = x	
vl_quick_stop_time	6051 _{hex}	▶ P1174	RFG1StopTime	P1174	▶ 6051 _{hex}	Hochlaufgeber Hochlaufzeit (1 = 1/1000 s, 1s = 1000). Die Auflösung beträgt 10 ms
	x = 0 .. FFFFFFFF _{hex}	▶ y = x		x = 0..65000	▶ y = x	
quick_stop_option_code	605A _{hex}	▶ P1004	QuickstopCode (Schnellhalt)	P1004	▶ 605A _{hex}	
Umrechnungsformalismus	x = -32768 .. 32767	▶ y = x		x = 0 .. 3	▶ y = x	
Manufacturer specific	x = -32768 .. -1	▶ y = x	unbenutzt	x = -32768 .. -1		
Disable Drive	x = 0	▶ y = x	Antrieb gesperrt	x = 0	▶ y = x	
Slow Down On Slow Down Ramp	x = 1	▶ y = x	Rücklauf auf Rücklauftrampe	x = 1	▶ y = x	
Slow Down On Quick Stop Ramp	x = 2	▶ y = x	Rücklauf auf Schnellhaltrampe	x = 2	▶ y = x	
Slow Down On Current Ramp	x = 3	▶ y = x	Rücklauf an Stromgrenze	x = 3	▶ y = x	
Slow Down On Voltage Limit	x = 4	▶ y = x	Rücklauf an Spannungsgrenze	x = 4	y = x	
Slow Down On Slow Down Ramp and stay in Quick-Stop	x = 5	▶ y = x	Rücklauf auf Rücklauftrampe und im Quickstop bleiben	x = 5	y = x	
Slow Down On Quick Stop Ramp and stay in Quick-Stop	x = 6	▶ y = x	Rücklauf auf Schnellhaltrampe und im Quickstop bleiben	x = 6	y = x	
Slow Down On Current and stay in Quick-Stop	x = 7	▶ y = x	Rücklauf an Stromgrenze und im Quickstop bleiben	x = 7	y = x	
Slow Down On Voltage Limit and stay in Quick-Stop	x = 8	▶ y = x	Rücklauf an Spannungsgrenze und im Quickstop bleiben	x = 8	y = x	
reserved	x = 9 .. 32767		unbenutzt		y = 9 .. 32767	
shutdown_option_code	605B _{hex}	▶ P1005	ShutDownCode (Stillsetzen)	P1005	▶ 605B _{hex}	
Manufacturer specific	x = -32768 .. -3	▶ y = x	unbenutzt	x = -32768..-3		
Manufacturer specific	x = -2	▶ y = 3	Rücklauf an Stromgrenze	x = 3	▶ y = -2	
Manufacturer specific	x = -1	▶ y = 2	Rücklauf an Schnellhaltrampe	x = 2	▶ y = -1	
Disable Drive	x = 0	▶ y = x	Antrieb gesperrt	x = 0	▶ y = x	



CANopen-Objekt	Index Wertebereich	P-Nr. Normierung	Regler-Parameter	P-Nr. Wertebereich	Index Rück-Normierung	Kommentar
Slow Down On Slow Down Ramp	x = 1	y = x	Rücklauf auf Rücklauframpe	x = 1	y = x	Des ausgewählten HLG einstellbar über P1174 HLG Haltezeit oder in 6051 _{hex}
Reserved	x = 2 .. 32767		unbenutzt		y = .. 32767	
disable_operation_option_code	605C _{hex}	P1006	DisableOpCode (Sperrern)	P1006	605C _{hex}	
Manufacturer specific	x = -32768 .. -3	y = x	unbenutzt	x = -32768..-3		
Manufacturer specific	x = -2	y = 3	unbenutzt		y = -2	
Manufacturer specific	x = -1	y = 2	unbenutzt		y = -1	
Disable Drive	x = 0	y = x	Antrieb gesperrt		y = 0	
Slow Down ...	x = 1	y = x	Rücklauf auf Rücklauframpe	x = 1	y = x	
reserved	x = 2		Rücklauf auf Schnellhalt-rampe	x = 2	y = -1	
reserved	x = 3		Rücklauf an Stromgrenze	x = 3	y = -2	
reserved	x = 4 .. 32767		unbenutzt		y = 4 .. 32767	
stop_option_code	605D _{hex}	P1003	StopOptionCode (Halt)	P1003	605D _{hex}	
Umrechnungsformalismus	x = -32768 .. 32767	y = x		x = 0 .. 3	y = x	
Manufacturer specific	x = -32768 .. -1	y = x	unbenutzt		y = -32768 .. -1	
Disable Drive	x = 0	y = x	Antrieb gesperrt	x = 0	y = x	
Slow Down On Slow Down Ramp	x = 1	y = x	Rücklauf auf Rücklauframpe	x = 1	y = x	Des ausgewählten HLG einstellbar über P1174 HLG Haltezeit oder in 6051 _{hex}
Slow Down On Quick Stop Ramp	x = 2	y = x	Rücklauf auf Schnellhalt-rampe	x = 2	y = x	
Slow Down On Current Ramp	x = 3	y = x	Rücklauf an Stromgrenze	x = 3	y = x	
Slow Down On Voltage Limit	x = 4	y = x	unbenutzt		y = 4	
reserved	x = 5 .. 32767		unbenutzt		y = 5 .. 32767	

CANopen-Objekt	Index Wertebereich	P-Nr. Normierung	Regler-Parameter	P-Nr. Wertebereich	Index Rück-Normierung	Kommentar
fault_reaction_option_code	605E _{hex}	▶ P1007	ErrorReactionCode	P1007	▶ 605E _{hex}	Nur für statische und dynamische Schleppfehler und für die Reaktionen für das FBO 6007 _{hex} „Modus 1 Malfunction“ einstellbar
Umrechnungsformalismus	x = -32768 .. 32767	▶ y = x		x = 0 .. 3	▶ y = x	
Manufacturer specific	x = -32768 .. -1	▶ y = x	unbenutzt	x = -32768 .. -1		
Disable Drive, motor is free to rotate	x = 0	▶ y = x	Antrieb gesperrt	x = 0	▶ y = x	
Slow down on slow down ramp	x = 1	▶ y = x	Rücklauf auf Rücklauframpe	x = 1	▶ y = x	
Slow down on quick stop ramp	x = 2	▶ y = x	Rücklauf auf Schnellhaltrampe	x = 2	▶ y = x	
Slow down on current ramp	x = 3	▶ y = x	Rücklauf an Stromgrenze	x = 3	▶ y = x	
Slow down on voltage limit	x = 4	▶ y = x	Rücklauf an Spannungsgrenze		y = 4	
reserved	x = 5 .. 32767		unbenutzt		y = 5 .. 32767	
modes_of_operation	6060 _{hex} /wo	▶ P1000				
Umrechnungsformalismus	x = -128 .. 127	▶ y = x				
Manufacturer specific	x = -128 .. -7	▶ y = x				
Manufacturer specific	x = -6	▶ y = 5				
Manufacturer specific	x = -5	▶ y = x				
Manufacturer specific	x = -4	▶ y = x				
Manufacturer specific	x = -3	▶ y = x				
Manufacturer specific	x = -2	▶ y = x				
Manufacturer specific	x = -1	▶ y = x				
reserved	x = 0					
Profile Position Mode	x = 1	▶ y = x				
Velocity Mode	x = 2	▶ y = x				
Profile Velocity Mode	x = 3	▶ y = -3				
Torque Profile Mode	x = 4	▶ y = x				
reserved	x = 5					
Homing Mode	x = 6	▶ y = x				
Interpolated Position Mode	x = 7	▶ y = x				
reserved	x = 8 .. 127					
modes_of_operation_display	6061 _{hex} /ro		OperationModeAct (Ist-Betriebsart)	P0304	▶ 6061 _{hex}	Die CANopen Standardbezeichnung, siehe 6060 _{hex}



CANopen-Objekt	Index Wertebereich	P-Nr. Normierung	Regler-Parameter	P-Nr. Wertebereich	Index Rück-Normierung	Kommentar
Umrechnungsformalismus				x = -128 .. 127	y = x	Wurde über FBO 6060 _{hex} der Mode 3 gewählt, so wird hier der Wert 3 zurückgegeben, andernfalls der Wert -3. Nach dem Power-Down geht die Information, welcher Mode gewählt wurden verloren.
			Selbstoptimierung	x = 1 .. -7		
			unbenutzt	x = -6		
			Gleichlauf el. Getriebe	x = -5	y = x	
			Lageregelung	x = -4	y = x	
			Drehzahlregelung	x = -3	y = 3, y = -3	
			Stromregelung	x = -2	y = x	
			Rastlage	x = -1	y = x	
			unbenutzt	x = 0		
			Lagezielvorgabe	x = 1	y = x	
			Geschwindigkeitsvorgabe 1	x = 2	y = x	
			unbenutzt	x = 3		
			unbenutzt	x = 4		
			Handfahrbetrieb	x = 5	y = -6	
			Referenzbetrieb	x = 6	y = x	
unbenutzt	x = 7					
unbenutzt	x = 8 .. 127					
position_demand_value	6062 _{hex} /ro		PPosSetValue (Lage-Istwert)	P0463 x = 80000000 .. 7FFFFFFF _{hex}	6062 _{hex} y = x-2 ³¹	USIGN32 wird auf der CoE Optionskarte mit einem Offset von 2 ³¹ versehen. USIGN32 -> INT32. (Offset von 2 ³¹ wird Subtrahiert. BE - Übersetzung hinzugefügt.
position_actual_value*	6063 _{hex} /ro		PPosActValue (Lage-Istwert)	P0462 x = 80000000 .. 7FFFFFFF _{hex}	6063 _{hex} y = x	BE - Übersetzung hinzugefügt.
position_actual_value	6064 _{hex} /ro		PPosActValue (Lage-Istwert)	P0462 x = 80000000 .. 7FFFFFFF _{hex}	6064 _{hex} y = x-2 ³¹	USIGN32 wird auf der CoE Optionskarte mit einem Offset von 2 ³¹ versehen. USIGN32 ⇒ INT32. (Offset von 2 ³¹ wird Subtrahiert. BE - Übersetzung hinzugefügt.
following_error_time_out	6066 _{hex} /ro		PosDevTime	P1056 x = 0 .. 65000	6066 _{hex} y = x	Die Einheit ist im CANopen-Objekt und im b maXX [®] -Regler-Parameter in ms.



CANopen-Objekt	Index Wertebereich	P-Nr. Normierung	Regler-Parameter	P-Nr. Wertebereich	Index Rück-Normierung	Kommentar
position_window	6067 _{hex}	▶ P1194	PPosWindow (Pos.-Fenster)	P1194	▶ 6067 _{hex}	
	x = 0 .. FFFFFFFF _{hex}	▶ y = x		x = 0 .. FFFFFFFF _{hex}	▶ y = x	
position_window_time	6068 _{hex}	▶ P1195	PPosWindow Time (Pos.-Fensterzeit)	P1195	▶ 6068 _{hex}	
	x = 0 .. 65535	▶ y = x		x = 1 .. FFFF _{hex}	▶ y = x	
velocity_sensor_actual_value	6069 _{hex}	▶ P0362	Enc1ActAngle	P0391	▶ 6069 _{hex}	
	x = -2 ¹⁵ .. 2 ¹⁵ -1	▶ y = x		x = 0 .. FFFFFFFF _{hex}	▶ y = x	
sensor_selection_code	606A _{hex} /ro		„kein“			Der b maXX [®] -Regler unterstützt nur Positions-Encoder, daher nur Anzeige.
velocity_actual_value_from_position_encoder				x = 0	▶ y = x	
velocity_actual_value_from_velocity_encoder			nicht unterstützt			
velocity_demand_value	606B _{hex} /ro		SetValueTotal	P0352	▶ 606B _{hex}	Die benutzerdefinierte Einheit (speed units) wird im Regler als U/min interpretiert.
				x = 8000 _{hex} .. 7FFF _{hex}	▶ y = x*Motor-MaxSpeed / 4000 _{hex}	
velocity_actual_value	606C _{hex} /ro		SpeedActValue	P0353	▶ 606C _{hex}	Die Skalierung der Getriebeübersetzung ist im FBO 604C _{hex} abgelegt.
				x = -2 ³¹ .. 2 ³¹ -1	▶ y = x*Motor-MaxSpeed / 40000000 _{hex}	
velocity_threshold	606F _{hex}	▶ P1073	Enc1Mon_Llim	P1073	▶ 606F _{hex}	Die Schwelle kann im b maXX [®] -Regler bis auf 25% der maximalen Drehzahl des Reglers heraufgesetzt werden. Die Eingabe erfolgt dann in U/min z.B. Max: 1000 U/min Eingabe für 25 % = 250 U/min
	x = -2 ³¹ .. 2 ³¹ -1	▶ y = x*4000 _{hex} / 10000		x = -0 .. 1000 _{hex}	▶ y = x*10000 / 4000 _{hex}	
max_torque	6072 _{hex}	▶ P0357	TrqSynDirect	P0357	▶ 6072 _{hex}	
	x = 0 ... FFFF _{hex}	▶ y = x*4000 _{hex} / 1000		x = 0000 .. FFFF _{hex}	▶ y = x*1000 / 4000 _{hex}	
target_position	607A _{hex}	▶ P0607 (P0600)	PPosTarget1	P0607 (P0600)	▶ 607A _{hex}	BE - Übersetzung hinzugefügt.
	x = 80000000 _{hex} .. 7FFFFFFF _{hex}	▶ y = x		x = 800000000 .. 7FFFFFFF _{hex}	▶ y = x	
home_offset	607C _{hex}	▶ P1200	PPosEncoderOffset	P1200	▶ 607C _{hex}	Abweichung der Homeposition vom Referenz- bzw. Endschalter BE - Übersetzung und ein Offset von 2 ³¹ hinzugefügt (Zahlenskalawandlung).
	x = -2 ³¹ .. 2 ³¹ -1	▶ y = x+2 ³¹		x = 0 .. 2 ³² -1	▶ y = x-2 ³¹	



CANopen-Objekt	Index Wertebereich	P-Nr. Normierung	Regler-Parameter	P-Nr. Wertebereich	Index Rück-Normierung	Kommentar
software_position_limit	607D _{hex}		SW-Endschalter		607D _{hex}	USIGN32 wird auf der CANopen Optionskarte mit einem Offset von 2 ³¹ versehen. USIGN32 ⇒ INT32. (Offset von 2 ³¹ wird Subtrahiert. BE - Übersetzung hinzugefügt.
	Sub. 01 _{hex}	▶ P1196	PPosSWLimitSwitch1	P1196	▶ Sub. 01 _{hex}	
	x = -2 ³¹ .. 2 ³¹ -1	▶ y = x		x = 0 .. FFFFFFFF _{hex}	▶ y = x	
	Sub. 02 _{hex}	▶ P1197	PPosSWLimitSwitch2	P1197	▶ Sub. 02 _{hex}	
	x = -2 ³¹ .. 2 ³¹ -1	▶ y = x		x = 0 .. FFFFFFFF _{hex}	▶ y = x	
max_profile_velocity	607F _{hex}	▶ P0057	MotorNomSpeed	P0057	▶ 607F _{hex}	Die benutzerdefinierte Einheit (speed units) wird im Regler als U/min interpretiert
	x = 0 .. 2 ³² -1	▶ y = x		x = 1 .. 24000	▶ y = x	
max_motor_speed	6080 _{hex}	▶ P1031	SpeedMax	P1031	▶ 6080	Die benutzerdefinierte Einheit (speed units) wird im Regler als U/min interpretiert
	x = 0 .. FFFF _{hex}	▶ y = x		x = 1 .. 24000	▶ y = x	
profile velocity	6081 _{hex}	▶ P0602	PPosSetSpeed1	P0602	▶ 6081 _{hex}	BE - Übersetzung hinzugefügt.
	x = 0 .. 2 ³² -1	▶ y = x		x = 1 .. 13200	▶ y = x	
profile acceleration	6083 _{hex}	▶ P0603	PPosAcceleraton1	P0603	▶ 6083 _{hex}	BE - Übersetzung hinzugefügt.
	x = 0 .. 2 ³² -1	▶ y = x		x = 25 .. 45000	▶ y = x	
profile deceleration	6084 _{hex}	▶ P0604	PPosDeceleraton1	P0604	▶ 6084 _{hex}	
	x = 0 .. 2 ³² -1	▶ y = x		x = 25 .. 45000	▶ y = x	
quick_stop_deceleration	6085 _{hex}	▶ P1213	PPosStopDeceleraton	P1213	▶ 6085 _{hex}	BE - Übersetzung hinzugefügt.
	x = 0 .. 2 ³² -1	▶ y = x		x = 25 .. 45000	▶ y = x	
motion profile type	6086 _{hex}	▶ P1190	PPosMode	P1190	▶ 6086 _{hex}	
	x = -2 ¹⁶ ... 2 ¹⁶ -1	▶		x = 0.. FFFF _{hex}		
Manufacturer specific	x = -32768..1		Unbenutzt			
Linear ramp (trapezoidal profile)	x = 0	▶ Bit 3 und Bit 4	Trapez	Bit 3 und Bit 4	▶ 0	Geschwindigkeitsprofil: Bit 4 Bit 3: 0 0: Trapez 1 0: Sin ² 0 1: S-Kurve 1 1: Reserviert
Sin ² ramp	x = 1	▶ Bit 3 und Bit 4	Sin ²	Bit 3 und Bit 4	▶ 1	
Jerk-free ramp	x = 2	▶ Bit 3 und Bit 4	S-Kurve	Bit 3 und Bit 4	▶ 2	
Jerk-limited ramp	x = 3		Unbenutzt			
For future profile type	x = 4..32767		Unbenutzt			
feed_constant	6092 _{hex} /ro				▶ 6092 _{hex}	
feed	Sub. 01 _{hex}	▶ P3050	PosScalingUserUnit	P3050	▶ Sub. 01 _{hex}	
	x = 0 .. FFFFFFFF _{hex}	▶ y = x		x = 2 ²⁴ .. 1	▶ y = x	
shaft_revolutions	Sub. 02 _{hex}	▶ P3051	PosScalingRevolution	P3051	▶ Sub. 02 _{hex}	
	x = 0 .. FFFFFFFF _{hex}	▶ y = x		x = 1 .. 2 ²⁴ -1	▶ y = x	

CANopen-Objekt	Index Wertebereich	P-Nr. Normierung	Regler-Parameter	P-Nr. Wertebereich	Index Rück-Normierung	Kommentar
homing_method	6098 _{hex}	▶ P1205	PPosHomingMode (Ref. Fahrmodus)	P1205	▶ 6098 _{hex}	
Manufacturer specific	x = -128 .. -12		unbenutzt		y = -128 ..-13	
Manufacturer specific	x = -10	▶ y = -10	Anfahren mechanischer Anschlag mit Nullimpuls, Linksdrehung	x = -10	▶ y = -10	
Manufacturer specific	x = -9	▶ y = -9	Anfahren mechanischer Anschlag mit Nullimpuls, Rechtsdrehung	x = -9	▶ y = -9	
Manufacturer specific	x = -8	▶ y = -8	Anfahren mechanischer Anschlag, Linksdrehung	x = -8	▶ y = -8	
Manufacturer specific	x = -7	▶ y = -7	Anfahren mechanischer Anschlag, Rechtsdrehung	x = -7	▶ y = -7	
Manufacturer specific	x = -6	▶ y = -6	Anfahren des nächsten Gebernulwinkels	x = -6	▶ y = -6	
Manufacturer specific	x = -5	▶ y = -5	Anfahren des pos. Endschal- ters	x = -5	▶ y = -5	
Manufacturer specific	x = -4	▶ y = -4	Anfahren des neg. Endschal- ters	x = -4	▶ y = -4	
Manufacturer specific	x = -3	▶ y = -3	Referenzpunkt setzen	x = -3	▶ y = -3	
Manufacturer specific	x = -2	▶ y = -2	Anfahren des Gebernulwinkels bzw. des Nullimpulses mit Linksdrehung	x = -2	▶ y = -2	
Manufacturer specific	x = -1	▶ y = -1	Anfahren des Gebernulwinkels bzw. des Nullimpulses mit Rechtsdrehung	x = -1	▶ y = -1	
No homing operation	x = 0		unbenutzt		y = 0	
Homing on the neg. Limit Switch	x = 1	▶ y = 1	Anfahren des neg. Endschal- ters mit Gebernulwinkel- bzw. Nullimpulsreferenzierung	x = 1	▶ y = 1	
Homing on the pos.Limit Switch	x = 2	▶ y = 2	Anfahren des pos. Endschal- ters mit Gebernulwinkel- bzw. Nullimpulsreferenzierung	x = 2	▶ y = 2	
Homing on the positive Home Switch & Index Pulse	x = 3	▶ y = 3	Anfahren des pos. Nullpunk- tumschalers mit Gebernul- winkel- bzw. Nullimpulsreferenzierung	x = 3	▶ y = 3	
Homing on the positive Home Switch & Index Pulse	x = 4	▶ y = 4	Anfahren des pos. Nullpunk- tumschalers mit Gebernul- winkel- bzw. Nullimpulsreferenzierung	x = 4	▶ y = 4	

CANopen-Objekt	Index Wertebereich	P-Nr. Normierung	Regler-Parameter	P-Nr. Wertebereich	Index Rück-Normierung	Kommentar
Homing on the negative Home Switch & Index Pulse	x = 6	y = 6	Anfahren des neg. Nullpunktumschalters mit Gebernulswinkel- bzw. Nullimpulsreferenzierung	x = 6	y = 6	
Zero reference cam switch, left to pos. edge with Zero pulse; CW move	x = 7	y = 7	Nullpunktschalter, links von pos. Flanke mit Nullimpuls; Rechtsdrehung	x = 7	y = 7	
Zero reference cam switch, right fo pos. edge with Zero pulse; CW move	x = 8	y = 8	Nullpunktschalter, rechts von pos. Flanke mit Nullimpuls; Rechtsdrehung	x = 8	y = 8	
Zero reference cam switch, left to neg. edge with Zero pulse; CW move	x = 9	y = 9	Nullpunktschalter, links von neg. Flanke mit Nullimpuls; Rechtsdrehung	x = 9	y = 9	
Zero reference cam switch, right to neg. edge with Zero pulse; CW move	x = 10	y = 10	Nullpunktschalter, rechts von neg. Flanke mit Nullimpuls; Rechtsdrehung	x = 10	y = 10	
Zero reference cam switch, right to neg. edge with Zero pulse; CCW move	x = 11	y = 11	Nullpunktschalter, rechts von neg. Flanke mit Nullimpuls; Linksdrehung	x = 11	y = 11	
Zero reference cam switch, right fo pos. edge with Zero pulse; CCW move	x = 12	y = 12	Nullpunktschalter, rechts von pos. Flanke mit Nullimpuls; Linksdrehung	x = 12	y = 12	
Zero reference cam switch, left to neg. edge with Zero pulse; CCW move	x = 13	y = 13	Nullpunktschalter, links von neg. Flanke mit Nullimpuls; Linksdrehung	x = 13	y = 13	
Zero reference cam switch, right to neg. edge with Zero pulse; CCW move	x = 14	y = 14	Nullpunktschalter, rechts von neg. Flanke mit Nullimpuls; Linksdrehung	x = 14	y = 14	
CANopen Spec.	x = 15, 16		unbenutzt			
Negative Limit Switch	x = 17	y = 17	negativer Endschalters	x = 17	y = 17	
Positive Limit Switch	x = 18	y = 18	positiver Endschalters	x = 18	y = 18	
Positive Zero reference switch, CCW move	x = 19	y = 19	positiver Nullpunktumschalter; Linksdrehung	x = 19	y = 19	
Positive Zero reference switch, CW move	x = 20	y = 20	positiver Nullpunktumschalter; Rechtsdrehung	x = 20	y = 20	
Negative Zero reference switch, CW move	x = 21	y = 21	negativer Nullpunktumschalter; Rechtsdrehung	x = 21	y = 21	

CANopen-Objekt	Index Wertebereich	P-Nr. Normierung	Regler-Parameter	P-Nr. Wertebereich	Index Rück-Normierung	Kommentar
Negative Zero reference switch, CCW move	x = 22	y = 22	negativer Nullpunktschalter; Linksdrehung	x = 22	y = 22	
Zero reference cam switch, left to pos. edge; CW move	x = 23	y = 23	Nullpunktschalter, links von pos.; Rechtsdrehung	x = 23	y = 23	
Zero reference cam switch, right to pos. edge; CW move	x = 24	y = 24	Nullpunktschalter, rechts von pos. Flanke; Rechtsdrehung	x = 24	y = 24	
Zero reference cam switch, left to neg. edge; CW move	x = 25	y = 25	Nullpunktschalter, links neg. Flanke; Rechtsdrehung	x = 25	y = 25	
Zero reference cam switch, right to neg. edge; CW move	x = 26	y = 26	Nullpunktschalter, rechts von neg. Flanke; Rechtsdrehung	x = 26	y = 26	
Zero reference cam switch, right to neg. edge; CCW move	x = 27	y = 27	Nullpunktschalter, rechts von neg. Flanke; Linksdrehung	x = 27	y = 27	
Zero reference cam switch, left to neg. edge; CCW move	x = 28	y = 28	Nullpunktschalter, links von neg. Flanke; Linksdrehung	x = 28	y = 28	
Zero reference cam switch, right to pos. edge; CCW move	x = 29	y = 29	Nullpunktschalter, rechts von pos. Flanke; Linksdrehung	x = 29	y = 29	
Zero reference cam switch, left to pos. edge; CCW move	x = 30	y = 30	Nullpunktschalter, links von pos. Flanke; Linksdrehung	x = 30	y = 30	
CANopen-Spec.	31..32		unbenutzt	31..32		
Nearest Zero pulse; CCW move	x = 33	y = 33	nächster Nullimpuls; Linksdrehung	x = 33	y = 33	
Nearest Zero pulse; CW move	x = 34	y = 34	nächster Nullimpuls mit Rechtsdrehung	x = 34	y = 34	
Homing on the Current Position	x = 35	y = 35	Referenzpunkt setzen	x = 35	y = 35	
reserved	x = 36 .. 127		unbenutzt			
homing_speeds	6099 _{hex}		(Ref. Geschw.)		6099 _{hex}	
Speed_during_search_for_switch	Sub. 01 _{hex}	P1201	PPosHomingSpeed	P1201	Sub. 01 _{hex}	
	x = 0 .. 2 ³² -1	y = x		x = 1..13200	y = x	
Speed_during_search_for_zero	Sub. 02 _{hex}	P1202	PPosHomingFinalSpeed	P1202	Sub. 02 _{hex}	
	x = 0 .. 2 ³² -1	y = x		x = 1..50	y = x	



CANopen-Objekt	Index Wertebereich	P-Nr. Normierung	Regler-Parameter	P-Nr. Wertebereich	Index Rück-Normierung	Kommentar
homing_acceleration	609A _{hex}	▶ P1203	PPosHomingAcceler (Referenzbeschleunigung)	P1203	▶ 609A _{hex}	
	x = 0 .. 2 ³² -1	▶ y = x		x = 25 .. 45000	▶ y = x	
position_control_parameter_set	60FB _{hex}					Herstellerspezifisches CANopen-Objekt
	Sub. 01 _{hex} /ro		PosCtrlStatus	P0360	▶ Sub. 01 _{hex}	Default = 0
				x = 0 .. 2 ¹⁶ -1	▶ y = x	
	Sub. 02 _{hex}	▶ P1050	PosCtrlMode	P1050	▶ Sub. 02 _{hex}	Default = 0
	x = 0 .. 2 ¹⁶ -1	▶ y = x		x = 0 .. 2 ¹⁶ -1	▶ y = x	
	Sub. 03 _{hex}	▶ P1051	PosCtrl_Kv-Faktor	P1051	▶ Sub. 03 _{hex}	Default = 10,0
	x = 0 .. 32767	▶ y = 0 .. 32767		x = 0 .. 32767	▶ y = x	
	Sub. 04 _{hex}	▶ P0364	PosSetRev	P0364	▶ Sub. 04 _{hex}	Default = 0
	x = 0 .. 2 ¹⁶ -1	▶ y = x		x = 0 .. 2 ¹⁶ -1	▶ y = x	
	Sub. 05 _{hex}	▶ P0363	PosSetAngle	P0363	▶ Sub. 05 _{hex}	Default = 0
	x = 0 .. 2 ¹⁶ -1	▶ y = x		x = 0 .. 2 ¹⁶ -1	▶ y = x	
	Sub. 06 _{hex}	▶ P1053	SpeedFeedForFactor	P1053	▶ Sub. 06 _{hex}	Default = 4000 _{hex}
	x = 0 .. 2 ¹⁶ -1	▶ y = 0 .. 5000 _{hex}		x = 0 .. 5000 _{hex}	▶ y = 0 .. 2 ¹⁶ -1	
	Sub. 07 _{hex} /ro		PosCtrlDev	P0367	▶ Sub. 07 _{hex}	Default = 0
				x = -2 ³¹ .. 2 ³¹ -1	▶ y = x	
	Sub. 08 _{hex}		PosActValue	P0362	▶ Sub. 08 _{hex}	Default = 0
				x = 0 .. 2 ³² -1	▶ y = x	
	Sub. 09 _{hex} /ro		Enc1ActRev	P0392	▶ Sub. 09 _{hex}	Default = 0
				x = 0 .. 2 ³² -1	▶ y = x	
	Sub. 0A _{hex} /ro		Enc1ActAngle	P0391	▶ Sub. 0A _{hex}	
				x = 0 .. 2 ³² -1	▶ y = x	
	Sub. 0B _{hex} /ro		SpeedFeedFor	P0365	▶ Sub. 0B _{hex}	
				x = -2 ³¹ .. 2 ³¹ -1	▶ y = x	
	Sub. 0C _{hex} /ro		PPosStatus	P0460	▶ Sub. 0C _{hex}	
				x = 0 .. FFFF _{hex}	▶ y = x	
	Sub. 0D _{hex}	▶ P1191	PPosActRecordNumber	P1191	▶ Sub. 0D _{hex}	
	x = 0 .. 2 ¹⁶ -1	▶ y = x		x = 0 .. 2 ¹⁶ -1	▶ y = x	
	Sub. 0E _{hex}	▶ P1190	PPosMode	P1190	▶ Sub. 0E _{hex}	
	x = 0 .. 2 ¹⁶ -1	▶ y = x		x = 0 .. FFFF _{hex}	▶ y = x	



CANopen-Objekt	Index Wertebereich	P-Nr. Normierung	Regler-Parameter	P-Nr. Wertebereich	Index Rück-Normierung	Kommentar
	Sub. 0F _{hex} x = 0 .. 2 ³² -1	▶ P1200 ▶ y = x	PPosHomePosition	P1200 x = 0 .. FFFFFFFF _{hex}	▶ Sub. 0F _{hex} ▶ y = x	
	Sub. 10 _{hex} x = 0 .. 2 ¹⁶ -1	▶ P1208 ▶ y = x	PPosSwitchMode	P1208 x = 0 .. FFFF _{hex}	▶ Sub. 10 _{hex} ▶ y = x	
	Sub. 11 _{hex} /ro		PPosSpeedSetValue	P0464 x = -32768 .. 32767	▶ Sub. 11 _{hex} ▶ y = x	
	Sub. 12 _{hex} x = 0 .. 2 ¹⁶ -1	▶ P0605 ▶ y = x	PPosBend0	P0605 x = 0 .. 8191	▶ Sub. 12 _{hex} ▶ y = x	
	Sub. 13 _{hex} x = 0 .. 2 ³² -1	▶ P1198 ▶ y = x	PPosClipEnvironment1	P1198 x = 0.. FFFFFFFF _{hex}	▶ Sub. 13 _{hex} ▶ y = x	
	Sub. 14 _{hex} x = 0 .. 2 ³² -1	▶ P1199 ▶ y = x	PPosClipEnvironment2	P1199 x = 0 .. FFFFFFFF _{hex}	▶ Sub. 14 _{hex} ▶ y = x	
	Sub. 15 _{hex} x = -2 ¹⁵ .. 2 ¹⁵ -1	▶ P0601 ▶ y = x	PPosTargetInput0	P0601 x = -2 ¹⁵ .. 2 ¹⁵ -1	▶ Sub. 15 _{hex} ▶ y = x	
	Sub. 16 _{hex} x = -2 ¹⁵ .. 2 ¹⁵ -1	▶ P0608 ▶ y = x	PposTargetInput1	P0608 -2 ¹⁵ .. 2 ¹⁵ -1	▶ Sub. 16 _{hex} ▶ y = x	
	Sub. 17 _{hex} x = 0 .. 2 ³² -1	▶ P0370 ▶ y = x	PosIpSetAngle	P0370 x = 0 .. 2 ³² -1	▶ Sub. 17 _{hex} ▶ y = x	
	Sub. 18 _{hex} x = 0 .. 2 ¹⁶ -1	▶ P1209 ▶ y = x	PPosEncoderOffset	P1209 x = 0 .. 2 ¹⁶ -1	▶ Sub. 18 _{hex} ▶ y = x	
	Sub. 19 _{hex} x = 0 .. 2 ¹⁶ -1	▶ P1209 ▶ y = x	PPosHomingDeceler	P1204 x = 0 .. 2 ¹⁶ -1	▶ Sub. 19 _{hex} ▶ y = x	
	Sub. 1A _{hex} / ro		SpeedActValue	P0353 x = -2 ³² .. 2 ³² -1	▶ Sub. 1A _{hex} ▶ y = x	Der Drehzahlwert (P0353) wird von einem 32 Bit Wert auf 16384 umnormiert. 100 % der maximalen Drehzahl (in P1031) entsprechen 16384 Einheiten. Es wird der Betrag ausgegeben.

CANopen-Objekt	Index Wertebereich	P-Nr. Normierung	Regler-Parameter	P-Nr. Wertebereich	Index Rück-Normierung	Kommentar
	Sub. 1B _{hex} / ro		AmpWarning/MotorWarning	P0262, P0263 x = 0 .. 2 ¹⁶ -1	Sub. 1B _{hex}	Bit 0 P0263 Bit 1 Motortemperatur hat Schwelle 2 überschritten Bit 1 P0263 Bit 1 Motortemperatur hat Schwelle 2 überschritten Bit 2 P0262 Bit 1 Leistungsteil Temperatur > 80°C Bit 3 nicht belegt Bit 4 P0263 Bit 0 Motortemperatur hat Schwelle 1 überschritten Bit 5 P0263 Bit 0 Motortemperatur hat Schwelle 1 überschritten
digital_inputs	60FD _{hex} / ro x = 0 .. 2 ¹⁶ -1	▶ P1208	DignOutStatus	P1208 x = 0 .. FFFF _{hex}	▶ 60FD _{hex} y = 0 .. 2 ³² -1	
Negative limit switch			Zustand Endschalter pos.	Bit 0	▶ Bit 1	
Positive limit switch			Zustand Endschalter neg.	Bit 1	▶ Bit 0	
Home switch			Zustand Nullpunktumschalter	Bit 2	▶ Bit 2	
Interlock			reserviert	Bit 4		
reserved			reserviert	Bit 3..15		
Man. specific			unbenutzt		Bit 16..31	
target_velocity	60FF _{hex} x = -2 ³¹ .. 2 ³¹ -1	▶ P1171 y = x	RFG1Input	P1171 x = 8000 _{hex} .. 7FFF _{hex}	▶ 60FF _{hex} y = x	Die benutzerdefinierte Einheit (velocity units) wird im b maxX [®] -Regler als U/min interpretiert. Nur bei Änderungen in Optionsmodul G/H-Konfiguration 1 Bit 2 = 1: Angabe der akt. Drehzahl in 1/10 U/min. z.B.: 200,0 Umdrehungen ⇒ Eingabe 2000.
drive_data	6510 _{hex}				6510 _{hex}	
Manufacturer specific	Sub. 01 _{hex} / ro			P0001	▶ Sub. 01 _{hex} y = x	
Manufacturer specific	Sub. 02 _{hex} / ro		ControllerType	P0002	▶ Sub. 02 _{hex} y = x	
Manufacturer specific	Sub. 03 _{hex} / ro		SoftwareType	P0003	▶ Sub. 03 _{hex} y = x	
Manufacturer specific	Sub. 04 _{hex} / ro		SoftwareID	P0004	▶ Sub. 04 _{hex} y = x	
Manufacturer specific			SoftwareVersion		▶ y = x	

CANopen-Objekt	Index Wertebereich	P-Nr. Normierung	Regler-Parameter	P-Nr. Wertebereich	Index Rück-Normierung	Kommentar
Manufacturer specific	Sub. 05 _{hex} / ro			P0005	Sub. 05 _{hex}	
			ParamTableVersion		y = x	
Manufacturer specific	Sub. 06 _{hex} / ro			P0009	Sub. 06 _{hex}	
			AmpSW_Version		y = x	
Manufacturer specific	Sub. 07 _{hex} / ro			P0555	Sub. 07 _{hex}	
			FbgaVersion		y = x	
Manufacturer specific	Sub. 08 _{hex} / ro			P0556	Sub. 08	
			BootloaderVersion		y = x	





ANHANG D - TECHNISCHE DATEN

In diesem Anhang finden Sie eine Übersicht der Technischen Daten der CoE-Optionskarte.

D.1 CoE-Optionskarte: Technische Merkmale

CPU	SH3
FPGA	XC35400 der SpartanII Reihe (Fa. XILINX)
Speicher	512 kByte DPRAM, 8 MByte SDRAM, 8 MByte Flash-Eprom
Baudrate	100 Mbit
Betriebsspannung	+5 V intern
Steckverbinder	2 RJ45 Buchsen, 8-polig

D.2 CANopen Optionskarte: Datenkanäle zum b maXX[®]-Regler

Für die Datenübertragung vom b maXX[®]-Regler zum Optionsmodul CoE-Slave stehen drei Kanäle zur Verfügung:

- zwei Prozessdatenkanäle (1 PDO je Kommunikationsrichtung)
- ein Bedarfsdatenkanal (Server-SDO)

Mit PDOs können im zyklischen Datenaustausch Objekte übertragen werden. Für den PDO-Transfer sind nicht alle Objekte verfügbar.

Mit dem SDO-Transfer kann über das Objektverzeichnis auf alle b maXX[®] 4400-Parameter zugegriffen werden (Ausnahme String-Parameter).



Abbildungsverzeichnis

Flexible Topologie: Linie, Baum oder Stern [1]	10
EtherCAT: Standard -IEEE 802.3-Frames [1]	11
Device Profile bei EtherCAT[1]	12
EtherCAT - Frame [1].....	12
EtherCAT Telegramm [1]	13
EtherCAT Kommunikationsübergänge [1].....	15
ProDrive Seite Optionsmodule G, H-Konfiguration	18
Profilstruktur von CANopen.....	19
Aufbau der Mailbox	40
Mailbox-Header	40
CoE-Header	40
Mapping	45
Beispiel-Mapping mit einem b maXX®.....	49





Stichwortverzeichnis

A		EtherCAT	9
Abkürzungen	67	EtherCAT State Machine	15
Abort Code	43	EtherCAT-Adresse	36
Abort Codes	43	EtherCAT-Header	12
Abort-Telegramm	43	EtherCAT-Telegramm	13
Access - Typ	70	Ethernet Header	13
Administrative Nachrichten	20	Ethernet Technology Group	10
Adressraum	44	Ethernet-Bus	5
AL Management	15	Event Mechanismus	49
Applikationsparameter	29	Expedited Transfer	42
B		F	
BACI	26, 52	Factor Group	24, 71
Baumtopologie	10	Fehlerauslösung	29
BE	30	fehlerhafte Zugriffe	43
Bedarfsdaten	25	Fehlermeldungen	54
Benutzer Einheiten	30	Fehlerreaktionen	43
Betriebsart	70	Fehlertelegramm	53
Betriebsarten	22	Feldbus Standard	10
Broadcast-Telegramm	17	Feldbus-Objekte	22
C		Fieldbus Memory Management Unit	44
CAL-Spezifikation	19	FMMU	6, 44
CANopen	9	FPGA	48
CANopen Fehlercode	54, 61	Frame Check Sequence	14
CANopen- Objekt-Nummer	70	Frame Header	13
CANopen over EtherCAT	5	Frame-Struktur	12
CANopen-Kommunikationsobjekte	75	G	
CoE	5	Geräteprofil	19
CoE-Header	40	Geräteprofile	12
CoE-Knoten	5	Geschwindigkeitsprofil	29
CoE-Slave	49	Getriebefaktor	31
Command Specifier	41	Grundlagen	10, 19
Common entries	73	H	
Control Register	15	Herstellerspezifische Objekte	69
D		Homing Mode	22, 71
Datenbereich	12	I	
Datenkanäle	93	Identity Object	36
Datentypen	20	Index	70
DC	47	Info	73
Device Control	22, 70	Init	15
Device Type	35	Installation	5
Distributed Clocks	47	Istwerte	46
Dummy-Mappin	46	K	
E		Kommunikation	25
Echtzeitanforderungen	44	Kommunikationsprofil	20
EDS-Datei	6	Kommunikationsrichtung	44
Einstellungen	27	Kommunikationssteuerung	35
EMCY	53	Kommunikationszeiten	26
EMCY-Code	17	Kommunikationszeitscheibe	26
EMCY-Errorcode	27	Kommunikationszykluszeit	26
ESM	15		



Stichwortverzeichnis

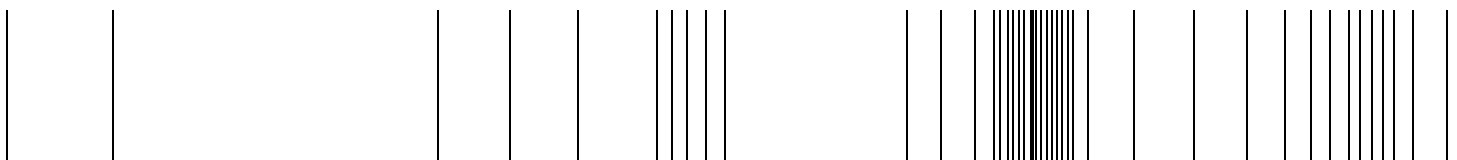
Komplexe Datentype	20	RxPDO	25
Konfiguration	27	RX-PDO1	49
Kurzreferenz	69		
L		S	
Lichtwellenleiter	10	Safe-Operational	15
Linientopologie	10	SDO	20, 40
Literatur zum Thema CAN	9	SDO Protokoll	43
		SDO-Zugriffe	43
M		Segmented Transfer	42
Mailbox-Header	40	Service Daten Objekte	40
Manufacturer Device Name	36	Servicedaten	20
Manufacturer Hardware Version	36	Sollwerte	46
Manufacturer Software Version	36	Speicher	93
Mapping	45	Statische Datentypen	20
Mapping konfigurieren	50	Statuswort	49
Montage	5	Steckverbinder	93
Multiplexor	41	Sterntopologie	10
		Steuerwort	49
N		Subindex	70
NMT	20	Sync Manager Communication Type	37, 38
Number of assigned RxPDOs	38	Sync Manager Synchronization	39
		synchron	20
O		Synchronisieren	47
Object Dictionary	24	Synchronisierung	17, 44, 51
Objekte	35	SyncManager	6, 44
Objektnummer	43	SyncManager Channels	44
Objektnummern	70		
Objektverzeichnis	20	T	
Offset	32	Telegramm-Header	40
Operational	16	Topologie Daten	10
		Torque Mode	22, 24
P		Transmit PDO Mapping	37
Parameternummer	70	TxPDO	25
PDO	20, 44		
PDO-Mapping	20, 44, 70	V	
Position Control Function	71, 73	Velocity Mode	23, 24, 70
Position Control Funktion	23	Vordefinierte Nachrichten	20
Position Mode	23		
Positioniermodus	33	W	
Pre-Operational	15	Wertebereich	43
Priorität	20		
Profile Velocity Mode	71	X	
Profilstruktur	35	XML-Datei	6
Prozessdaten	11, 20, 25		
Prozessdatenistwerte	26	Z	
Prozess-Daten-Objekte	44	Zugriffe	
Prozessdatensollwerte	26	fehlerhaft	43
		Zustandsmaschine	21
R			
Rahmenaufbau	11		
Receive PDO Mapping	37		
Referenzfahrt	32		
Regler Fehlercode	54, 61		
Regler-Fehlermeldungen	54		



Revisionsübersicht

Version	Stand	Änderungen
5.07017.01	08.11.2007	Neuerstellung
5.07017.02	14.04.2011	Kapitel 3.2.6 Ethernet over EtherCAT (EoE) - TCP/IP- Tunneling über EtherCAT ergänzt.
5.07017.03	14.08.2014	Kapitel 3.2.6 geändert

be in motion



Baumüller Nürnberg GmbH Ostendstraße 80-90 90482 Nürnberg T: +49(0)911-5432-0 F: +49(0)911-5432-130 www.baumueller.de

Alle Angaben in diesem Programmierhandbuch sind unverbindliche Kundeninformationen, unterliegen einer ständigen Weiterentwicklung und werden fortlaufend durch unseren permanenten Änderungsdienst aktualisiert. Bitte beachten Sie, dass Angaben/Zahlen/Informationen aktuelle Werte zum Druckdatum sind.
Zur Ausmessung, Berechnung und Kalkulationen sind diese Angaben nicht rechtlich verbindlich. Bevor Sie in diesem Programmierhandbuch aufgeführte Informationen zur Grundlage eigener Berechnungen und/oder Verwendungen machen, informieren Sie sich bitte, ob Sie den aktuellsten Stand der Informationen besitzen.
Eine Haftung für die Richtigkeit der Informationen wird daher nicht übernommen.