

# Kommunikations- / Funktionshandbuch

Motion Control

MCBL 300x CO  
MCDC 300x CO  
3564...B CO  
32xx...BX4 CO  
22xx...BX4 COD

**CAN**open®

## Impressum

---

Version:  
3. Auflage, 09.12.2020

Copyright  
by Dr. Fritz Faulhaber GmbH & Co. KG  
Daimlerstr. 23 / 25 · 71101 Schönaich

Alle Rechte, auch die der Übersetzung, vorbehalten.  
Ohne vorherige ausdrückliche schriftliche Genehmigung  
der Dr. Fritz Faulhaber GmbH & Co. KG darf kein Teil  
dieser Beschreibung vervielfältigt, reproduziert, in einem  
Informationssystem gespeichert oder verarbeitet oder in  
anderer Form weiter übertragen werden.

Dieses Dokument wurde mit Sorgfalt erstellt.  
Die Dr. Fritz Faulhaber GmbH & Co. KG übernimmt jedoch  
für eventuelle Irrtümer in diesem Dokument und  
deren Folgen keine Haftung. Ebenso wird keine Haftung  
für direkte Schäden oder Folgeschäden übernommen,  
die sich aus einem unsachgemäßen Gebrauch der Geräte  
ergeben.

Bei der Anwendung der Geräte sind die einschlägigen  
Vorschriften bezüglich Sicherheitstechnik und Funkentstörung  
sowie die Vorgaben dieses Dokuments zu beachten.

Änderungen vorbehalten.

Die jeweils aktuelle Version dieses Dokuments  
finden Sie auf der Internetseite von FAULHABER:  
[www.faulhaber.com](http://www.faulhaber.com)

# Inhalt

<b>1</b>	<b>Zu diesem Dokument .....</b>	<b>6</b>
1.1	Gültigkeit dieses Dokuments .....	6
1.2	Mitgeltende Dokumente .....	6
1.3	Umgang mit diesem Dokument .....	7
1.4	Abkürzungsverzeichnis .....	7
1.5	Symbole und Kennzeichnungen .....	7
<b>2</b>	<b>Schnellstart .....</b>	<b>8</b>
2.1	Start mit unkonfiguriertem Controller .....	8
2.2	Knotennummer und Baudrate einstellen .....	9
2.3	Betrieb über den FAULHABER Motion Manager .....	10
2.4	Betrieb über eigene Host-Anwendung .....	13
2.4.1	CANopen-Knoten aktivieren .....	13
2.4.2	Antriebe konfigurieren .....	13
2.4.3	Betrieb in einem der CANopen CiA 402 Antriebsprofile .....	14
<b>3</b>	<b>CANopen-Protokollbeschreibung .....</b>	<b>15</b>
3.1	Einführung .....	15
3.2	Kommunikationsdienste .....	16
3.3	Identifizierungs-Verteilung .....	18
3.4	PDO (Prozessdatenobjekt) .....	19
3.4.1	PDO-Konfiguration .....	19
3.4.2	PDO-Mapping in der Standardkonfiguration (Auslieferungszustand) .....	21
3.4.3	Behandlung von Mapping-Fehlern .....	22
3.5	SDO (Servicedatenobjekt) .....	23
3.5.1	Expedited Transfer .....	23
3.5.2	SDO-Fehlerbeschreibung .....	25
3.6	Emergency-Objekt (Fehlermeldung) .....	26
3.7	SYNC-Object .....	28
3.7.1	Triggern Synchroner PDOs .....	28
3.8	NMT (Netzwerkmanagement) .....	29
3.8.1	Boot-Up .....	31
3.8.2	Überwachungsfunktionen .....	32
3.8.2.1	Node Guarding .....	32
3.8.2.2	Heartbeat .....	33
3.8.3	Einstellung der Überwachungsfunktionen .....	34
3.9	Einträge im Objektverzeichnis .....	34
3.10	Fehlerbehandlung .....	35
3.10.1	CAN-Fehler .....	35
3.10.2	Gerätefehler .....	36
<b>4</b>	<b>Funktionsbeschreibung .....</b>	<b>38</b>
4.1	Antriebsdaten .....	38
4.2	Device Control .....	40
4.2.1	Zustandsmaschine des Antriebs .....	40
4.2.2	Controlword .....	42

# Inhalt

4.2.2.1	Beispiel: Enable Operation.....	43
4.2.2.2	Beispiel: Reset des Fehlerzustands.....	44
4.2.3	Quick Stop .....	44
4.2.4	Statusword .....	45
4.2.5	Betriebsart wählen.....	47
4.3	Factor Group .....	48
4.4	Profile Position Mode .....	52
4.4.1	Betriebsart im Überblick.....	52
4.4.2	Benachrichtigung der übergeordneten Steuerung .....	52
4.4.3	Grundeinstellungen .....	53
4.4.4	Abfrage aktueller Werte / Position Control Function.....	54
4.4.5	Zusätzliche Einstellungen .....	54
4.4.5.1	Inkrementalencoder als Positionsgeber .....	54
4.4.5.2	Rampengenerator.....	54
4.4.5.3	Drehzahlregler / Strombegrenzung.....	54
4.4.5.4	Befehle zur Bewegungssteuerung.....	55
4.5	Homing Mode .....	57
4.5.1	Homing-Methoden .....	57
4.5.2	Ablauf einer Homing-Referenzfahrt .....	63
4.6	Profile Velocity Mode .....	64
4.6.1	Betriebsart im Überblick.....	64
4.6.2	Benachrichtigung der übergeordneten Steuerung .....	64
4.6.3	Grundeinstellungen .....	65
4.6.4	Drehzahlwert.....	65
4.6.5	Zusätzliche Einstellungen .....	66
4.6.5.1	Grenzen der Bewegung .....	66
4.6.5.2	Rampengenerator.....	66
4.6.5.3	Strombegrenzung.....	66
4.6.6	Befehle zur Bewegungssteuerung.....	66
4.6.7	Zusammengesetzte Bewegungsprofile .....	66
4.7	Cyclic Synchronous Position Mode .....	67
4.7.1	Betriebsart im Überblick.....	67
4.7.2	Eingänge und Ausgaben .....	67
4.7.3	Statusword und Controlword .....	68
4.7.4	Einstellungen.....	68
4.8	Eingänge/Ausgänge .....	69
4.8.1	Endschalteranschlüsse und Schaltpegel.....	69
4.8.1.1	Konfiguration der digitalen Eingänge .....	69
4.8.2	Sonderfunktionen des Fault-Pins .....	71
4.8.2.1	Fault-Pin als Fehlerausgang .....	72
4.8.2.2	Fault-Pin als digitaler Ausgang .....	72
4.8.2.3	Fault-Pin als Referenzeingang .....	72
4.8.2.4	Fault-Pin als „Position erreicht“-Ausgang.....	72
4.8.3	Abfrage der Eingangszustände.....	73
4.9	Abfrage des Gerätezustands .....	74

# Inhalt

4.10	Technische Informationen .....	75
4.10.1	Rampengenerator .....	75
4.10.1.1	Grundeinstellungen .....	75
4.10.1.2	Rampengenerator im Profile Velocity Mode .....	76
4.10.1.3	Rampengenerator im Profile Position Mode .....	77
4.10.2	Sinuskommutierung .....	78
4.10.3	Stromregler und I <sup>2</sup> t-Strombegrenzung .....	79
4.10.4	Übertemperatursicherung .....	80
4.10.5	Unterspannungsüberwachung .....	81
4.10.6	Überspannungsregelung .....	81
4.10.7	Einstellung der Reglerparameter .....	81
4.10.7.1	Standardverhalten .....	81
4.10.7.2	Zur Verfügung stehende Reglerparameter .....	82
4.10.7.3	Mögliche Vorgehensweise .....	83
<b>5</b>	<b>Inbetriebnahme .....</b>	<b>84</b>
5.1	Kommunikationseinstellungen .....	84
5.1.1	Einstellung über das CAN-Netzwerk .....	84
5.1.1.1	Einstellung der Knotennummer .....	85
5.1.1.2	Einstellung der Baudrate .....	85
5.1.2	Knotennummer auslesen .....	85
5.2	Grundeinstellungen .....	86
5.3	Konfiguration mit dem Motion Manager .....	87
5.3.1	Verbindung herstellen .....	88
5.3.2	Motor auswählen .....	89
5.3.3	Antrieb konfigurieren .....	89
5.3.3.1	Grundeinstellungen durchführen .....	90
5.3.3.2	Factor Group .....	91
5.3.3.3	Reglerparameter einstellen .....	91
5.3.3.4	Fehlerbehandlung .....	93
5.3.3.5	Ein-/Ausgänge und Homing einstellen .....	93
5.3.4	Datensatz verwalten .....	94
5.3.5	Diagnose .....	95
5.3.5.1	Statusanzeige .....	95
5.3.5.2	Trace-Funktion .....	95
<b>6</b>	<b>Parameterbeschreibung .....</b>	<b>96</b>
6.1	Kommunikationsobjekte nach CiA 301 .....	96
6.2	Herstellerspezifische Objekte .....	104
6.3	Objekte des Antriebsprofils CiA 402 .....	110

## Zu diesem Dokument

# 1 Zu diesem Dokument

## 1.1 Gültigkeit dieses Dokuments

Dieses Dokument beschreibt:

- Schnellstart:
  - Erstmalige Inbetriebnahme und Bedienung des Geräts mit CANopen-Schnittstelle
- Kommunikation:
  - Kommunikation mit dem Antrieb über CANopen
  - Basisdienste der Kommunikationsstruktur
  - Methoden für den Parameterzugriff
  - Antrieb aus Kommunikationssicht
- Funktion:
  - Prinzip der Gerätesteuerung
  - Inbetriebnahme und Konfiguration des Geräts
  - Betriebsarten und Funktionen

Dieses Dokument richtet sich an folgende Personen:

- Anwender, die erstmalig einen Motor am FAULHABER Motion Controller in Betrieb nehmen
- Softwareentwickler und Projekt Ingenieure mit Schnittstellen-Erfahrung
- Techniker und Ingenieure mit Erfahrung in der Applikation von geregelten elektrischen Antrieben sowie industriellen Kommunikationssystemen

Alle Angaben in diesem Dokument beziehen sich auf Standardausführungen der Antriebe. Änderungen aufgrund kundenspezifischer Ausführungen dem entsprechenden Datenblatt entnehmen.

## 1.2 Mitgeltende Dokumente

Für bestimmte Handlungsschritte bei der Inbetriebnahme und Bedienung der FAULHABER Produkte sind zusätzliche Informationen aus folgenden Handbüchern hilfreich:

Handbuch	Beschreibung
Motion Manager 6	Bedienungsanleitung zur FAULHABER Motion Manager PC Software
Gerätehandbuch	Anleitung zur Installation und zum Gebrauch des FAULHABER Motion Controllers

Diese Handbücher können im PDF-Format von der Internetseite [www.faulhaber.com/manuals](http://www.faulhaber.com/manuals) heruntergeladen werden.

## Zu diesem Dokument

### 1.3 Umgang mit diesem Dokument

- ▶ Dokument vor der Konfiguration aufmerksam lesen.
- ▶ Dokument während der Lebensdauer des Produkts aufbewahren.
- ▶ Dokument dem Bedienpersonal jederzeit zugänglich halten.
- ▶ Dokument an jeden nachfolgenden Besitzer oder Benutzer des Produkts weitergeben.

### 1.4 Abkürzungsverzeichnis

Abkürzung	Bedeutung
EEPROM	Electrically Erasable Programmable Read-Only Memory
MOSFET	Metal-Oxide Semiconductor Field-Effect Transistor
PWM	Pulse Width Modulation
PLC	Speicherprogrammierbare Steuerung
TTL	Transistor Transistor Logic

### 1.5 Symbole und Kennzeichnungen



#### **VORSICHT!**

**Gefahren für Personen. Nichtbeachtung kann zu leichten Verletzungen führen.**

- ▶ Maßnahme zur Vermeidung



#### **HINWEIS!**

**Gefahr von Sachschäden.**

- ▶ Maßnahme zur Vermeidung



Hinweise zum Verständnis oder zum Optimieren der Arbeitsabläufe

- ✓ Voraussetzung zu einer Handlungsaufforderung
  1. Erster Schritt einer Handlungsaufforderung
    - ↩ Resultat eines Schritts
  2. Zweiter Schritt einer Handlungsaufforderung
    - ↩ Resultat einer Handlung
- ▶ Einschrittige Handlungsaufforderung

## 2 Schnellstart

Für einen einfachen Einstieg sind in diesem Kapitel die ersten Schritte zur Inbetriebnahme und Bedienung der FAULHABER Motion Controller mit CANopen-Schnittstelle aufgezeigt. Zusätzlich muss die ausführliche Dokumentation gelesen und berücksichtigt werden, insbesondere Kap. 5.3.3.1, S. 90.

### 2.1 Start mit unkonfiguriertem Controller

Im unkonfigurierten Zustand ist an den Motion Control Systemen die Knotennummer 255 voreingestellt und die automatische Erkennung der Baudrate ist aktiv.

1. Knotennummer und Baudrate per LSS einstellen.

Die korrekte Knotennummer und Baudrate wird über den LSS-Dienst entsprechend CiA 305 eingestellt. Dazu kann der FAULHABER Motion Manager oder ein anderes CANopen-Konfigurationswerkzeug verwendet werden.

2. Die Kommunikation zum Antriebsknoten aufbauen.

☞ Der Antriebsknoten erscheint mit der korrekten Bezeichnung in der Baumansicht des Motion Managers.

3. Passende Motor- und Reglerparameter über den Assistenten für Motorauswahl einstellen.

☞ Die Regler für den verwendeten Motor und die entsprechende Last werden vorkonfiguriert.

4. Anwendungsparameter über den Konfigurationsdialog einstellen.

- Über den Konfigurationsdialog **Antriebsfunktionen** mindestens die Grundeinstellungen (Betriebsart, Bereichsgrenzen etc.) auf die Anwendung anpassen.
- Bei externen BL-Controllern eine Optimierung der Hallsensorsignale durchführen.

5. Antrieb über das Tool **Regler-Tuning** betreiben.

Beim Erstbetrieb den Motor über das Tool **Regler-Tuning** in den gewünschten Betriebsarten betreiben. Damit können die Reglerverstärkungen genau an die Anwendung angepasst werden.


6. Weitere Einstellungen durchführen.

Weitere anwendungsspezifische Einstellungen können gegebenenfalls über den Konfigurationsdialog durchgeführt werden. Alternativ kann der Antrieb auch direkt an der eigenen Steuerung in Betrieb genommen werden.



### 2.2 Knotennummer und Baudrate einstellen


Die Einheiten werden standardmäßig ohne gültige Knotenadresse (Node-ID = 255) und mit eingestellter automatischer Baudratenerkennung (AutoBaud) ausgeliefert.

 Zur Einstellung von Baudrate und Knotenadresse muss die Einheit über CAN an ein entsprechendes Konfigurationstool angeschlossen werden, das das LSS-Protokoll (Layer Setting Services and Protocol) nach CiA DSP305 unterstützt.

Dazu kann der FAULHABER Motion Manager verwendet werden, der auf einem PC mit unterstützter CAN-Schnittstelle installiert ist. Über das LSS-kompatible Konfigurationstool kann die Knotenadresse und die Baudrate auf folgende Weisen eingestellt werden (siehe Kap. 5.1, S. 84):

- Im Global-Modus: Wenn nur ein Antrieb angeschlossen ist
- Im Selective-Modus über die Serien-Nummer: Wenn ein Antrieb im Netzwerk konfiguriert werden soll


Vorgehensweise bei Verwendung des FAULHABER Motion Managers:

1. Antriebseinheit an eine 24 V Spannungsquelle anschließen.  
Zur Belegung der Anschlussleitung siehe Gerätehandbuch.
2. Antriebseinheit an die CAN-Schnittstelle des PCs anschließen und einschalten bzw. PC mit dem CAN-Netzwerk verbinden.
3. FAULHABER Motion Manager starten.
4. Verbindung über die CAN-Schnittstelle herstellen.
5. Menü **Konfiguration - Verbindungsparameter...** auswählen.
6. LSS-Modus auswählen:
  - Falls nur ein LSS-Knoten angeschlossen ist und Sie keine weiteren Daten eingeben wollen, den Antrieb global konfigurieren (LSS Switch Mode Global).
  - Falls ein Knoten im Netzwerk konfiguriert werden soll, angegebenen Knoten selektiv konfigurieren (LSS Switch Mode Selective). Wurde der Knoten im Node-Explorer noch nicht gefunden, die Seriennummer des zu konfigurierenden Antriebsknotens eingeben. Andernfalls sind die Datenfelder bereits richtig vorbelegt.
7. Gewünschte Übertragungsrate oder **Auto** auswählen und die gewünschte Knotennummer einstellen.
8. Schaltfläche **Senden** betätigen.
  -  Die Einstellungen werden übertragen und dauerhaft im Controller gespeichert. Der Motion Manager ruft danach erneut die Scan-Funktion auf und der Knoten sollte nun im Node-Explorer mit der richtigen Knotennummer angezeigt werden. Nach erneutem Aus- und Einschalten arbeitet der Antrieb nun mit der eingestellten Konfiguration.

### 2.3 Betrieb über den FAULHABER Motion Manager

Der FAULHABER Motion Manager bietet einen einfachen Zugriff auf die CANopen-Zustandsmaschinen über Menüeinträge, die entweder über das Kontextmenü des Node-Explorers (rechte Maustaste) oder über das Menü **Befehle** aufgerufen werden können. Der gewünschte Knoten muss zuvor durch Doppelklick im Node-Explorer aktiviert werden. In der Statuszeile am unteren Rand des Motion Manager Fensters werden immer die aktuellen Zustände angezeigt.

Weitere Informationen zur Zustandsmaschine eines CANopen-Knotens siehe Kap. 3, S. 15.

 Zur vereinfachten Bedienung stellt der Motion Manager auch für die CO-Varianten spezielle Befehle zur Verfügung. Diese können direkt in die Befehlseingabezeile im Terminal-Fenster eingegeben oder aus dem Befehle-Menü ausgewählt werden. Nach dem Senden des Kommandos wird ein Befehlsinterpret durchlaufen, der den Befehl in ein entsprechendes CAN-Telegramm umwandelt.

1. Antriebsfunktionen konfigurieren:

Zur Auswahl des Motors und der dafür geeigneten Grundparameter steht im Motion Manager der Assistent für Motorauswahl zur Verfügung.

Weitere Einstellungen, z. B. zur Funktion des Fault-Ausgangs, können unter dem Menüpunkt **Konfiguration – Antriebsfunktionen** über einen komfortablen Dialog vorgenommen werden (siehe Kap. 5.3, S. 87). Der Konfigurationsdialog steht auch als Direktzugriff in der Schnellzugriffleiste des Motion Managers zur Verfügung.




#### HINWEIS!

**Schäden am Controller und/oder Antrieb durch falsche Werte in den Einstellungen der Motion Controller**

▶ Grundeinstellungen prüfen (siehe Kap. 5.3.3.1, S. 90).

2. Im Kontextmenü des Node-Explorers oder im Menü **Befehle** den Eintrag **Netzwerkmanagement - Start Remote Node** wählen.

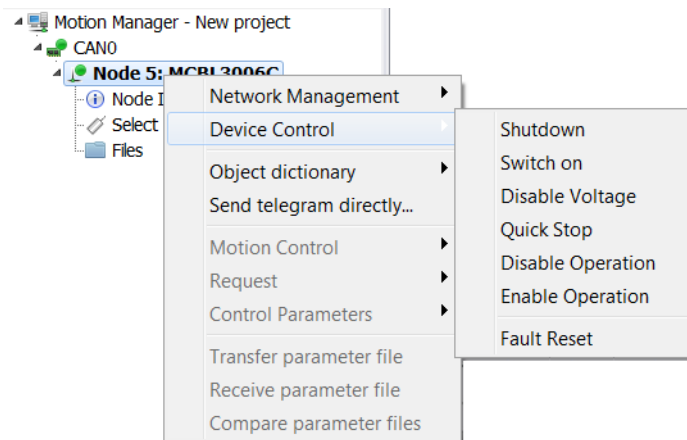
 Der Knoten befindet sich im Zustand *Operational*. PDO-Kommunikation ist nun verfügbar.

3. Antrieb über die CiA 402 Zustandsmaschine aktivieren:

Ein Antrieb nach CiA 402 muss nach einer festgelegten Schrittfolge aktiviert werden. Die nötigen Kommandos stehen im Kontextmenü des Node-Explorers oder über das Menü **Befehle** direkt zur Verfügung.

Alternativ können die Befehle auch direkt durch Eingabe von entsprechenden Motion Manager Befehlen in das Befehlseingabefeld des Terminal-Fensters ausgeführt werden.

## Schnellstart



- Über den Menüeintrag **Device Control - Shutdown** oder den Befehl `SHUTDOWN` das Einschalten des Antriebs vorbereiten.
- Über den Menüeintrag **Device Control - Switch On** oder den Befehl `SWITCHON` die Endstufe einschalten.
- Über den Menüeintrag **Device Control - Enable Operation** oder den Befehl `ENOP` den Betrieb freischalten.

Alternativ kann auch die Schaltfläche **Endstufe einschalten** gedrückt werden, um diese Schritte auf einmal auszuführen.

#### 4. Motor drehzahl geregelt antreiben (z. B. mit $100 \text{ min}^{-1}$ ):

- **Profile Velocity Mode** einstellen:  
Über das Kontextmenü des Node-Explorers oder über das Menü **Befehle** den Eintrag **Bewegungssteuerung - Betriebsart 0x6060 setzen (OPMOD)** auswählen und auf **Profile Velocity Mode (3)** setzen  
- oder -  
den Befehl `OPMOD3` in das Befehlseingabefeld des Terminal-Fensters eingeben.
- **Drehzahlvorgabe (Target Velocity)** auf den Wert 100 setzen:  
Über das Kontextmenü des Node-Explorers oder über das Menü **Befehle** den Eintrag **Bewegungssteuerung - Drehzahl 0x60FF setzen (V)** auswählen und den gewünschten Wert in das Eingabefeld eintragen  
- oder -  
den Befehl `V100` in das Befehlseingabefeld des Terminal-Fensters eingeben.
- **Motor stoppen:**  
Drehzahlvorgabe auf den Wert 0 setzen (`V0`)  
- oder -  
den Menüeintrag **Device Control - Disable Operation** auswählen bzw. den Befehl `DIOP` ausführen.

## Schnellstart

---

### 5. Motor verfahren (z. B. relativ um 10 000 Inkremente):

- *Profile Position Mode* einstellen:

Über das Kontextmenü des Node-Explorers oder über das Menü **Befehle** den Eintrag **Bewegungssteuerung - Betriebsart 0x6060 setzen (OPMOD)** auswählen und auf *Profile Position Mode (1)* setzen

- oder -

den Befehl `OPMOD1` in das Befehlseingabefeld des Terminal-Fensters eingeben.

- Sollposition (*Target Position*) auf den Wert 10 000 setzen:

Über das Kontextmenü des Node-Explorers oder über das Menü **Befehle** den Eintrag **Bewegungssteuerung - Sollposition 0x607A setzen (SPOS)** auswählen und den gewünschten Wert in das Eingabefeld eintragen

- oder -

den Befehl `SPOS10000` in das Befehlseingabefeld des Terminal-Fensters eingeben.

- Sollposition anfahren:

– Über das Kontextmenü des Node-Explorers oder über das Menü **Befehle** den Eintrag **Device Control - Enable Operation** auswählen

- oder -

den Befehl `ENOP` in das Befehlseingabefeld des Terminal-Fensters eingeben.

– Über das Kontextmenü des Node-Explorers oder über das Menü **Befehle** den Eintrag **Bewegungssteuerung - Relative Positionierung starten (MR)** auswählen

- oder -

den Befehl `MR` in das Befehlseingabefeld des Terminal-Fensters eingeben.

- 6. Endstufe ausschalten:

Schaltfläche **Endstufe ausschalten** in der Symbolleiste auswählen

- oder -

Taste F5 drücken

- oder -

Menüeintrag **Device Control - Disable Voltage** auswählen bzw. den Befehl `DISABLE` ausführen.

## Schnellstart

### 2.4 Betrieb über eigene Host-Anwendung

#### 2.4.1 CANopen-Knoten aktivieren

- Über den Broadcast-Befehl `Start Remote Node` mit der CAN-ID 0 entweder einen einzelnen Knoten oder das gesamte Netzwerk starten und in den Zustand *Operational* setzen.

11-Bit Identifier	2 Byte Nutzdaten	
ID 0x000	01	00

Das erste Datenbyte enthält den Start-Befehl `Start Remote Node`, das zweite Datenbyte die Knotenadresse oder 0 für das gesamte Netzwerk.

Nachdem der Knoten gestartet wurde, sind alle Funktionen ansprechbar. Über die Device Control Funktionen nach CiA DSP402 kann der Antrieb nun aktiviert und betrieben werden.


Die Identifier der einzelnen Objekte sind entsprechend dem *Predefined Connection Set* vor eingestellt und von der Knotennummer abhängig.

Objekt	CAN-ID	Beschreibung
TxPDO1	0x180 + Node-ID	Daten (z. B. Statuswerte) des Antriebs empfangen
RxPDO1	0x200 + Node-ID	Daten (z. B. Steuerbefehle) an den Antrieb senden
TxPDO2	0x280 + Node-ID	Daten (z. B. Statuswerte) des Antriebs empfangen
RxPDO2	0x300 + Node-ID	Daten (z. B. Steuerbefehle) an den Antrieb senden
TxPDO3	0x380 + Node-ID	Daten (z. B. Statuswerte) des Antriebs empfangen
RxPDO3	0x400 + Node-ID	Daten (z. B. Steuerbefehle) an den Antrieb senden
TxPDO4	0x480 + Node-ID	Daten (z. B. Statuswerte) des Antriebs empfangen
RxPDO4	0x500 + Node-ID	Daten (z. B. Steuerbefehle) an den Antrieb senden
TxSDO1	0x580 + Node-ID	Eintrag des Objektverzeichnisses lesen
RxSDO1	0x600 + Node-ID	Eintrag des Objektverzeichnisses schreiben

Im Auslieferungszustand befinden sich die Antriebe nach dem Einschalten im Betriebsmodus *Modes of Operation = 1 (Profile Position Mode)*. Die Antriebssteuerung erfolgt über die Device Control Zustandsmaschine, die über das Controlword (Objekt 0x6040 oder RxPDO) bedient und über das Statusword (Objekt 0x6041 oder TxPDO) abgefragt wird.

#### 2.4.2 Antriebe konfigurieren

- Antriebe mittels SDO-Transfer über die Objekte des Objektverzeichnisses konfigurieren.

 Es wird empfohlen, für die Grundeinstellungen den FAULHABER Motion Manager zu verwenden (siehe Kap. 5.3.3.1, S. 90).

### 2.4.3 Betrieb in einem der CANopen CiA 402 Antriebsprofile

Ein Antrieb nach CiA 402 muss nach einer festgelegten Schrittfolge aktiviert werden (siehe Kap. 4.2, S. 40). Der Schreibzugriff auf das Controlword kann über das Objektverzeichnis an Adresse 0x6040 oder über ein RxPDO erfolgen:

1. Shutdown-Befehl ausführen (Controlword = 0x00 06).
2. Switch-On-Befehl ausführen (Controlword = 0x00 07).
  - ↪ Der Antrieb befindet sich im Zustand *Switched On*. Um Fahrbefehle ausführen zu können muss anschließend der Betrieb freigeschaltet werden.
3. Enable-Operation-Befehl ausführen (Controlword = 0x00 0F):
  - ↪ Der Antrieb befindet sich im Zustand *Operation Enabled*, in dem er über die entsprechenden Objekte der eingestellten Betriebsart bedient werden kann (siehe Kap. 4.2, S. 40 und Kap. 4.3, S. 48).
4. Motor drehzahl geregelt antreiben (z. B. mit 500 min<sup>-1</sup>):
  - Modes of Operation (Objekt 0x6060): 3 (*Profile Velocity Mode*) per SDO-Zugriff einstellen.
  - Target Velocity (Objekt 0x60FF): 500
5. Motor stoppen:
  - Drehzahlvorgabe (Target Velocity) auf den Wert 0 setzen (Objekt 0x60FF).  
oder
  - Controlword = 0x00 07 (Disable Operation)
6. Motor verfahren (z. B. relativ um 10 000 Inkremente):
  - Modes of Operation (Objekt 0x6060): 1 (*Profile Position Mode*) per SDO-Zugriff einstellen.
  - Target Position (Objekt 0x607A): 10 000
  - Controlword = 0x00 7F (New Set-Point, Change Set Immediately, Rel)

# CANopen-Protokollbeschreibung

## 3 CANopen-Protokollbeschreibung

### 3.1 Einführung

#### CANopen

CANopen ist ein Standard-Softwareprotokoll. Für die Kommunikation mit CANopen wird eine CAN-Hardwareumgebung benötigt. Innerhalb eines CANopen-Netzwerks können bis zu 127 Knoten adressiert werden. Die maximale Übertragungsgeschwindigkeit beträgt 1 MBit/s.

#### CAN-Normung

Die CiA definiert in der CiA 301 folgende Aspekte:

- Kommunikationsstruktur
- Steuer- und Überwachungsfunktionen

Für eine Reihe von Geräteklassen wurden CANopen-Geräteprofile definiert, wie zum Beispiel:

- CiA 402 für Antriebe
- CiA 401 für Ein- und Ausgabegeräte

#### Aufbau eines CANopen-Telegramms

Ein CANopen-Telegramm besitzt einen 11-Bit-Identifizier und kann bis zu 8 Byte Nutzdaten beinhalten.

Tab. 1: Schematischer Aufbau eines CANopen-Telegramms

11-Bit-Identifizier	bis zu 8 Byte Nutzdaten							
11 Bit	8 Bit	8 Bit	8 Bit	8 Bit	8 Bit	8 Bit	8 Bit	8 Bit

# CANopen-Protokollbeschreibung

## 3.2 Kommunikationsdienste

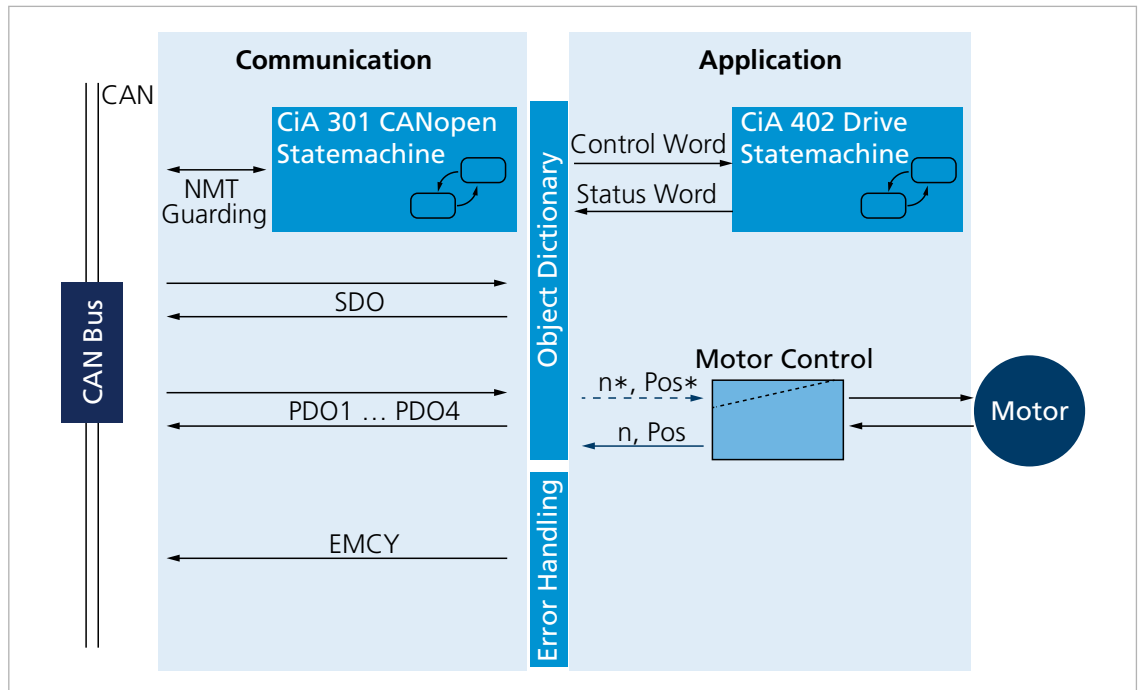


Abb. 1: Kommunikationsdienste des Motion Controllers

Der Kommunikationsteil enthält Kommunikationsdienste entsprechend CiA 301.

Tab. 2: Kommunikationsdienste nach CiA 301

Kommunikationsdienste	Beschreibung
NMT (Netzwerkmanagement)	Aktiviert Knoten und überwacht den aktuellen Zustand eines Knotens (siehe Kap. 3.8, S. 29).
SDO (Servicedatenobjekt)	Über das SDO greift der CANopen-Master auf Parameter in einem Knoten zu. Bei jedem SDO-Zugriff wird genau ein Parameter gelesen oder beschrieben. Ein SDO kann stets nur einen Knoten im Netzwerk ansprechen (siehe Kap. 3.5, S. 23).
PDO (Prozessdatenobjekt)	Über das PDO wird auf Echtzeitdaten zugegriffen. Mit einem PDO kann über eine CAN-Nachricht auf mehrere Antriebsparameter gleichzeitig zugegriffen werden. Die in einem PDO versendeten oder empfangenen Parameter können frei konfiguriert werden (siehe Kap. 3.4, S. 19).
SYNC-Objekt	Über SYNC-Objekte werden verschiedene Anwendungen am CAN-BUS synchronisiert (siehe Kap. 3.7, S. 28).
EMCY (Emergency-Objekt)	Eine Emergency-Nachricht informiert den CANopen-Master über Fehler. Eine CAN-Nachricht übermittelt asynchron den Fehlercode, sodass der Zustand des CANopen-Slaves nicht laufend nach Fehlern abgefragt werden muss (siehe Kap. 3.6, S. 26).



## CANopen-Protokollbeschreibung

---

### Kommunikationsprofil

Der FAULHABER Motion Controller unterstützt das CANopen-Kommunikationsprofil gemäß CiA 301 V4:

- 4 Sende-PDOs
- 4 Empfangs-PDOs
- 1 Server-SDO
- Emergency-Object
- NMT mit Node-Guarding und Heartbeat
- SYNC-Object



Die Datenbelegung der PDOs ist entsprechend dem "PDO set for servo drive" nach CiA 402 V3 voreingestellt und kann vom Anwender geändert werden (dynamisches PDO-Mapping).

Viele Hersteller bieten CANopen-Libraries für PC- und SPS-Systeme an, über die die einzelnen Objekte komfortabel ansprechbar sind, ohne sich mit dem internen Aufbau beschäftigen zu müssen.

## CANopen-Protokollbeschreibung

### 3.3 Identifizier-Verteilung

Der Communication Object Identifier (COB-ID) setzt sich aus einer 7-Bit Knotenadresse (Node ID) und einem 4-Bit Funktionscode zusammen.

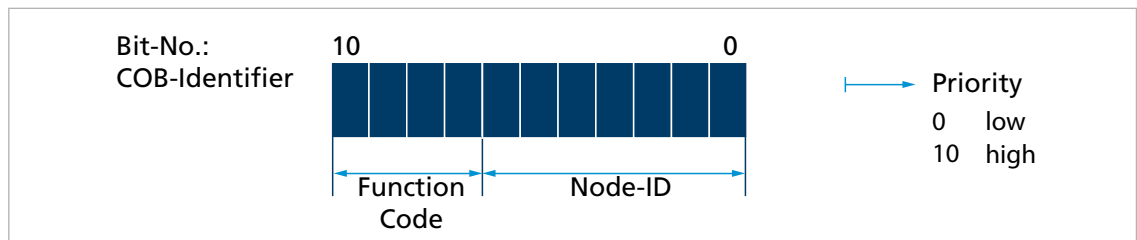


Abb. 2: Identifizier-Verteilung

Das Predefined-Connection-Set definiert Standard-Identifizier für die wichtigsten Objekte.

Tab. 3: Standard-Identifizier

Objekt	Funktionscode (binär)	Resultierende COB-ID	Objektindex für Kommunikationseinstellung
NMT	0000	0	–
SYNC	0001	128 (80h)	1005h
EMERGENCY	0001	129 (81h) bis 255 (FFh)	1014h
PDO1 (tx)	0011	385 (181h) bis 511 (1FFh)	1800h
PDO1 (rx)	0100	513 (201h) bis 639 (27Fh)	1400h
PDO2 (tx)	0101	641 (281h) bis 767 (2FFh)	1801h
PDO2 (rx)	0110	769 (301h) bis 895 (37Fh)	1401h
PDO3 (tx)	0111	897 (381h) bis 1023 (3FFh)	1802h
PDO3 (rx)	1000	1025 (401h) bis 1151 (47Fh)	1402h
PDO4 (tx)	1001	1153 (481h) bis 1279 (4FFh)	1803h
PDO4 (rx)	1010	1281 (501h) bis 1407 (57Fh)	1403h
SDO (tx)	1011	1409 (581h) bis 1535 (5FFh)	1200h
SDO (rx)	1100	1537 (601h) bis 1663 (67Fh)	1200h
NMT Error Control	1110	1793 (701h) bis 1919 (77Fh)	–

Die COB-IDs der PDOs, des SYNC-Objekts und des Emergency-Objekts können über die Kommunikationsparameter im Objektverzeichnis geändert werden. Die COB-ID der SDO-Telegramme kann nicht verändert werden und ergibt sich immer aus der Knotennummer.



Im Auslieferungszustand ist die Knotennummer 1 konfiguriert. Die COB-IDs sind entsprechend voreingestellt:

- RxPDO: 201h, 301h, 401h und 501h
- TxPDO: 181h, 281h, 381h und 481h
- EMCY: 81h
- RxSDO: 581h
- TxSDO: 601h

## CANopen-Protokollbeschreibung

- i** Bei Änderungen der Knotennummer über das LSS-Protokoll bleiben die COB-IDs der PDOs und des Emergency-Objekts unverändert. Wenn sich die COB-IDs der PDOs und des Emergency-Objekts zusammen mit der Knotennummer ändern sollen, muss die Knotennummer zunächst auf 255 (undefiniert) gesetzt und dann auf die gewünschte Nummer eingestellt werden.

### 3.4 PDO (Prozessdatenobjekt)

PDOs sind CAN-Nachrichten mit bis zu 8 Byte Nutzdaten. PDOs enthalten Prozessdaten zur Steuerung und Überwachung des Geräteverhaltens. Aus Sicht des Antriebs werden PDOs in Empfangs- und Sende-PDOs unterschieden.

- Empfangs-PDO (RxPDO): wird von einem Antrieb empfangen und enthält z. B. Steuerdaten
- Sende-PDO (TxPDO): wird von einem Antrieb gesendet und enthält z. B. Überwachungsdaten

PDOs werden nur ausgewertet oder übertragen, wenn sich das Gerät im NMT-Zustand *Operational* befindet (siehe Kap. 3.8, S. 29).

Die Übertragung von PDOs kann auf unterschiedliche Arten angestoßen werden. Das Verhalten kann für jedes PDO über den Parameter Transmission Type der Kommunikationsparameter im Objektverzeichnis eingestellt werden:

Tab. 4: PDO-Übertragungsarten

Transmission Type	Beschreibung
Ereignisgesteuert	Ereignisgesteuerte RxPDOs werden direkt beim Eintreffen verarbeitet. Ereignisgesteuerte TxPDOs werden versandt, wenn das Statusword des Geräts enthalten ist und sich geändert hat.
Remote Request (RTR)	Daten werden nach einer Anforderungsmeldung gesendet.
Synchronisiert	Daten werden nach Eintreffen eines SYNC-Objekts gesendet (siehe Kap. 3.7, S. 28).

#### 3.4.1 PDO-Konfiguration

- Maximal 4 Parameter können in einem PDO gemappt werden.
- Über die Objekte 0x1600 bis 0x1603 und 0x1A00 bis 0x1A03 kann die Datenbelegung der PDOs geändert werden. Die dafür notwendige Mapping-Prozedur ist in der CiA 301 beschrieben. Zur Durchführung der Mapping-Prozedur ist ein geeignetes Tool notwendig (z. B. FAULHABER Motion Manager oder System-Manager der verwendeten SPS-Steuerung).
- Über die Objekte 0x1400 bis 0x1403 bzw. 0x1800 bis 0x1803 können Transmission Type und COB-ID der PDOs geändert werden.
- Über den Parameter Transmission Type kann das Verhalten eines PDO eingestellt werden:

## CANopen-Protokollbeschreibung

Tab. 5: Transmission-Types eines PDOs

Transmission Type	Bedeutung
255	<p>asynchron (ereignisgesteuert)</p> <p>Betrifft nur TxPDOs, in denen das Statusword (0x6041) enthalten ist. Ändert sich dessen Wert, wird dieses PDO versandt.</p> <p>Bei allen anderen PDOs (d. h. in denen das Statusword nicht enthalten ist) hat diese Übertragungsart keine Auswirkung. Das Versenden kann jedoch über den Event Timer erzwungen werden (siehe unten).</p>
253	asynchron, auf Anforderung (RTR)
1 bis 240	<p>synchron, zyklisch</p> <p>PDO wird nach jedem SYNC-Objekt gesendet (siehe Kap. 3.7, S. 28). Der Wert ist dabei gleich der Anzahl der SYNC-Objekte, welche bis zum erneuten Senden der PDO empfangen worden sein müssen (1 = PDO wird mit jedem SYNC-Objekt gesendet)</p>
0	<p>synchron, azyklisch</p> <p>PDO wird einmal nach einem SYNC-Objekt gesendet bzw. ausgeführt, wenn sich der Inhalt des PDOs geändert hat (siehe Kap. 3.7, S. 28).</p>

### Event Timer bei TxPDOs

Wenn die Übertragungsart 255 eingestellt ist, wird zusätzlich die Einstellung des Event Timer (Subindex 5) des PDOs berücksichtigt. Der Event Timer gibt die maximale Zeitspanne zum Versenden von TxPDOs vor. Wenn seit dem letzten Versenden des TxPDOs die im Event Timer angegebene Zeit verstrichen ist, wird das PDO selbständig versendet. Damit wird erreicht, dass der Master ständig Informationen von Objekten bekommt, auch wenn diese nicht zusammen mit dem Statusword gemappt sind.

Der Wert des Event Timer wird in Millisekunden angegeben. Die einstellbare Zeitspanne beträgt 5...65 000 ms.

**i** FAULHABER empfiehlt, den Event Timer immer >100 ms einzustellen. Andernfalls wird der CAN-Bus mit Botschaften überlastet und die Leistungsfähigkeit des Gesamtsystems ist nicht mehr gewährleistet. Bei eingeschränkter Leistungsfähigkeit werden auch wichtige Botschaften (z. B. Emergency-Nachrichten) nicht mehr versandt.

- Die Bus-Last möglichst gering halten, damit Ereignisse, die zu vielen CAN-Telegrammen führen, nicht zu Fehlverhalten führen.

Damit TxPDOs unter Berücksichtigung des Event Timer verschickt werden, müssen folgende Vorbedingungen erfüllt sein:

- Der Antrieb ist im NMT-Zustand *Operational*
- Der Wert des Event Timer ist  $\geq 5$  ms
- Das TxPDO ist valide
- Die Übertragungsart ist 255 (asynchron)

Die Event Timer starten immer gleichzeitig. Für RxPDOs gibt es keine vergleichbare Funktion.

## CANopen-Protokollbeschreibung

### 3.4.2 PDO-Mapping in der Standardkonfiguration (Auslieferungszustand)

#### RxPDO1: Controlword

11-Bit-Identifizier	2 Byte Nutzdaten	
0x200 (512d) + Node ID	LB	HB

Das RxPDO1 enthält das 16-Bit-Controlword nach CiA DSP402. Das Controlword steuert die State-machine der Antriebseinheit und verweist auf den Objektindex 0x6040 im Objektverzeichnis. Die Bitaufteilung ist in Kap. 1, S. 1 beschrieben.

#### TxPDO1: Statusword

11-Bit-Identifizier	2 Byte Nutzdaten	
0x180 (384d) + Node ID	LB	HB

Das TxPDO1 enthält das 16-Bit-Statusword nach CiA 402. Das Statusword zeigt den Zustand der Antriebseinheit an und verweist auf den Objektindex 0x6041 im Objektverzeichnis. Die Bitaufteilung ist in Kap. 1, S. 1 beschrieben.

#### RxPDO2: Controlword, Target Position (PP)

11-Bit-Identifizier	6 Byte Nutzdaten					
0x300 (768d) + Node ID	LB	HB	LLB	LHB	HLB	HHB

Das RxPDO2 enthält das 16-Bit-Controlword und den 32-Bit-Wert der Zielposition (Objekt 0x607A) für den Profile Position Mode (PP).

#### TxPDO2: Statusword, Position Actual Value

11-Bit-Identifizier	6 Byte Nutzdaten					
0x280 (640d) + Node ID	LB	HB	LLB	LHB	HLB	HHB

Das TxPDO2 enthält das 16-Bit-Statusword und den 32-Bit-Wert der Istposition (Objekt 0x6064).

#### RxPDO3: Controlword, Target Velocity (PV)

11-Bit-Identifizier	6 Byte Nutzdaten					
0x400 (1024d) + Node ID	LB	HB	LLB	LHB	HLB	HHB

Das RxPDO3 enthält das 16-Bit-Controlword und den 32-Bit-Wert der Soll-drehzahl (Objekt 0x60FF) für den Profile Velocity Mode (PV).

#### TxPDO3: Statusword, Velocity Actual Value

11-Bit-Identifizier	6 Byte Nutzdaten					
0x380 (896d) + Node ID	LB	HB	LLB	LHB	HLB	HHB

Das TxPDO3 enthält das 16-Bit-Statusword und den 32-Bit-Wert der Ist-drehzahl (Objekt 0x606C).

## CANopen-Protokollbeschreibung

### RxPDO4: Controlword, Target Position Internal Value (PP)

11-Bit-Identifizier	6 Byte Nutzdaten					
0x500 (1280d) + Node ID	LB	HB	LLB	LHB	HLB	HHB

Enthält das 16-Bit Controlword und im Profile Position Mode (PP) den 32-Bit-Wert der Sollposition in internen Einheiten (Objekt 0x257A).

### TxPDO4: Position Actual Value, Velocity Actual Value

11-Bit-Identifizier	6 Byte Nutzdaten					
0x480 (1152d) + Node ID	LB	HB	LLB	LHB	HLB	HHB

Enthält den 32-Bit-Wert der Istposition (Objekt 0x6064) und den 32-Bit-Wert der Istdrehzahl (Objekt 0x606C).


### 3.4.3 Behandlung von Mapping-Fehlern

Wenn die von CiA 301 vorgegebene Mapping-Prozedur nicht eingehalten wird, wird einer der folgenden SDO-Fehler zurückgegeben:

Tab. 6: SDO-Fehler bei fehlerhafter Mapping-Prozedur

SDO-Fehler	Bedeutung	Ursache
0x06090030	Allgemeiner Wertebereichfehler	Der Mapping-Parameter ist nicht in der vorgegebenen Mapping-Prozedur beschrieben.
0x06020000	Objekt nicht im Objektverzeichnis vorhanden	Der Wert für die Anzahl gemappter Objekte ist größer als die Anzahl der gültigen Einträge in den jeweiligen Subindexen der Mapping-Parameter-Objekte.

Ist die Anzahl gemappter Objekte 0, wird das PDO intern als ungültig markiert und nicht bedient.

 Weitere Mapping-Fehler sind in der SDO-Fehlertabelle beschrieben (siehe Kap. 3.5.2, S. 25).

## CANopen-Protokollbeschreibung

### 3.5 SDO (Servicedatenobjekt)

Das SDO liest und beschreibt Parameter im OV (Objektverzeichnis). Über den 16-Bit-Index und den 8-Bit-Subindex greift das SDO auf das Objektverzeichnis zu. Der Motion Controller stellt auf Anforderung des Clients (PC, SPS (Speicherprogrammierbare Steuerung)) Daten zur Verfügung (Upload) bzw. empfängt Daten vom Client (Download).

Tab. 7: Allgemeine Strukturierung der SDO-Nutzdaten

Byte0	Byte 1 bis 2	Byte 3	Byte 4 bis 7
Command Specifier	16-Bit-Index	8-Bit-Subindex	4 Byte-Parameter-Data

Tab. 8: Einteilung der SDO-Übertragungsarten

Übertragungsart	Byteanzahl	Verwendungszweck
Expedited Transfer	maximal 4 Byte	Lesen und Beschreiben einzelner numerischer Parameter
Segmented Transfer	mehr als 4 Byte	Lesen von Text-Parametern (z. B. Gerätename, Firmware-Version) und Übertragung von Datenblöcken (z. B. Trace-Puffer)

In diesem Dokument ist nur der Expedited Transfer beschrieben. Der Segmented Transfer ist in der CiA 301 beschrieben.

#### 3.5.1 Expedited Transfer

SDO-Nachrichten sind immer 8 Byte groß.

##### Lesen von OV-Einträgen (Client-to-Server, Upload-Request)

11-Bit-Identifizier	8 Byte Nutzdaten							
0x600 (1536d) + Node ID	0x40	Index LB	Index HB	Subindex	0	0	0	0

##### Server-to-Client, Upload-Response

11-Bit-Identifizier	8 Byte Nutzdaten							
0x580 (1408d) + Node ID	CS(0x4x)	Index LB	Index HB	Subindex	LLB (D0)	LHB (D1)	HLB (D2)	HHB (D3)

Der Command Specifier CS(0x4x) gibt die Anzahl der gültigen Datenbytes in D0 bis D3 und die Transferkennung an. Der Command Specifier ist wie folgt codiert:

- CS = 0x4F, 1 Datenbyte in D0
- CS = 0x4B, 2 Datenbytes in D0 bis D1
- CS = 0x47, 3 Datenbytes in D0 bis D2
- CS = 0x43, 4 Datenbytes in D0 bis D3

# CANopen-Protokollbeschreibung

## Schreiben von OV-Einträgen (Client-to-Server, Download-Request)

11-Bit-Identifizier	8 Byte Nutzdaten							
0x600 (1536d) + Node ID	CS(0x2x)	Index LB	Index HB	Subindex	LLB (D0)	LHB (D1)	HLB (D2)	HHB (D3)

Der Command Specifier CS(0x2x) gibt die Anzahl der gültigen Datenbytes in D0 bis D3 und die Transferkennung an. Der Command Specifier ist wie folgt codiert:

- CS = 0x2F, 1 Datenbyte in D0
- CS = 0x2B, 2 Datenbytes in D0 bis D1
- CS = 0x27, 3 Datenbytes in D0 bis D2
- CS = 0x23, 4 Datenbytes in D0 bis D3
- CS = 0x22, keine Angabe der Anzahl Datenbytes

## Server-to-Client, Download-Response

11-Bit-Identifizier	8 Byte Nutzdaten							
0x580 (1407d) + Node ID	0x60	Index LB	Index HB	Subindex	0	0	0	0

## Abbruch bei SDO-Fehlern

### SDO-Abort Client-to-Server

11-Bit-Identifizier	8 Byte Nutzdaten							
0x600 (1536d) + Node ID	0x80	Index LB	Index HB	Subindex	ERROR 0	ERROR 1	ERROR 2	ERROR 3

### SDO-Abort Server-to-Client

11-Bit-Identifizier	8 Byte Nutzdaten							
0x580 (1536d) + Node ID	0x80	Index LB	Index HB	Subindex	ERROR 0	ERROR 1	ERROR 2	ERROR 3



## CANopen-Protokollbeschreibung

### 3.5.2 SDO-Fehlerbeschreibung

Kann das SDO-Protokoll auf einer Seite nicht weiter verarbeitet werden, wird ein SDO-Abort-Telegramm versendet (siehe Kap. 3.5.1, S. 23). Die Fehlerarten sind wie folgt codiert:

- Error0: Zusätzlicher Fehlercode HB
- Error1: Zusätzlicher Fehlercode LB
- Error2: Fehlercode
- Error3: Fehlerklasse

Fehler-klasse	Fehler-code	Zusatz-code	Beschreibung
0x05	0x03	0x0000	Toggle-Bit nicht geändert
0x05	0x04	0x0001	SDO-Command-Specifier ungültig oder unbekannt
0x06	0x01	0x0000	Zugriff auf dieses Objekt wird nicht unterstützt
0x06	0x01	0x0001	Versuch, einen Write-Only-Parameter zu lesen
0x06	0x01	0x0002	Versuch, auf einen Read-Only-Parameter zu schreiben
0x06	0x02	0x0000	Objekt nicht im Objektverzeichnis vorhanden
0x06	0x04	0x0041	Objekt kann nicht in PDO gemappt werden
0x06	0x04	0x0042	Anzahl und/oder Länge der gemappten Objekte würde PDO-Länge überschreiten
0x06	0x04	0x0043	Allgemeine Parameter-Inkompatibilität
0x06	0x04	0x0047	Allgemeiner interner Inkompatibilitätsfehler im Gerät
0x06	0x07	0x0010	Datentyp oder Parameterlänge stimmen nicht überein oder sind unbekannt
0x06	0x07	0x0012	Datentypen stimmen nicht überein, Parameterlänge zu groß
0x06	0x07	0x0013	Datentypen stimmen nicht überein, Parameterlänge zu klein
0x06	0x09	0x0011	Subindex nicht vorhanden
0x06	0x09	0x0030	Allgemeiner Wertebereichfehler
0x06	0x09	0x0031	Wertebereichfehler: Parameterwert zu groß
0x06	0x09	0x0032	Wertebereichfehler: Parameterwert zu klein
0x06	0x09	0x0036	Wertebereichfehler: Maximumwert kleiner als Minimumwert
0x08	0x00	0x0000	Allgemeiner SDO-Fehler
0x08	0x00	0x0020	Zugriff nicht möglich
0x08	0x00	0x0022	Zugriff bei aktuellem Gerätestatus nicht möglich

## CANopen-Protokollbeschreibung

### 3.6 Emergency-Objekt (Fehlermeldung)

Das Emergency-Objekt informiert asynchron andere Busteilnehmer über Fehler und muss nicht abgefragt werden. Das Emergency-Objekt ist immer 8 Byte groß:

11-Bit-Identifizier		8 Byte Nutzdaten						
0x80 (128d) + Node ID	Error0(LB)	Error1(HB)	Error-Reg	FE0 (LB)	FE1 (HB)	0	0	0

Belegung der Nutzdaten:

- Error0(LB)/Error1(HB): 16-Bit-Error-Code
- Error-Reg: Error-Register (Inhalt von Objekt 0x1001, siehe Kap. 6.1, S. 96)
- FE0(LB)/FE1(HB): 16-Bit FAULHABER Fehlerregister (Inhalt von Objekt 0x2320, siehe Tab. 12)
- Bytes 5 bis 7: unbenutzt (0)

Das Error Register kennzeichnet die Fehlerart. Die einzelnen Fehlerarten sind bitcodiert und den jeweiligen Error Codes zugeordnet. Über das Objekt 0x1001 kann der letzte Wert des Error Registers abgefragt werden.

Tab. 9 listet alle Fehler auf, die über Emergency-Nachrichten gemeldet werden, sofern der entsprechende Fehler in der Emergency-Mask (0x2321.01) für das FAULHABER Fehlerregister gesetzt ist (Tab. 13).

Tab. 9: *Emergency Error Codes*

Error Code	Error	Error Mask 0x2321	Error Register Bit
0x0000	No error	–	–
0x1000	Generic error	–	0
0x2000	Current	–	–
0x2300	Current, device output side		
0x2310	Continuous over current	0x0001	1
0x3000	Voltage	–	–
0x3200	Voltage inside the device	–	–
0x3210	Overvoltage	0x0004	2
0x4000	Temperature		
0x4300	Drive temperature	–	–
0x4310	Overtemperature	0x0008	3
0x5000	Device hardware	–	–
0x5500	Data storage	–	–
0x5530	Flash memory error	0x0010	5
0x6000	Device software	–	–
0x6100	Internal software	0x1000	5
0x8000	Monitoring	–	–
0x8100	Communication	–	–
0x8110	CAN Overrun (objects lost)	0x0080	4
0x8120	CAN in error passive mode	0x0040	4

## CANopen-Protokollbeschreibung

Error Code	Error	Error Mask 0x2321	Error Register Bit
0x8130	Life guard or heartbeat error	0x0100	4
0x8140	Recovered from bus off	0x0200	4
0x8200	Protocol error	–	–
0x8210	PDO not processed due to length error	0x4000	4
0x8220	PDO length exceeded	0x2000	4
0x8400	Velocity speed controller (deviation)	0x0002	5
0x8600	Positioning controller	–	–
0x8611	Following error (deviation)	0x0002	5
0xFF00	Device specific	–	–
0xFF01	Conversion overflow	0x0800	0

### Beispiel:

Ein Emergency-Telegramm mit 8 Datenbytes wird verschickt (siehe Tab. 10), wenn folgende Bedingungen zutreffen:

- In der Error Mask 0x2321 des FAULHABER Fehlerregisters ist unter Subindex 1 Bit 1 gesetzt.
- Der über Objekt 0x2333 eingestellte Dauerstrombegrenzungswert wurde länger als die über Objekt 0x2322 eingestellte Fehlerverzögerungszeit überschritten.

Tab. 10: Beispielhafte Nutzdatenbelegung einer Emergency-Nachricht

8 Byte Nutzdaten							
0x10	0x23	0x01	0x00	0x00	0x00	0x00	0x00

### Deviation-Fehler

Im Profile Velocity Mode wird der Emergency-Fehler 0x8611 und im Profile Position Mode der Emergency-Fehler 0x8400 versendet, wenn die über Objekt 0x2322.02 eingestellte maximal zulässige Drehzahlabweichung überschritten wurde. Der Fehler wird zurückgesetzt, wenn die DSP402-Zustandsmaschine geschaltet oder eine neue Positionierung gestartet wird.

# CANopen-Protokollbeschreibung

## 3.7 SYNC-Object

Das SYNC-Object ist eine Nachricht ohne Nutzdaten. Das SYNC-Object wird zum Triggern synchroner PDOs und zum gleichzeitigen Starten von Prozessen auf verschiedenen Geräten verwendet.

Der Identifier des SYNC-Objekts wird im Objektverzeichnis unter Index 0x1005 eingestellt (standardmäßig 0x80).

11-Bit-Identifier	0 Byte Nutzdaten
0x80	keine Nutzdaten

**i** Damit ein SYNC-Object ein PDO triggert, muss der Transmission Type des zu triggern- den PDOs entsprechend eingestellt werden (siehe Tab. 5).

### 3.7.1 Triggern Synchroner PDOs

**Synchrone RxPDO:** Der mit dem PDO übertragene Befehl wird erst nach Erhalt eines SYNC Objekts ausgeführt. Der Transmission Type 1 bis 240 eines RxPDO ist identisch mit dem Transmission Type 0.

**Synchrone TxPDO:** Sofort nach Erhalt eines SYNC-Objekts werden die synchronen TxPDOs mit den aktuellen Daten verschickt.

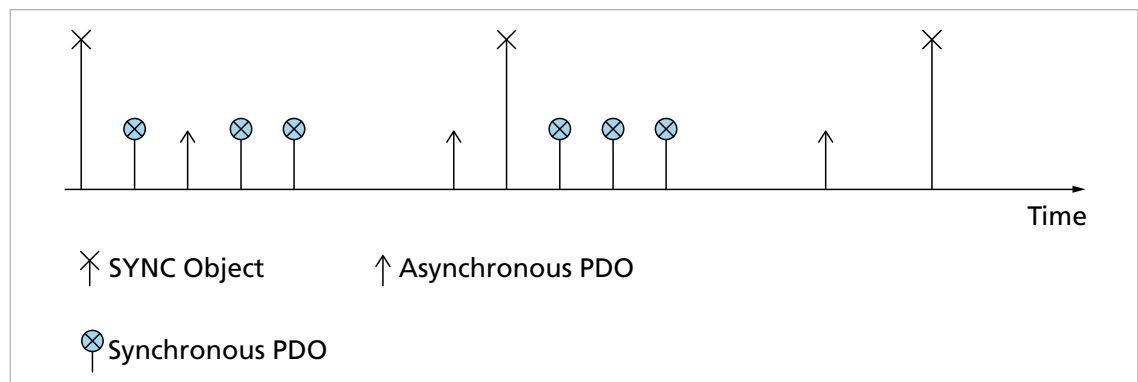


Abb. 3: Schaubild TxPDO mit SYNC

**i** Mit den Übertragungsarten 1-240 können Knoten auch gruppiert werden.

# CANopen-Protokollbeschreibung

## 3.8 NMT (Netzwerkmanagement)

Das Netzwerkmanagementobjekt steuert die CiA 301 State machine des CANopen-Geräts und überwacht Netzwerk-Knoten.

Nach dem Einschalten und der Initialisierung befindet sich der Motion Controller automatisch im Zustand *Pre-Operational*. Im Zustand *Pre-Operational* kann nur mit NMT-Nachrichten und über SDOs mit dem Gerät kommuniziert werden.

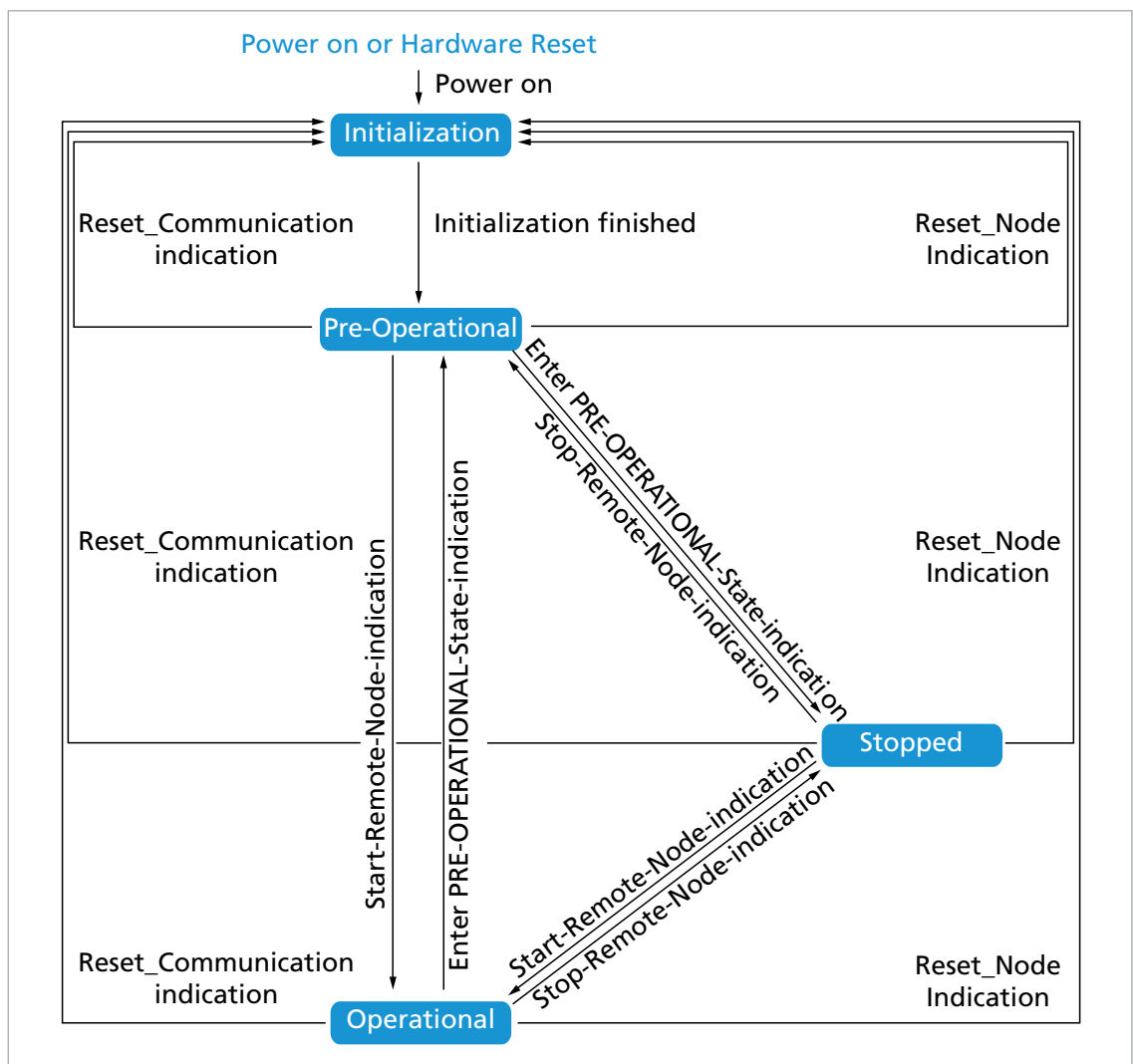


Abb. 4: CiA 301 State machine

Tab. 11: NMT-Zustandsänderungen

Statusübergang	CS	Bedeutung
Power on	–	Der Initialisierungs-Status wird beim Einschalten selbsttätig erreicht.
Initialization finished	–	Nach der Initialisierung befindet sich das Gerät automatisch im Pre-Operational-Status und eine Boot-Up-Nachricht wird abgeschickt.
Start Remote-Node indication	0x01 (1d)	Startet das Gerät und gibt die Übertragung von PDOs frei.
Enter Pre-Operational-State indication	0x80 (128d)	Stoppt die PDO-Übertragung, SDO sind weiter aktiv.

## CANopen-Protokollbeschreibung

Statusübergang	CS	Bedeutung
Stop Remote-Node indication	0x02 (2d)	Antrieb geht in den Stopped-Zustand, SDO und PDO sind abgeschaltet.
Reset Node indication	0x81 (129d)	Führt einen Reset durch. Alle Objekte werden auf Power-On-Standards zurückgesetzt.
Reset Communication indication	0x82 (130d)	Führt einen Reset der Kommunikationsfunktionen durch. Einstellungen des Benutzers bleiben erhalten.

**i** Die FAULHABER Motion Controller sind mit Standardkonfiguration für alle Objekte ausgestattet. Nach abgeschlossener Inbetriebnahme können die anwendungsspezifischen Einstellungen direkt im Gerät gespeichert werden. Im Regelfall ist daher keine weitere Parametrierung beim Systemstart notwendig.

### Starten eines CANopen Knotens

Start Remote-Node:

11-Bit-Identifizier	2 Byte Nutzdaten
0x000	0x01 Node ID

Mit einer CAN-Nachricht kann auch ein gesamtes Netzwerk gestartet werden:

Start All Remote-Nodes:

11-Bit-Identifizier	2 Byte Nutzdaten
0x000	0x01 0x00

Nach Start des Knotens oder des gesamten Netzwerks befindet sich das Gerät im Zustand *Operational*. Das Gerät kann dann über PDOs bedient werden.

Im Zustand *Stopped* befindet sich das Gerät im Fehlerzustand und kann nicht mehr über SDO oder PDO bedient werden. Die Kommunikation mit dem Gerät ist hier nur mit NMT-Nachrichten möglich.

Eine NMT-Nachricht besteht immer aus 2 Byte auf dem Identifizier 0x000.

### NMT-Nachricht

11-Bit-Identifizier	2 Byte Nutzdaten
0x000	CS Node ID

Belegung der Nutzdaten:

- CS: Command Specifier (siehe Tab. 11)
- Node ID: Knotenadresse (0 = alle Knoten)

**i** Bei schweren Kommunikationsfehlern geht der Motion Controller standardmäßig in den NMT-Zustand *Pre-Operational*. Im Objekt 0x1029 kann ein anderes Verhalten eingestellt werden.

## CANopen-Protokollbeschreibung

### 3.8.1 Boot-Up

Der Motion Controller sendet unmittelbar nach der Initialisierungsphase eine Boot-up-Nachricht. Eine Boot-up-Nachricht signalisiert das Ende der Initialisierungsphase einer neu eingeschalteten Baugruppe. Eine Boot-up-Nachricht ist eine CAN-Nachricht mit einem Datenbyte (Byte 0 = 0x00) auf dem Identifier der Node-Guarding-Nachricht (0x700 + Node ID).

11-Bit-Identifizier	1 Byte Nutzdaten
---------------------	------------------

0x700 (1792d) + Node ID	0x00
----------------------------	------

## CANopen-Protokollbeschreibung

### 3.8.2 Überwachungsfunktionen

**i** Nur eine Überwachungsfunktion, Node Guarding oder Heartbeat, kann verwendet werden.

#### 3.8.2.1 Node Guarding

Das Node Guarding Objekt fragt den momentanen Zustand des Geräts ab. Dazu setzt der Master ein Remote-Frame mit einer Anforderung auf dem Guarding-Identifizier des zu überwachenden Knotens. Der zu überwachende Knoten antwortet mit der Guarding-Nachricht, die den aktuellen Status des Knotens und ein Toggle-Bit enthält.

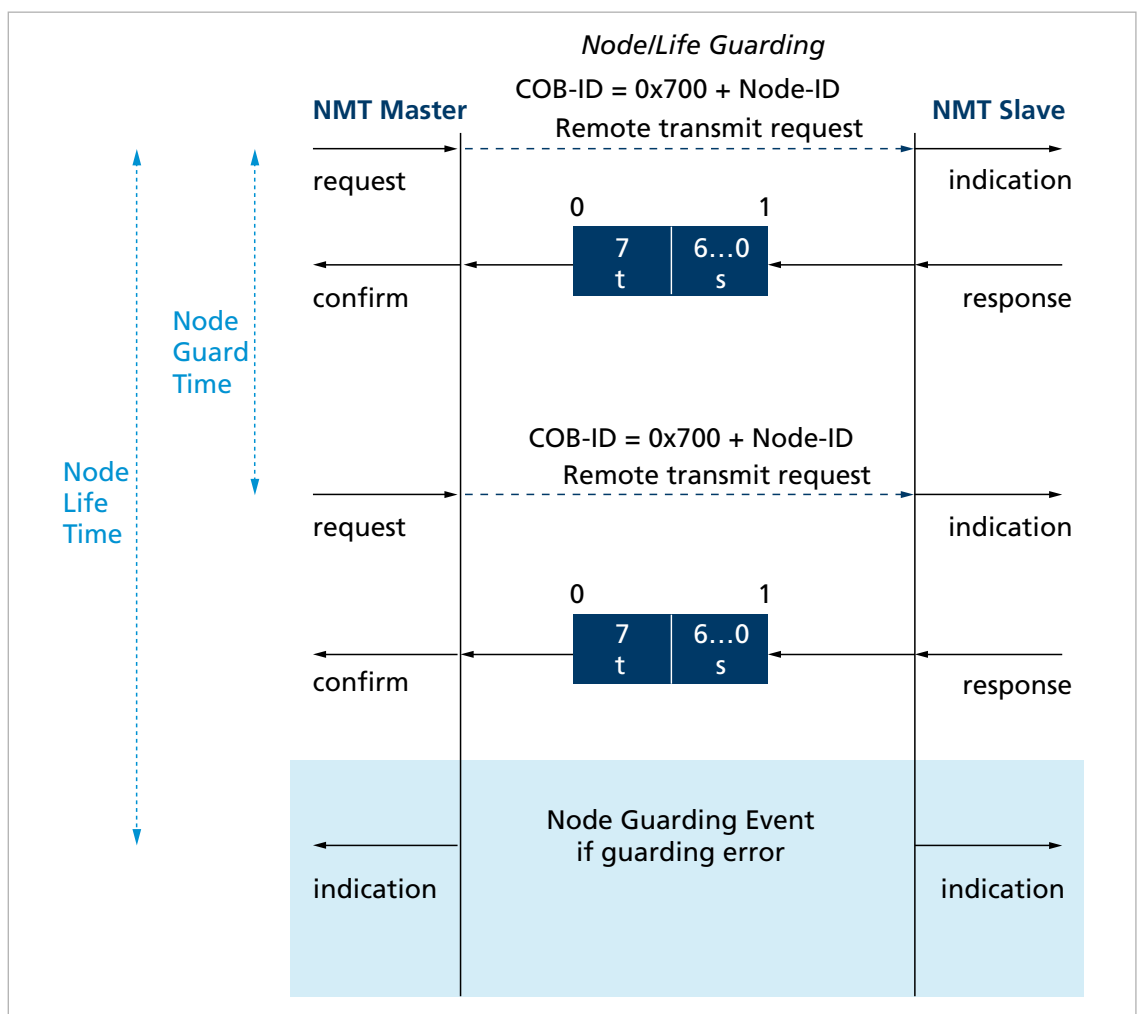


Abb. 5: Schaubild Node Guarding Protokoll

**t: Toggle Bit**

Anfänglich 0, wechselt in jedem Guarding-Telegramm seinen Wert

**s: Status**

$s = 0x04$  (4d): Stopped

$s = 0x05$  (5d): Operational

$s = 0x7F$  (127d): Pre-Operational

Wenn eine Node Life Time > 0 eingestellt ist (Objekte 0x100C und 0x100D) und innerhalb der angegebenen Life Time keine Node Guarding Abfrage des Masters eintrifft, wird ein Node Guarding Fehler gesetzt. Über das FAULHABER Fehlerregister (Objekt 0x2321) wird



## CANopen-Protokollbeschreibung

die Reaktion auf einen Node Guarding Fehler eingestellt (siehe Tab. 14). Standardmäßig wird die Emergency-Nachricht 0x8130 versendet.

### 3.8.2.2 Heartbeat

Der Motion Controller kann sowohl als Heartbeat Producer wie auch als Heartbeat Consumer eingestellt werden.

- **Heartbeat Producer:** Der Motion Controller setzt zyklisch eine Nachricht ab, die von einem oder mehreren Heartbeat-Consumern im Netzwerk empfangen wird.
- **Heartbeat Consumer:** Der Motion Controller reagiert mit dem im FAULHABER Fehlerregister (Objekt 0x2320) eingestellten Verhalten, wenn innerhalb der Heartbeat Consumer Time keine Heartbeat Nachricht des zu überwachenden Heartbeat Producer eintrifft (siehe Tab. 12).

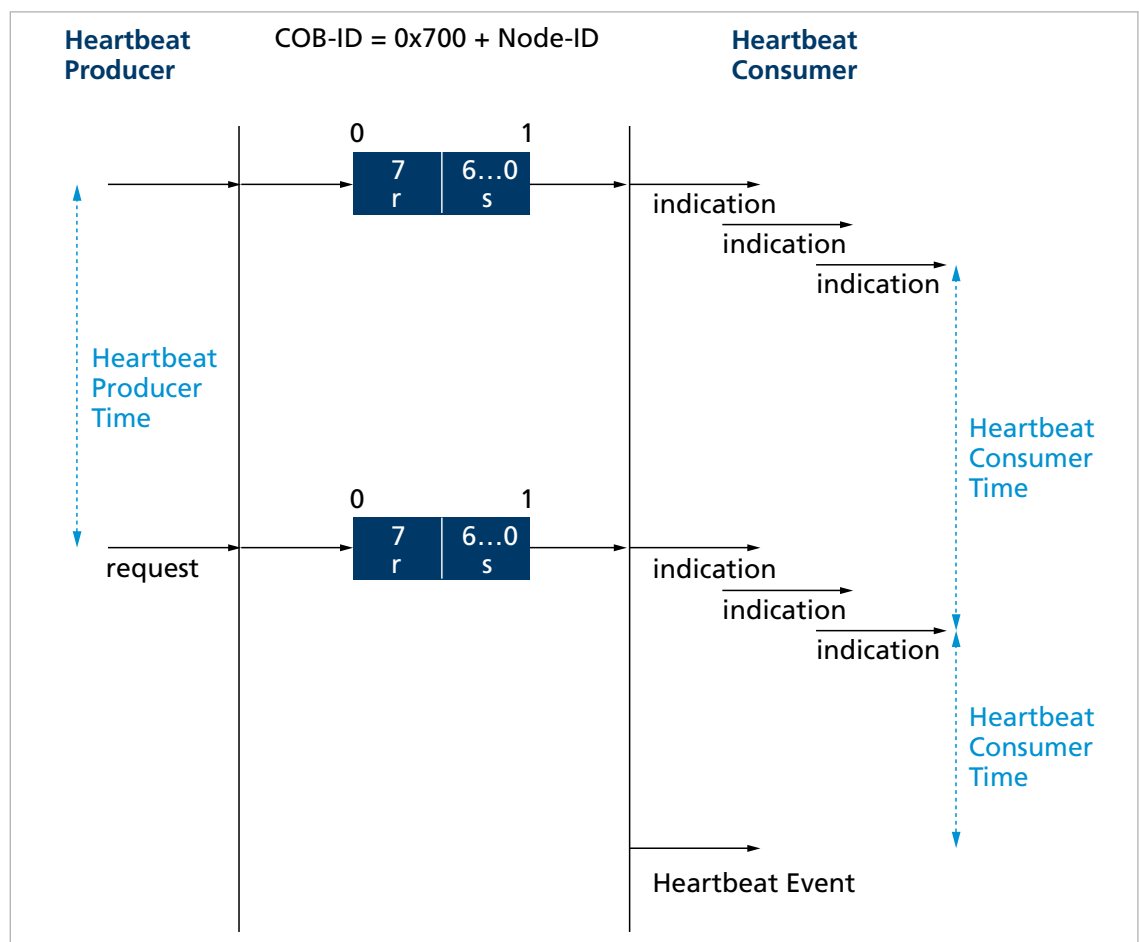


Abb. 6: Schaubild Heartbeat Protokoll

**r: Reserved**

Immer 0

**s: Status**

$s = 0x00$  (0d): Boot-Up

$s = 0x04$  (4d): Stopped

$s = 0x05$  (5d): Operational

$s = 0x7F$  (127d): Pre-Operational

## CANopen-Protokollbeschreibung

### 3.8.3 Einstellung der Überwachungsfunktionen

- Nur einer der beiden Überwachungsmechanismen (Node Guarding, Heartbeat) kann aktiviert sein.
- Bei einer Producer Heartbeat Time > 0 (Objekt 0x1017) arbeitet der Motion Controller als Heartbeat Producer. Der Motion Controller sendet im Zeitintervall der Producer Heartbeat Time eine Heartbeat Nachricht. Die Node Guarding Time wird auf 0 gesetzt (siehe Kap. 3.8.2.1, S. 32).
- Ist Heartbeat aktiviert, entspricht die Boot-up-Nachricht nach dem Einschalten der ersten Heartbeat Nachricht. Weitere Heartbeats folgen im Abstand der Producer Heartbeat Time.
- Ist zusätzlich zur Producer Heartbeat Time eine Heartbeat Consumer Time > 0 eingestellt (Objekt 0x1016.01), arbeitet der Motion Controller als Heartbeat-Consumer. Die Einstellungen des Heartbeat Producer sind unwirksam. Die Node ID des zu überwachenden Masters und die Heartbeat Consumer Time werden in das Objekt 0x1016 eingetragen.
- Die Heartbeat Consumer Time muss immer größer sein als die Producer Heartbeat Time des Masters.
- Falls der Motion Controller innerhalb der eingestellten Heartbeat Consumer Time keine Heartbeat Nachricht des Masters erhält, wird ein Heartbeat Event ausgelöst. Die Reaktion auf ein Heartbeat Event wird über die Error Mask des FAULHABER Fehlerregisters (Objekt 0x2321) eingestellt (siehe Tab. 12). Standardmäßig wird die Emergency-Nachricht 0x8130 versendet.
- Beim Versuch eine Node Guarding Zeit einzustellen, während der Heartbeat Producer aktiviert ist, wird der SDO-Fehler 0x08000020 (Zugriff nicht möglich) versendet.

### 3.9 Einträge im Objektverzeichnis

Das Objektverzeichnis verwaltet die Konfigurationsparameter. Das Objektverzeichnis ist in drei Bereiche unterteilt. Jedes Objekt kann über seinen Index und Subindex referenziert werden (SDO-Protokoll).

- Kommunikationsparameter (Index 0x1000 bis 0x1FFF, enthält Kommunikationsobjekte nach CiA 301, siehe Kap. 6.1, S. 96)
- Herstellerspezifischer Bereich (Index 0x2000 bis 0x5FFF, enthält herstellerspezifische Objekte, siehe Kap. 6.2, S. 104)
- Standardisierte Geräteprofile (0x6000 bis 0x9FFF, enthält die vom Motion Controller unterstützten Objekte, siehe Dokumentation der Antriebsfunktionen)

## CANopen-Protokollbeschreibung

### 3.10 Fehlerbehandlung

#### 3.10.1 CAN-Fehler

##### **CAN Overrun (object lost)**

Falls Nachrichten verloren gehen, versendet der Controller die Emergency-Nachricht 0x8110. Im Error-Register wird das Bit 4 (Communication Error) und im FAULHABER Fehlerregister das Bit 7 (CAN Overrun) gesetzt. Die Emergency-Nachricht wird zeitverzögert versendet. Der Fehler wird nicht durch die Emergency-Nachricht (0x000) zurückgenommen. Die entsprechenden Bits im Error-Register und im FAULHABER Fehlerregister werden nicht gelöscht.

##### **CAN in Error Passive Mode**

Falls das CAN-Modul des Antriebs im Zustand *Error-Passive* ist, wird die Emergency-Nachricht 0x8120 versendet. Im Error-Register wird das Bit 4 (Communication Error) und im FAULHABER Fehlerregister das Bit 6 (CAN in Error Passive Mode) gesetzt. Die Emergency-Nachricht (0x000) wird versendet und der Fehler wird zurückgenommen, wenn der Antrieb wieder in den Zustand *Error-Active* geht.

##### **Recovered from Bus Off**

Falls das CAN-Modul des Antriebs im Zustand *Bus Off* ist und eine gültige Nachricht empfängt, wird die Emergency-Nachricht 0x8140 versendet. Die Emergency-Nachricht meldet, dass der Zustand *Bus Off* verlassen wurde. Im Error-Register wird das Bit 4 (Communication Error) und im FAULHABER Fehlerregister das Bit 9 (Recovered from Bus Off) gesetzt. Der Fehler wird nicht zurückgenommen. Die entsprechenden Bits im Error-Register und im FAULHABER Fehlerregister werden nicht gelöscht.



Die Fehler CAN Overrun und Recovered from Bus-Off sind schwere Kommunikationsfehler. Die entsprechenden Bits im Error-Register und im FAULHABER Fehlerregister können nur mit einem Neustart des Motion Controllers zurückgesetzt werden. Weitere schwere Kommunikationsfehler sind:

- Node Guarding Timeouts
- Heartbeat Timeouts

## CANopen-Protokollbeschreibung

### 3.10.2 Gerätefehler

Tab. 12: FAULHABER Fehlerregister (0x2320)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2320	0x00	Fault Register	U16	ro	–	FAULHABER Fehlerregister

Das FAULHABER Fehlerregister enthält bitcodiert die zuletzt aufgetretenen Fehler. Die Fehler können durch Selektion der gewünschten Fehlerarten über das Objekt Error Mask (0x2321) maskiert werden.

Tab. 13: Fehlercodierung

Error-Bit	Fehlermeldung	Beschreibung
0x0001	Continuous Over Current	Eingestellte Dauerstrombegrenzung überschritten
0x0002	Deviation	Eingestellte maximal zulässige Drehzahlabweichung überschritten
0x0004	Over Voltage	Überspannung detektiert
0x0008	Over Temperature	Maximale Spulen- bzw. MOSFET-Temperatur überschritten
0x0010	Flash Memory Error	Speicherfehler
0x0040	CAN In Error Passive Mode	CAN-Controller im Error Passive Modus
0x0080	CAN Overrun (objects lost)	Überlauf des CAN-Empfangspuffers
0x0100	Life Guard Or Heart-beat Error	CAN-Überwachungsfehler
0x0200	Recovered From Bus Off	CAN-Bus-Fehler „Bus off“ verlassen
0x0800	Conversion Overflow	Rechenüberlauf
0x1000	Internal Software	Interner Softwarefehler
0x2000	PDO Length Exceeded	PDO-Länge zu groß, wird aber verarbeitet
0x4000	PDO not processed due to length error	PDO-Länge zu klein, kann nicht verarbeitet werden

Jeder dieser Fehler entspricht auch einem Emergency Error Code. (siehe Kap. 3.6, S. 26).

Die Error Mask beschreibt die Behandlung interner Fehler entsprechend der Fehlercodierung (siehe Tab. 13).

Tab. 14: Error Mask (0x2321)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2321	0x00	Number of Entries	U8	ro	6	Anzahl Objekteinträge
	0x01	Emergency Mask	U16	rw	0xFFFF	Fehler, für die eine Fehlermeldung verschickt werden
	0x02	Fault Mask	U16	rw	0x0000	Fehler, für die die Zustandsmaschine des Antriebs in den Zustand <i>Fault Reaction Active</i> geht
	0x03	Error Out Mask	U16	rw	0x0000	Fehler, für die der Fehler-Ausgangspin gesetzt wird



Beim Setzen der Fault Mask (Subindex 0x02) werden die entsprechenden Bits auch in die Emergency Mask (Subindex 0x01) übernommen.

## CANopen-Protokollbeschreibung

### Beispiele:

- Beim Setzen der Fault Mask (Subindex 0x02) von Objekt 0x2321 auf 0x0001 wird der Antrieb bei Überstrom ausgeschaltet und in den Fehlerzustand versetzt. Ein Wert von 0x0101 schaltet den Antrieb auch bei einem CAN Life Guard oder Heartbeat Fehler aus.
- Wenn der Subindex 0x03 von Objekt 0x2321 auf 0x00 gesetzt ist, zeigt der Fehlerausgang (Fault-Pin) keine Fehler an. Wenn der Subindex 0x03 von Objekt 0x2321 auf 0xFFFF gesetzt ist, zeigt der Fehlerausgang (Fault-Pin) alle Fehler (auch CAN-Fehler) an.

Über das Objekt Error Handling (0x2322) können zusätzliche Einstellungen zur Fehlerverarbeitung durchgeführt werden.

### Error Handling (0x2322)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2322	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Error Delay	U16	rw	200	Fehlerverzögerungszeit in 1/100 s
	0x02	Deviation	U16	rw	30 000	Zulässige Drehzahlabweichung in min <sup>-1</sup>

### Erläuterungen

Subindex	Name	Erläuterung
0x01	Error Delay	Fehlerverzögerungszeit, die angibt wie lange einer der folgenden Fehler anstehen muss, bis er gemeldet wird: <ul style="list-style-type: none"> <li>▪ Continuous Over Current</li> <li>▪ Deviation</li> <li>▪ Over Voltage</li> </ul>
0x02	Deviation	Größte zulässige betragsmäßige Abweichung der Istzahl von der Sollzahl. Eine Überschreitung wird nach Ablauf der Fehlerverzögerungszeit gemeldet.

# Funktionsbeschreibung

## 4 Funktionsbeschreibung

### 4.1 Antriebsdaten

Grundlegende Eigenschaften des Antriebssystem werden in den Objekten Motor Data (0x2350) und Encoder Data (0x2351) abgelegt.

#### Motordaten

Für die Modelle zur Motorüberwachung werden folgende Parameter benötigt:

- Drehzahlkonstante
- Anschlusswiderstand
- Polzahl, bei Bürstenlosmotoren
- Thermische Zeitkonstante

Bei integrierten Einheiten sind diese Werte bereits eingestellt. Für externe Steuerungen werden diese Werte durch Auswahl eines Motortyps im Motorassistenten des Motion Managers passend vorbelegt.

#### Motor Data (0x2350)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2350	0x00	Number of Entries	U8	ro	5	Anzahl Objekteinträge
	0x01	Speed Constant $K_N$	U16	rw	a)	Drehzahlkonstante
	0x02	Terminal Resistance RM	U16	rw	a)	Anschlusswiderstand
	0x03	Pole Number	U16	rw	2/4 a)	Polzahl bei BL-Motoren (nicht MCDC)
	0x05	Thermal Time Constant TW1	U16	rw	a)	Thermische Zeitkonstante 1

a) Abhängig von der Auslieferungskonfiguration des Motion Controllers

## Funktionsbeschreibung

### Encoder Data (0x2351)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2351	0x00	Number of Entries	U8	ro	3	Anzahl Objekteinträge
	0x01	Sensor Type	U8	rw	0	<ul style="list-style-type: none"> <li>▪ 0: Analog Hall (int. Encoder)</li> <li>▪ 1: Inkrementalencoder (ext.)</li> <li>▪ 10: Inkremental + Hall</li> <li>▪ 104: Absolutencoder AES-4096 (nicht MCDC)</li> </ul>
	0x02	Resolution External Encoder	U32	rw	2 048	4-Flanken-Auflösung eines extern angeschlossenen Inkrementalgebers
	0x03	Resolution Internal Encoder	U32	ro	3 000	Auflösung des internen Hallsensorgebers (nicht MCDC)

#### ■ Sensor Type:

Für Bürstenlosmotoren werden als Positionssysteme folgende Kombinationen unterstützt:

- Analoge Hallsensoren (3 000 Inkremente/Umdrehung, fest)
- Analoge Hallsensoren + Inkrementalgeber (Auflösung abhängig vom Inkrementalencoder)
- AES Geber (z. B. AES-4096)

Für DC-Motoren wird als Positionssystem ein Inkrementalencoder mit wählbarer Auflösung unterstützt.

#### ■ Resolution External Encoder:

Bei Verwendung eines externen Inkrementalgebers muss dessen Auflösung bei 4-Flankenauswertung (4-fache Impulszahl) angegeben werden.

#### ■ Resolution Internal Encoder:

Bei Verwendung der analogen Hallsensoren der Bürstenlosmotoren als Positionssystem werden fest 3 000 Impulse pro Umdrehung geliefert.

MCDC verwendet ausschließlich einen externen Encoder, daher steht hier die Sensor Type Umschaltung nicht zur Verfügung. Bei AES-Controllern ist die Auflösung durch den Sensor Type definiert, ein externer Encoder kann hier nicht verwendet werden.

## Funktionsbeschreibung

### 4.2 Device Control

FAULHABER Motion Control Systeme unterstützen Device Control gemäß CiA 402 und die Betriebsarten Profile Position Mode, Profile Velocity Mode, Cyclic Synchronous Position Mode und Homing Mode.

#### 4.2.1 Zustandsmaschine des Antriebs

FAULHABER Motion Controller durchlaufen während des Ein- und Ausschaltvorgangs eine Zustandsmaschine mit mehreren Schritten. Die Abfolge entspricht dem in der CiA 402 für CANopen Antriebe definierten Verhalten.

Gesteuert werden die Übergänge durch das Controlword (0x6040) des Antriebs.

Das Antriebsverhalten wird mit einer Zustandsmaschine abgebildet. Das Controlword steuert die Übergänge, das Statusword zeigt die Zustände an.

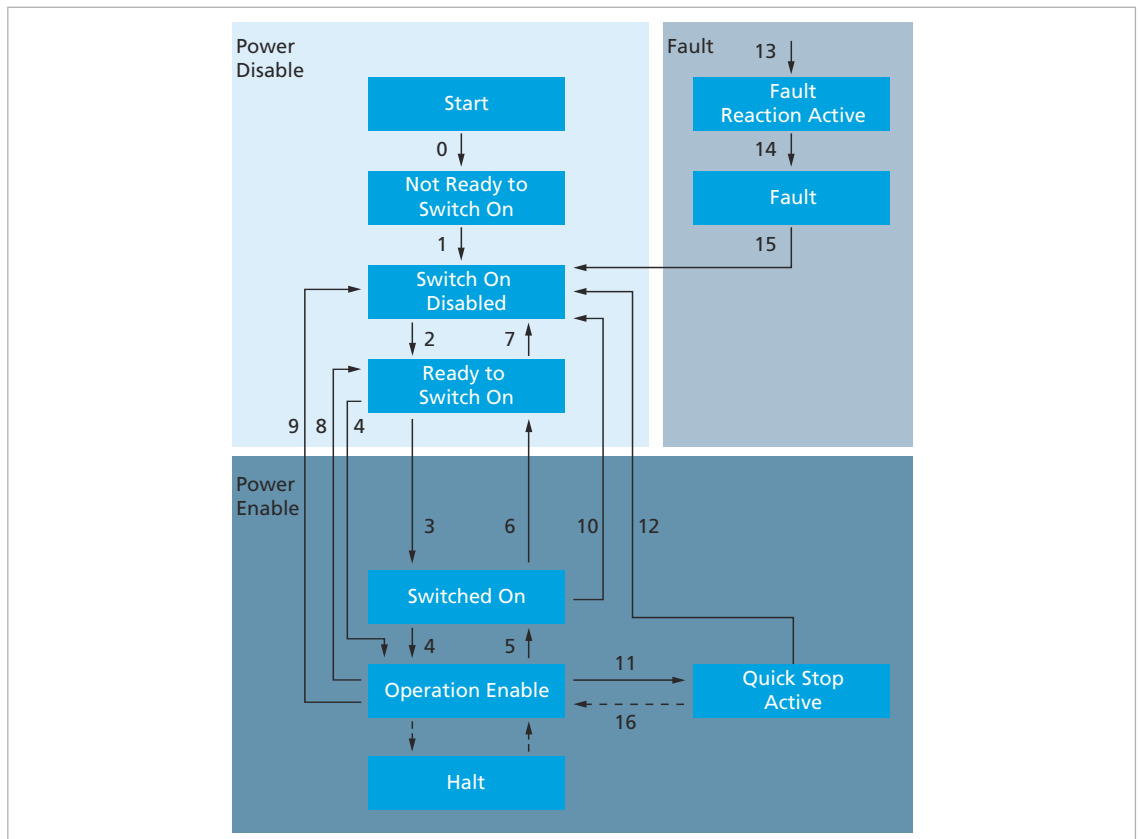


Abb. 7: Zustandsmaschine des Antriebs

Tab. 15: Kommandoübersicht

Kommando	Übergänge
Shut Down	2, 6, 8
Switch On	3
Disable Voltage	7, 9, 10, 12
Quick Stop	7, 10, 11
Disable Operation	5



## Funktionsbeschreibung

Kommando	Übergänge
Enable Operation	4, 16
Fault Reset	15

- Der Zustand *Not Ready to Switch On* wird automatisch durchlaufen. Der Motion Controller kann über das Objekt 0x2503 so konfiguriert werden, dass automatisch die Offsets der Strommessung neu abgeglichen werden.
- Nach dem Einschalten befindet sich der Antrieb im Zustand *Switch On Disabled*. Die Status LED beginnt grün zu blinken.
- Das Kommando **Shutdown** bringt den Antrieb in den Zustand *Ready to Switch On*. Über den Option Code im Objekt 0x605B kann vorgegeben werden, ob der Antrieb zunächst kontrolliert stillgesetzt werden soll.
- Über das Kommando **Switch On** wechselt der Motion Controller in den Zustand *Switched On*. Der Zustand *Switched On* kann automatisch durchlaufen werden, wenn im Zustand *Ready to Switch On* direkt das Kommando **Enable Operation** gegeben wird.
- Das Kommando **Enable Operation** bringt den Antrieb in den Zustand *Operation Enabled*. Der Übergang erfolgt nur, wenn die Versorgungsspannung im zulässigen Bereich liegt.
- Im Zustand *Operation Enabled* ist die Endstufe aktiviert. Die Status-LED leuchtet dauerhaft grün. Das Verhalten des Reglers hängt von der eingestellten Betriebsart ab.
- Das Kommando **Disable Operation** bringt den Antrieb zurück in den Zustand *Switched On*. Alle noch anstehenden Fahrbefehle werden dabei abgebrochen. Über den Option Code im Objekt 0x605C kann vorgegeben werden, ob der Antrieb zunächst kontrolliert stillgesetzt werden soll.
- Das Kommando **Disable Voltage** schaltet die Endstufe direkt ab. Der Motor wird nicht gebremst. Der Antrieb befindet sich danach im Zustand *Switch On Disabled*.
- Über das Kommando **Quick Stop** wechselt der Antrieb aus dem Zustand *Operation Enabled* in den Zustand *Quick Stop Active* und bremst den Antrieb mit der Quick Stop Rampe ab. Eventuell noch anstehende Fahrbefehle werden beim Wechsel in den Zustand *Quick Stop Active* verworfen. Der Zustand kann nur über das Kommando **Disable Voltage** verlassen werden. Anschließend kann der Antrieb normal über die Zustandsmaschine wieder in Betrieb genommen werden.
- Über das Halt-Bit im Controlword kann ein Antrieb während einer laufenden Bewegung angehalten werden. Der aktuelle und der folgende Fahrauftrag werden nicht verworfen, sondern lediglich unterbrochen, während das Halt-Bit gesetzt ist. Der Fahrauftrag wird wieder aufgenommen, sobald das Halt-Bit zurückgesetzt wird.
- In Reaktion auf einen erkannten Fehler kann der Antrieb aus jedem Zustand in den Fehlerzustand wechseln. Zunächst wird ein noch laufender Motor stillgesetzt. Danach wird die Endstufe abgeschaltet. Über die *Fault Mask* 0x2321.02 kann vorgegeben werden, für welche Fehlerarten dies geschehen soll. Der Benutzer muss über das Bit *Fault Reset* im Controlword den Fehler bestätigen. Erst dann kann er den Antrieb wieder einschalten.

## Funktionsbeschreibung

### 4.2.2 Controlword

Die Kommandos zur Ausführung einer Zustandsänderung werden durch Kombinationen der Bits 0–3 im Controlword definiert. Das Controlword befindet sich im Objektverzeichnis unter Index 0x6040.

#### Controlword

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6040	0x00	Controlword	U16	rw	–	Antriebssteuerung

Tab. 16: Überblick der Bits des Controlword und Kombinationsmöglichkeiten der Bits 0-3

Bit	Funktion	Kommandos für Device Control Zustandsmaschine						
		Shut Down	Switch On	Disable Voltage	Quick Stop	Disable Operation	Enable Operation	Fault Reset
0	Switch On	0	1	X	X	1	1	X
1	Enable Voltage	1	1	0	1	1	1	X
2	Quick Stop	1	1	X	0	1	1	X
3	Enable Operation	X	0	X	X	0	1	X
4	New Set-Point / Homing Operation Start							
5	Change Set Immediately							
6	Abs/Rel							
7	Fault Reset							0 → 1
8	Halt							
9	nicht verwendet							
10	nicht verwendet							
11	nicht verwendet							
12	nicht verwendet							
13	nicht verwendet							
14	nicht verwendet							
15	herstellerspezifisch: Life Bit							

1 = Bit gesetzt

0 = Bit gesetzt

0 → 1 = Steigende Flanke, Wechsel von 0 zu 1

X = Bit für dieses Kommando nicht verwendet (Zustand beliebig)

## Funktionsbeschreibung

Tab. 17: Bedeutung der Bits im Controlword

Bit	Funktion	Beschreibung
0	Switch On	0: Keine Spannung anliegend 1: Spannungszuführung wird aktiviert
1	Enable Voltage	0: Antrieb ausgeschaltet 1: Antrieb einschaltbereit
2	Quick Stop	0: Schnellstopp aktiviert 1: Schnellstopp deaktiviert
3	Enable Operation	0: Betrieb deaktiviert 1: Betrieb aktiviert
4	New Set-Point	0: Keine neue Zielposition vorgeben 1: Neue Zielposition vorgeben
5	Change Set Immediately	Nicht verwendet. Neue Positionieraufträge werden immer sofort gestartet.
6	Abs/Rel	0: Target Position ist ein absoluter Wert 1: Target Position ist ein relativer Wert
7	Fault Reset	0 → 1: Fehler zurücksetzen
8	Halt	0: Bewegung kann ausgeführt werden 1: Antrieb stoppen
15	Life Bit	Der Wert dieses Bits wird in Bit 15 des Statusword gespiegelt. So lässt sich feststellen, dass der Antrieb auf Änderungen im Controlword reagiert.

### 4.2.2.1 Beispiel: Enable Operation

Schrittfolge der Übergänge, um einen Antrieb in den Zustand *Operation Enabled* zu setzen.


- ✓ Antrieb befindet sich im Zustand *Switch On Disabled*.
- 1. Shut-Down-Befehl eingeben (Controlword = 0x00 06).
  - ↻ Antrieb wechselt in den Zustand *Ready to Switch On*.
- 2. Switch-On-Befehl eingeben (Controlword = 0x00 07).
  - ↻ Antrieb wechselt in den Zustand *Switched On*.
- 3. Enable-Operation-Befehl eingeben (Controlword = 0x00 0F).
  - ↻ Antrieb befindet sich im Zustand *Operation Enabled*. In diesem Zustand kann über entsprechende Objekte die eingestellte Betriebsart bedient werden.

## Funktionsbeschreibung

### 4.2.2.2 Beispiel: Reset des Fehlerzustands

Schrittfolge der Übergänge, um einen Antrieb aus dem Fehlerzustand zu holen.

1. Fault-Reset-Befehl eingeben (Controlword = 0x00 08).
  - ↪ Antrieb wechselt in den Zustand *Switch On Disabled*.
2. Shut-Down-Befehl eingeben (Controlword = 0x00 06).
  - ↪ Antrieb wechselt in den Zustand *Ready to Switch On*.
3. Enable-Operation-Befehl eingeben (Controlword = 0x00 0F).
  - ↪ Der Antrieb befindet sich im Zustand *Operation Enabled*. In diesem Zustand kann über entsprechende Objekte die eingestellte Betriebsart bedient werden.

 Der aktuelle Zustand der Zustandsmaschine des Antriebs (siehe Abb. 7) kann aus den Bits 0 bis 6 des Statusword abgelesen werden.

Nur die in aktuellen Zuständen definierten Übergänge können ausgeführt werden. Vor einem Zustandswechsel muss daher über die Auswertung des Statuswords geprüft werden, in welchem Zustand sich der Antrieb befindet.

### 4.2.3 Quick Stop

Der Antrieb wird mit der unter Quick Stop Deceleration (0x6085) angegebenen Bremsrampe heruntergefahren. Im Profile Position Mode hält er anschließend seine aktuelle Position.

Dieser Zustand kann nur durch das Kommando **Disable Voltage** beendet werden, indem zum Beispiel das Controlword auf 0 gesetzt wird.

## Funktionsbeschreibung

### 4.2.4 Statusword

Der aktuelle Zustand des Antriebs wird in den Bits 0–6 des Statusword abgebildet. Das Statusword befindet sich im Objektverzeichnis unter Index 0x6041.

#### Statusword

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6041	0x00	Statusword	U 16	ro	–	Statusanzeige

Tab. 18: Überblick der Bits des Statuswords und Kombinationsmöglichkeiten der Bits 0-6

Bit	Funktion	Zustand der Device Control Zustandsmaschine							
		Not Ready to Switch On	Switch On Disabled	Ready to Switch On	Switched on	Operation Enabled	Quick Stop Active	Fault Reaction Active	Fault
0	Ready to Switch On	0	0	1	1	1	1	1	0
1	Switched On	0	0	0	1	1	1	1	0
2	Operation Enabled	0	0	0	0	1	1	1	0
3	Fault	0	0	0	0	0	0	1	1
4	Voltage Enabled	X	X	X	X	X	X	X	X
5	Quick Stop	X	X	1	1	1	0	X	X
6	Switch On Disabled	0	1	0	0	0	0	0	0
7	Warning								
8	0								
9	Remote								
10	Target Reached								
11	Internal Limit Active								
12	Set-Point Acknowledge/Speed/Homing Attained								
13	Deviation Error								
14	Nicht verwendet								
15	herstellerspezifisch: Life Bit								

1 = Bit gesetzt

0 = Bit gesetzt


X = Bit für dieses Kommando nicht verwendet (Zustand beliebig)

## Funktionsbeschreibung

Tab. 19: Bedeutung der Bits im Statusword

Bit	Funktion	Beschreibung
0	Ready to Switch On	0: Nicht bereit zum Einschalten 1: Bereit zum Einschalten
1	Switched On	0: Keine Spannung anliegend 1: Antrieb befindet sich im Zustand <i>Switched On</i>
2	Operation Enabled	0: Betrieb deaktiviert 1: Betrieb aktiviert
3	Fault	0: Kein Fehler vorhanden 1: Fehler vorhanden
4	Voltage Enabled	0: Spannungsversorgung deaktiviert 1: Spannungsversorgung aktiviert
5	Quick Stop	0: Schnellstopp deaktiviert 1: Schnellstopp aktiviert
6	Switch On Disabled	0: Einschalten möglich 1: Einschalten nicht möglich
7	Warning	Nicht verwendet
8	0	Nicht verwendet
9	Remote	Nicht verwendet
10	Target Reached	0: Zielposition bzw. Soll Drehzahl noch nicht erreicht 1: Target Position bzw. Target Velocity erreicht Wird gesetzt, wenn der Antrieb im Profile Position Mode seine Sollposition erreicht hat oder im Profile Velocity Mode seine Soll Drehzahl erreicht hat. Die Vorgabe eines neuen Sollwerts löscht das Bit.
11	Internal Limit Active	Nicht verwendet
12	Set-Point Acknowledge <sup>a)</sup>  Speed  Homing Attained	0: Noch keine neue Sollposition übernommen (Profile Position Mode) 1: Neue Sollposition übernommen  0: Drehzahl ungleich 0 (Profile Velocity Mode) 1: Drehzahl gleich 0  0: Homing-Position noch nicht erkannt 1: Homing-Position erkannt
13	Deviation Error	0: Kein Fehler 1: Fehler
15	Life Bit	Der Wert dieses Bits wird aus Bit 15 des Controlwords gesetzt. So lässt sich feststellen, dass der Antrieb auf Änderungen im Controlword reagiert.

a) Wird nach Erhalt eines neuen Positionierbefehls (Controlword mit New Set Point) gesetzt. Wird zurückgesetzt, wenn New Set Point im Controlword zurückgesetzt wurde (Handshake für Positionierbefehl). Im Profile Velocity Mode wird das Bit bei Drehzahl 0 gesetzt und bei Drehzahl ungleich 0 zurückgesetzt.

 Im Zustand *Fault reaction active* wird der Antrieb mit der in Objekt 0x6084 eingestellten Bremsrampe gestoppt. Anschließend wird im Zustand *Fault* versucht die Drehzahl Null zu halten.

## Funktionsbeschreibung

### 4.2.5 Betriebsart wählen

Über den Parameter Modes of Operation wird das aktive Antriebsprofil gewählt. Über den Eintrag Modes of Operation Display kann die aktuelle Betriebsart ausgelesen werden.

**i** Nicht immer entspricht die aktive Betriebsart (0x6061) der eingestellten Betriebsart (0x6060). Die Betriebsarten PP und PV werden erst gestartet, wenn der erste auf die Wahl der Betriebsart folgende Sollwert eingeht. Die Betriebsart Homing startet erst an der steigenden Flanke von Bit 4 im Controlword, nachdem die Betriebsart über 0x6060.00 = 6 gewählt wurde.

#### Modes of Operation (0x6060)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6060	0x00	Modes of Operation	I8	rw	1	Auswahl der Betriebsart 0: Regler nicht aktiviert 1: Profile Position (Positionsregelung) 3: Profile Velocity (Drehzahlregelung) 6: Homing (Referenzfahrt) 8: Cyclic Synchronous Position (Positionsregelung)

#### Modes of Operation Display (0x6061)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6061	0x00	Modes of Operation Display	I8	ro	1	Anzeige der ausgewählten Betriebsart

Die Bedeutung der Rückgabewerte entspricht den Werten des Objekts 0x6060.

## Funktionsbeschreibung

### 4.3 Factor Group

Über die Objekte der Factor Group können die internen Positionswerte in benutzerdefinierte Einheiten umgerechnet werden. Interne Positionswerte sind in Inkrementen angegeben und abhängig von der Auflösung des verwendeten Encoders. Benutzerdefinierte Einheiten sind unabhängig von der Encoderauflösung und von angebauten Linear-Untersetzungen.

Aktuelle Position in benutzerdefinierten Einheiten:

$$\text{Position Actual Value} = \frac{\text{Position Actual Internal Value} \cdot \text{Feed Constant}}{\text{Position Encoder Resolution} \cdot \text{Gear Ratio}}$$

Getriebeverhältnis:

$$\text{Gear Ratio} = \frac{\text{Motor Revolutions}}{\text{Shaft Revolutions}}$$

Vorschubeinheiten pro Umdrehungen am Motor und am Abtrieb:

$$\text{Feed Constant} = \frac{\text{Feed}}{\text{Shaft Revolutions}}$$

$$\text{Feed Constant} = \frac{\text{Feed}}{\text{Shaft Movement}}$$

Encoder-Auflösung:

$$\text{Position Encoder Resolution} = \frac{\text{Encoder Increments}}{\text{Motor Revolutions}}$$

In Abb. 8 ist die Umrechnung von Benutzereinheiten in interne Einheiten über die Parameter des Objekts Position Factor (0x6093) angegeben. Position Factor zeigt dabei nur einen Zwischenwert an, der aus den Parametern der Objekte Position Encoder Resolution (0x608F), Gear Ratio (0x6091) und Feed Constant (0x6092) berechnet wird. Position Encoder Resolution zeigt dabei ebenfalls nur einen Zwischenwert an, der je nach ausgewähltem Sensortyp über das Objekt Encoder Data (0x2351) die jeweilige Encoder-Auflösung enthält.



## Funktionsbeschreibung

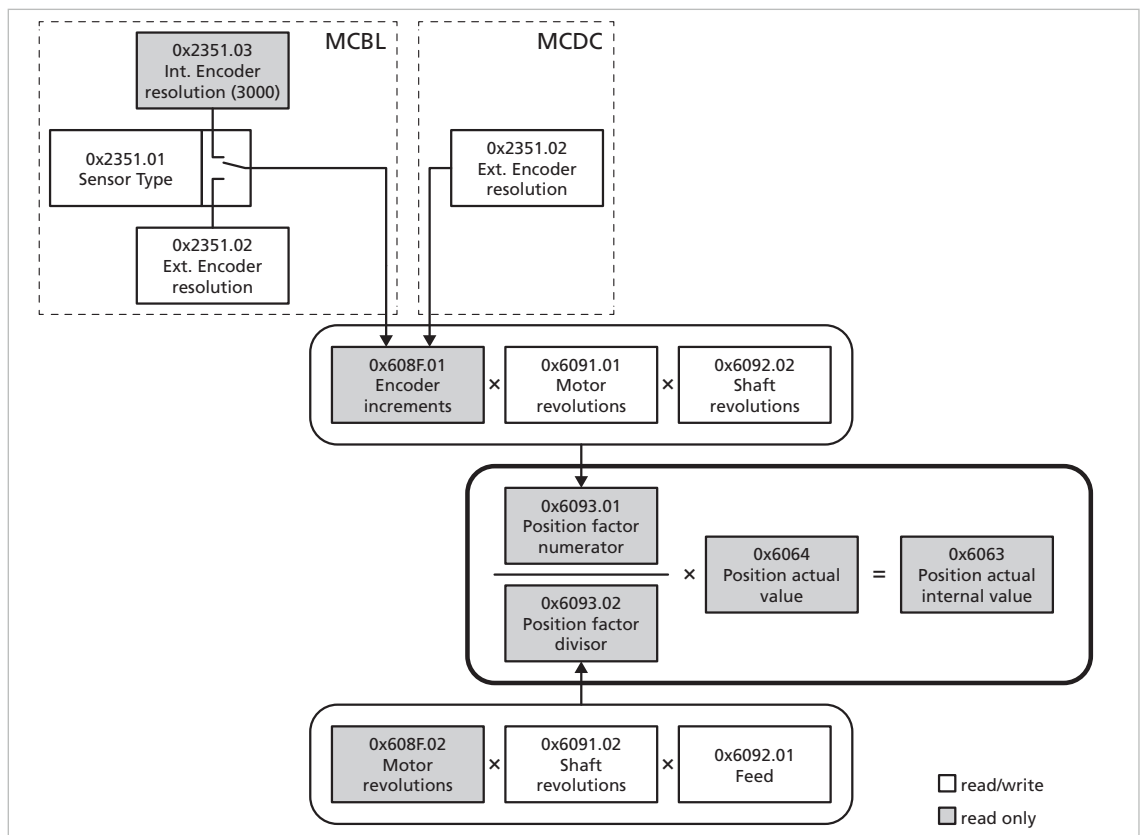


Abb. 8: Umrechnung von Benutzereinheiten in interne Einheiten

Bei der Einstellung der Umrechnungsfaktoren muss folgendes beachtet werden:

- Die resultierende Auflösung muss groß genug sein. Die maximale Auflösung mit Benutzereinheiten erhält man bei Gear Ratio = 1 und Feed Constant = Encoder Resolution.
- Der Zähler des Position Factor muss immer kleiner als  $2 \times 10^9$  sein.

Da aus den einzelnen Umrechnungsfaktoren Zähler und Nenner des eigentlichen Position Factor berechnet wird, kann es bei der Übertragung zu einem Zahlenüberlauf kommen, der durch den SDO-Fehler 0x06040047 gemeldet wird. Dieser Fehler kann auch beim Umschalten des Sensortyps oder beim Ändern der Encoderauflösung auftreten.

Wenn bei Folgeberechnungen ein Überlauf auftritt, wird das Emergency-Telegramm 0xFF01 verschickt und im FAULHABER Fehlerregister Bit 11 (Conversion overflow) gesetzt. Gibt es nach Korrektur der Faktoren keinen Umrechnungsfehler mehr, wird der Fehler gelöscht und das Emergency-Telegramm 0x0000 verschickt.

**i** Der Motion Controller verwaltet seine Positionsparameter grundsätzlich in internen Einheiten. Erst beim Beschreiben oder Auslesen werden diese mit Hilfe des Position Factor umgerechnet.

- Die Factor Group vor der allerersten Parametrisierung einstellen und dann möglichst nicht mehr verändern.

## Funktionsbeschreibung

### Position Encoder Resolution (0x608F)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x608F	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Encoder Increments	U32	ro	3 000/ 2 048 <sup>a)</sup>	Encoderinkremente
	0x02	Motor Revolutions	U32	ro	1	Anzahl der Motorumdrehungen bei der in Subindex 0x01 genannten Impulszahl

a) BL-Controller/MCDC

Der Wert der Encoderauflösung wird aus den Einstellungen in Objekt Encoder Data (0x2351) übernommen und kann hier nicht verändert werden.

### Gear Ratio (0x6091)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6091	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Motor Revolutions	U32	rw	1	Anzahl Motorumdrehungen
	0x02	Shaft Revolutions	U32	rw	1	Anzahl Umdrehungen der Abtriebswelle

### Feed Constant (0x6092)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6092	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Feed	U32	rw	3 000 / 2 048 <sup>a)</sup>	Vorschub in Benutzereinheiten
	0x02	Shaft Revolutions	U32	rw	1	Anzahl Umdrehungen der Abtriebswelle

a) BL-Controller/MCDC

Bei Auslieferung sind die Benutzereinheiten auf die Standard-Encoderauflösung eingestellt, d. h. Positionierungen erfolgen in internen Einheiten:

- Bürstenlose Controller: 3 000 Inkremente pro Umdrehung
- MCDC: 2 048 Inkremente pro Umdrehung
- MCBL-AES: 4 096 Inkremente pro Umdrehung

**i** Bei der Einstellung einer anderen Encoderauflösung unter Beibehaltung der maximalen internen Auflösung muss Feed Constant auf dieselbe Auflösung eingestellt werden. Andernfalls bleiben die voreingestellten Benutzereinheiten erhalten.

### Beispiel: Benutzereinheiten in Winkelgrad

Um eine 1°-Auflösung zu erhalten, muss Feed Constant auf 360 pro Umdrehung eingestellt werden.

### Position Factor (0x6093)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6093	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Numerator	U32	ro	1	Zähler des Positionsfaktors
	0x02	Divisor	U32	ro	1	Nenner des Positionsfaktors

Der Positionsfaktor wird berechnet aus Position Encoder Resolution (0x608F), Gear Ratio (0x6091) und Feed Constant (0x6092) und kann hier nicht verändert werden.

## Funktionsbeschreibung

### Polarity (0x607E)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x607E	0x00	Polarity	U8	rw	0	Bit-codiert

Über dieses Objekt kann die Drehrichtung global geändert werden, d. h. Vorgabe- und Istwerte für Position bzw. Drehzahl werden mit  $-1$  multipliziert:

- Bit 7 = 1: negative Drehrichtung im Positionierbetrieb
- Bit 6 = 1: negative Drehrichtung im Drehzahlbetrieb

**i** Die Änderung des Position Factor und der Polarity hat Einfluss auf die eingestellten Werte für Position Range Limit (0x607B), Software Position Limit (0x607D), Position Window (0x6067) und Homing Offset (0x607C), die sich entsprechend mit ändern.

- Diese Werte anschließend kontrollieren und gegebenenfalls anpassen.
- Bei negativer Polarity beachten, dass das Vorzeichen der Limits sich ebenfalls ändert und dadurch das Minimum Limit größer als das Maximum Limit wird.

Funktionen, die das komplette Objektverzeichnis einlesen (z. B. die FAULHABER Motion Manager Funktion **CANopen Konfigurationsdatei laden**), sollten zweimal aufeinanderfolgend ausgeführt werden, um sicherzustellen, dass Position Factor, Minimum Limit und Maximum Limit richtig interpretiert werden.

## Funktionsbeschreibung

### 4.4 Profile Position Mode

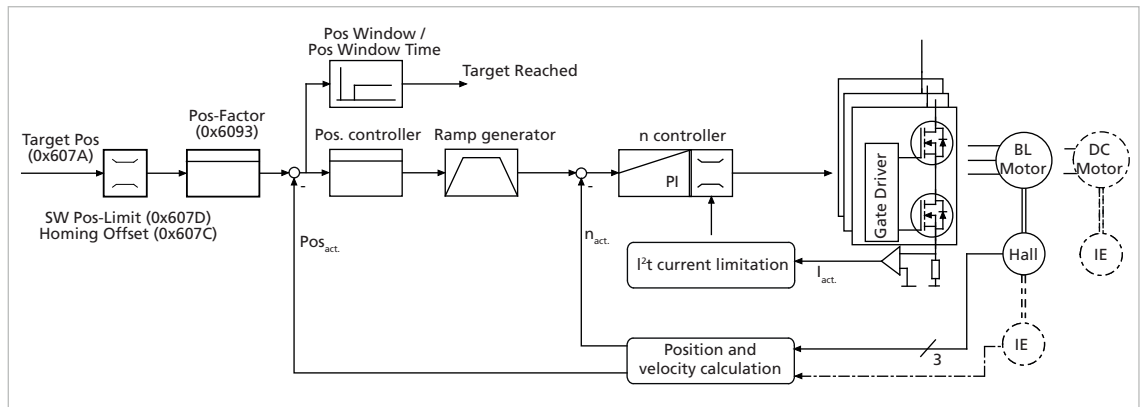


Abb. 9: Reglerstruktur bei Positionsregelung im Profile Position Mode

#### 4.4.1 Betriebsart im Überblick

Im Profile Position Mode (PP) positioniert der Antrieb auf die übergebene Zielposition.

Voraussetzungen für den Betrieb des Antriebs im PP Mode:

- Die Betriebsart Profile Position Mode ist im Parameter Modes of Operation (0x6060) eingestellt.
- Der Antrieb befindet sich im NMT-Zustand *Operation Enabled*.
- Drehzahlregler und Positionsregler sind korrekt eingestellt.

Nach dem Einschalten muss in der Regel eine Referenzfahrt über den Homing Mode ausgeführt werden, um den Positionswert am Homing-Endschalter zu nullen (siehe Kap. 4.5, S. 57).

Ein Positionssollwert wird über das Objekt Target Position (0x607A) vorgegeben. Der Positioniervorgang wird durch einen Wechsel von 0 auf 1 des Bit 4 (New Set Point) im Controlword gestartet. Über Bit 6 im Controlword kann zusätzlich vorgegeben werden, ob der Sollwert absolut oder relativ interpretiert werden soll.

Für den Bewegungsbereich können zusätzlich zum Sollwert über das Objekt Software Position Limit (0x607D) Bereichsgrenzen vorgegeben werden. Diese Bereichsgrenzen sind standardmäßig aktiviert, können aber über das Objekt General Settings (0x2338) deaktiviert werden.

Die eingestellten Maximalwerte für Beschleunigung, Bremsrampe und Geschwindigkeit werden zusätzlich berücksichtigt.

#### 4.4.2 Benachrichtigung der übergeordneten Steuerung

Das Erreichen der Zielposition wird über das Bit 10 (Target Reached) im Statusword des Antriebs signalisiert. Falls der Übertragungstyp für das jeweilige PDO auf 255 eingestellt ist, wird das PDO asynchron, getriggert durch den Zustandswechsel, übertragen.

## Funktionsbeschreibung

### 4.4.3 Grundeinstellungen

Für den Positionsregler können über das Objekt Position Control Parameter Set (0x2332) die Proportionalverstärkung und ein differentieller Anteil eingestellt werden.

Relativ zur Referenzposition können Grenzen des Positionierbereichs über das Objekt Software Position Limit (0x607D) vorgegeben werden.

Über das Objekt Position Window (0x6067) kann ein Fenster um die Zielposition definiert werden.

Die Zielposition wird über Bit 10 (Target Reached) im Statusword als erreicht signalisiert, wenn die Istposition mindestens für die im Objekt Position Window Time (0x6068) angegebene Zeit im Position Window verweilt.

#### Software Position Limit (0x607D)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x607D	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Min Position Limit	I32	rw	$-1,8 \cdot 10^9$	Untere Grenze des Positionsbereichs
	0x02	Max Position Limit	I32	rw	$1,8 \cdot 10^9$	Obere Grenze des Positionsbereichs

Die Positionierbereichsgrenzen werden in benutzerdefinierten Einheiten vorgegeben und über die Objekte der Factor Group in die interne Darstellung umgerechnet.

#### Position Control Parameter Set (0x2332)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2332	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Proportional Term PP	U16	rw	a)	Proportionalverstärkung des Positionsreglers
	0x02	Derivative Term PD	U16	rw	a)	Differenzialanteil des Positionsreglers

a) Abhängig von der Konfiguration des Motion Controllers

#### Position Window (0x6067)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6067	0x00	Position Window	U32	rw	32	Korridor um die Zielposition in benutzerdefinierten Einheiten

Symmetrischer Bereich um die Sollposition, der für die Meldung „Target Reached“ verwendet wird. Die Vorgabe erfolgt in benutzerdefinierten Einheiten, entsprechend des angegebenen Position Factor.

#### Position Window Time (0x6068)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6068	0x00	Position Window Time	U16	rw	48	Mindestverweildauer im Korridor, bis die Zielposition in der Betriebsart PP als erreicht signalisiert wird.

Wenn der Antrieb mindestens die hier eingestellte Zeit in Millisekunden im Bereich des Position Window verweilt, wird das Bit 10 im Statusword (Target Reached) gesetzt.

## Funktionsbeschreibung

### 4.4.4 Abfrage aktueller Werte / Position Control Function

Die letzte Sollposition kann über das Objekt Position Demand Value (0x6062) in benutzerdefinierten Einheiten zurück gelesen werden.

Die aktuelle Position kann über das Objekt Position Actual Internal Value (0x6063) in internen Einheiten bzw. über Objekt Position Actual Value (0x6064) in benutzerdefinierten Einheiten zurück gelesen werden (siehe Kap. 6.3, S. 110).

### 4.4.5 Zusätzliche Einstellungen

#### 4.4.5.1 Inkrementalencoder als Positionsgeber

Für BL-Motoren wird die Position standardmäßig über die analogen Hallsensoren mit einer Auflösung von 3 000 Inkrementen pro Umdrehung ausgewertet.

Alternativ kann auch bei BL-Motoren im Profile Position Mode mit einem Inkrementalgeber als Positionsgeber gearbeitet werden. Der Sensortyp und die Auflösung des externen Encoders werden über das Objekt 0x2351 eingestellt. Bei DC-Motoren wird die Position immer über einen Inkrementalencoder erfasst.

#### 4.4.5.2 Rampengenerator

Der Ausgang des Positionsreglers wird zusätzlich über einen Rampengenerator auf die zulässigen Beschleunigungswerte und Bremswerte und auf die maximale Geschwindigkeit begrenzt.

Unterstützt wird ausschließlich ein Trapezprofil mit linearen Geschwindigkeitsrampen.

#### Profile Velocity (0x6081) und Max Profile Velocity (0x607F)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6081	0x00	Profile Velocity	U32	rw	a)	Maximaldrehzahl in $\text{min}^{-1}$
0x607F	0x00	Max Profile Velocity	U32	rw	b)	Maximal erlaubte Drehzahl [ $\text{min}^{-1}$ ]

a) Abhängig von der Auslieferungskonfiguration des Motion Controllers

b) Abhängig von der Auslieferungskonfiguration des Motion Controllers

Maximale Drehzahl und maximal erlaubte Drehzahl während einer Positionierung.

#### Profile Acceleration (0x6083) und Profile Deceleration (0x6084)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6083	0x00	Profile Acceleration	U32	rw	30 000	Maximale Beschleunigung [ $1/\text{s}^2$ ]
0x6084	0x00	Profile Deceleration	U32	rw	30 000	Maximale Bremsrate [ $1/\text{s}^2$ ]

#### Quick Stop Deceleration (0x6085)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6085	0x00	Quick Stop Deceleration	U32	rw	30 000	Bremsrampenwert bei Quick Stop [ $1/\text{s}^2$ ]

#### 4.4.5.3 Drehzahlregler / Strombegrenzung

Für den unterlagerten Drehzahlregler können die Reglerparameter ebenfalls angepasst werden (Objekt 0x2331). Zusätzlich kann der Antrieb über die Spitzen- und Dauerstrombegrenzungswerte (Objekt 0x2333) vor Überlastung geschützt werden (siehe Kap. 4.6, S. 64).

## Funktionsbeschreibung

### 4.4.5.4 Befehle zur Bewegungssteuerung

Ein Positionssollwert wird über das Objekt Target Position (0x607A) vorgegeben. Der Positioniervorgang wird durch einen Wechsel von 0 auf 1 des Bit 4 (New Set-Point) im Controlword gestartet.

Über Bit 6 im Controlword kann zusätzlich vorgegeben werden, ob der Sollwert absolut oder relativ interpretiert werden soll.

#### Target Position (0x607A)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x607A	0x00	Target Position	I32	rw	–	Positionssollwert in benutzerdefinierten Einheiten

Die Sollposition wird in benutzerdefinierten Einheiten vorgegeben und über die Objekte der Factor Group in die interne Darstellung umgerechnet.

Die Übernahme einer neuen Sollposition wird vom Antrieb über das Statusword mit gesetztem Bit 12 (Set-Point Acknowledge) quittiert. Das Erreichen der Sollposition meldet der Antrieb über das Statusword mit gesetztem Bit 10 (Target Reached). Target Reached bleibt solange gesetzt, bis eine neue Positionierung gestartet oder die Endstufe abgeschaltet wird.

Wenn während einer Positionierung ein neuer Sollwert vorgegeben wird (New Set-Point), wird dieser sofort übernommen und die neue Sollposition wird angefahren. Auf diese Weise können kontinuierliche Bewegungsprofile abgefahren werden, ohne dass der Antrieb zwischendurch auf Drehzahl 0 abgebremst werden muss.

Vor der erneuten Ausführung einer Positionierung muss Bit 4 im Controlword immer zurückgesetzt werden, was vom Antrieb mit zurückgesetztem Bit 12 im Statusword quittiert wird.

Ein Statusword mit zurückgesetztem Bit 12 (Set-Point Acknowledge = 0) signalisiert die Bereitschaft zur Entgegennahme eines neuen Positionierauftrags.

## Funktionsbeschreibung

### Ablauf bei Positionierungen

- ✓ NMT-Zustand *Operational*
  - ✓ Antriebszustand *Operation Enabled*
  - ✓ Modes of Operation (0x6060) sind auf Profile Position Mode (1) gesetzt
  - 1. Target Position (0x607A) auf den gewünschten Wert für die Sollposition setzen.
  - 2. Im Controlword Bit 4 (New Set-Point) auf 1 setzen und mit Bit 6 (Abs/Rel) absolute oder relative Positionierung einstellen.
    - ✚ Der Antrieb antwortet mit Bit 12 (Set-Point Acknowledge) im Statusword gesetzt und startet die Positionierung.
  - 3. Controlword Bit 4 wieder zurücksetzen.
    - ✚ Bit 12 im Statusword wird zurückgesetzt.
- Sobald Bit 12 im Statusword zurückgesetzt ist, kann eine neue Positionierung durch Zustandsänderung des Bits 4 (New Set-Point) im Controlword von 0 auf 1 gestartet werden. Positionieraufträge werden immer sofort ausgeführt (Change Set Immediately).
- Das Erreichen der Sollposition meldet der Antrieb über das Statusword mit gesetztem Bit 10 (Target Reached).

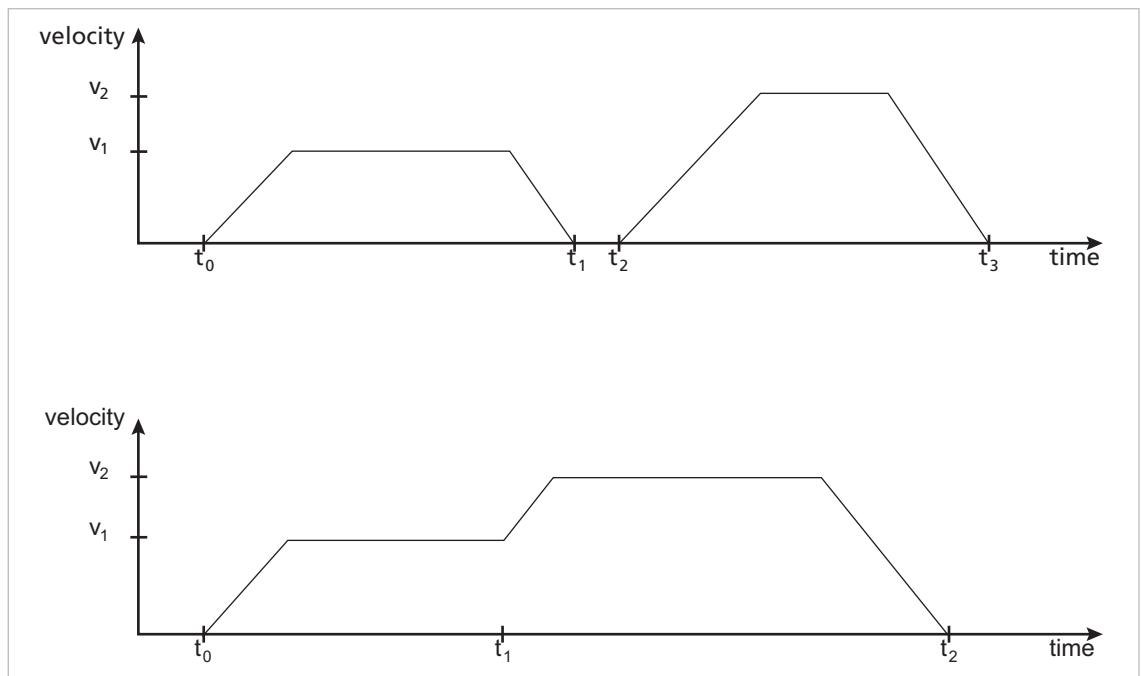


Abb. 10: Ablauf bei Positionierungen

**i** Bei relativen Positionierungen wird die neue Sollposition zur letzten Sollposition addiert.



## Funktionsbeschreibung

### 4.5 Homing Mode

Nach dem Einschalten muss in der Regel eine Referenzfahrt (Homing) ausgeführt werden, um den Positionswert am Homing-Endschalter zu nullen. Welche Eingänge als Homing-Endschalter verwendet werden sollen, kann über das Objekt 0x2310 eingestellt werden (siehe Kap. 4.8.1, S. 69).

#### Homing Offset (0x607C)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x607C	0x00	Min Range Limit	l32	rw	0	Offset der Null-Position gegenüber der Position des Referenzschalters

#### Homing Method (0x6098)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6098	0x00	Homing Method	l8	rw	20	Homing-Methode

#### 4.5.1 Homing-Methoden

Unterstützt werden folgende an die CiA 402 angelehnten Homing-Methoden:

- 1 bis 14 (Homing mit Indeximpuls, falls vorhanden)
- 17 bis 30 (Homing ohne Indeximpuls)
- 33, 34 (Homing am Indeximpuls, falls vorhanden)
- 35 (Homing an der aktuellen Position)

**i** Endschalter begrenzen den Bewegungsbereich (Negative/Positive Limit Switch), können aber auch gleichzeitig als Referenzschalter für die Null-Position verwendet werden. Ein Homing-Schalter ist ein eigener Referenzschalter für die Null-Position.

#### Methode 1 und 17

Homing am unteren Endschalter (Negative Limit Switch):

- Wenn der Endschalter inaktiv ist, fährt der Antrieb zunächst in Richtung des unteren Endschalters, bis dessen positive Flanke erkannt wurde. Ist der Endschalter aktiv, fährt der Antrieb nach oben aus dem Endschalter heraus, bis die negative Flanke erkannt wurde. Bei Methode 1 wird noch zusätzlich auf den nächsten Indeximpuls weiter gefahren, an dem die Home-Position gesetzt wird.

## Funktionsbeschreibung

### Methode 2 und 18

Homing am oberen Endschalter (Positive Limit Switch):

- Wenn der Endschalter inaktiv ist, fährt der Antrieb zunächst in Richtung des oberen Endschalters, bis dessen positive Flanke erkannt wurde. Ist der Endschalter aktiv, fährt der Antrieb nach unten aus dem Endschalter heraus, bis die negative Flanke erkannt wurde. Bei Methode 2 wird zusätzlich auf den nächsten Indeximpuls weiter gefahren, an dem die Home-Position gesetzt wird.

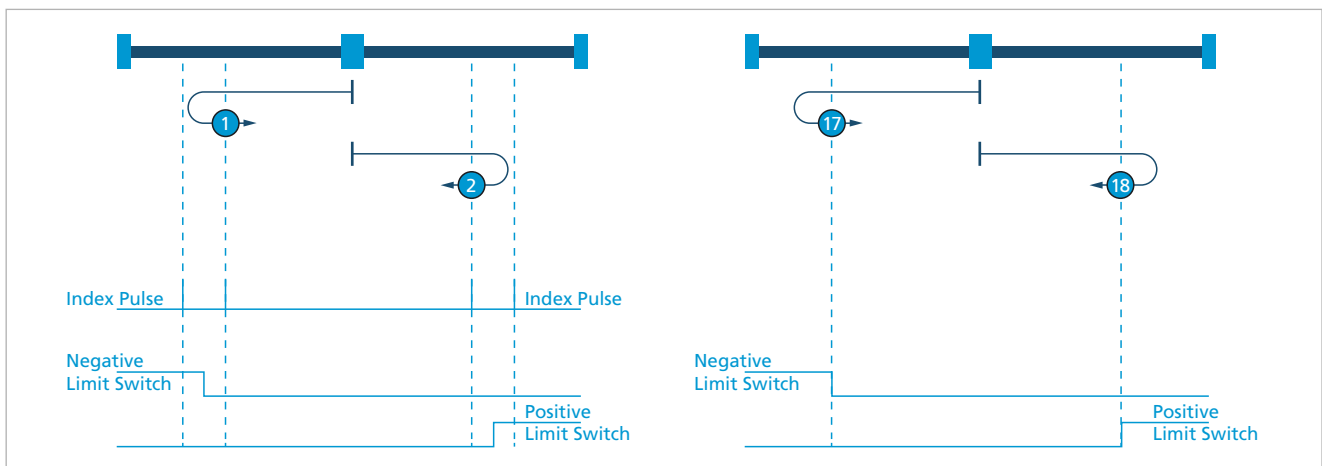


Abb. 11: Homing-Methode 1, 2, 17 und 18

### Methode 3, 4 und 19, 20

Homing an einem positiven Homing-Schalter (Positive Home Switch)

Je nachdem welchen Zustand der Homing-Schalter hat, fährt der Antrieb in eine Richtung, bis zur fallenden (3, 19) oder steigenden (4, 20) Flanke. Dabei gibt es in Richtung des oberen Endschalters nur eine steigende Flanke des Homing-Schalters.

Die Homingposition liegt an der Stelle, an der sich der Zustand des Home Switch ändert.

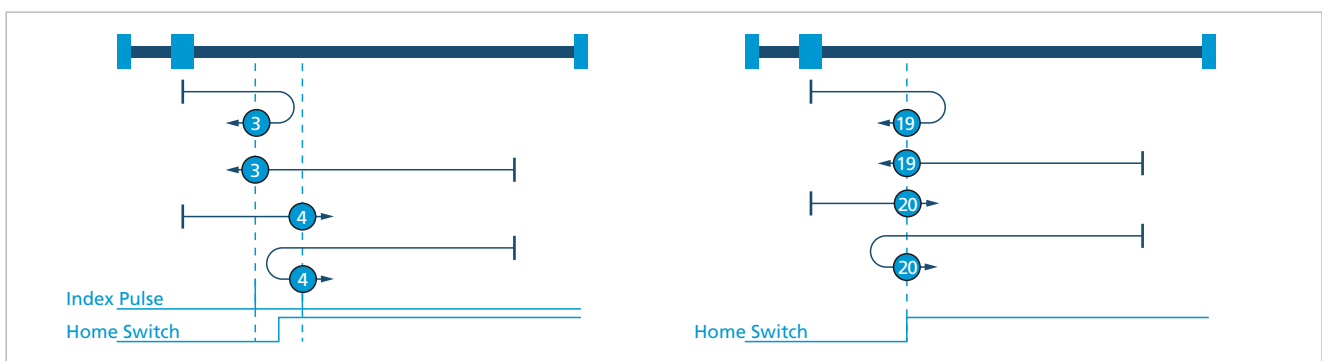


Abb. 12: Homing-Methode 3, 4, 19 und 20

## Funktionsbeschreibung

### Methode 5, 6 und 21, 22

Homing an einem negativen Homing-Schalter (Negative Home Switch).

Die initiale Bewegungsrichtung ist abhängig von dem Zustand des Home Switch. Die Homingposition liegt an der Stelle, an der sich der Zustand des Home Switch ändert. Falls bei einer Referenzfahrt die Bewegungsrichtung umgedreht werden muss, geschieht dies immer an der Stelle, an der sich der Zustand des Home Switch ändert.

Die Homingposition liegt an der Stelle, an der sich der Zustand des Home Switch ändert.

Bei den Methoden 21 und 22 wird die Homingposition auf eine Flanke gesetzt, Indeximpulse werden nicht detektiert.

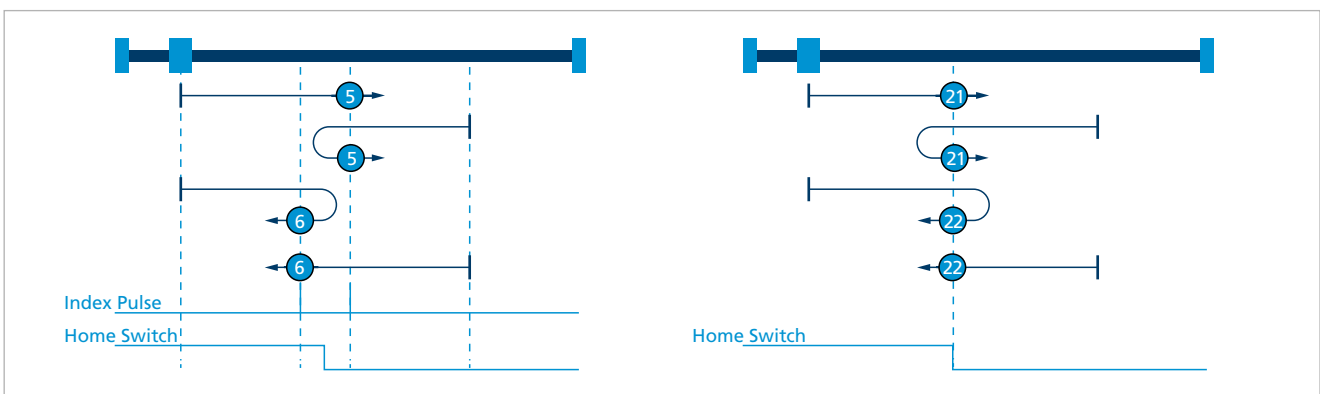


Abb. 13: Homing-Methode 5, 6, 21 und 22

### Methode 7 bis 14 und 23 bis 30

Homing am Homing-Schalter (Home Switch)

Diese Methoden verwenden einen Homing-Schalter, der nur in einem bestimmten Bereich des Weges aktiv ist. Dabei wird unterschieden, wie auf die beiden Flanken des Home Switch reagiert werden soll. Bei den Methoden 7 bis 14 wird nach Detektion der Flanke bis zum Indeximpuls weitergefahren, an dem die Homingposition dann gesetzt wird. Bei den Methoden 23 bis 30 wird die Homingposition auf eine Flanke gesetzt, Indeximpulse werden nicht detektiert.

#### ■ Methode 7 und 23:

Homing an fallender Flanke unten. Start in positiver Richtung, wenn Schalter inaktiv.

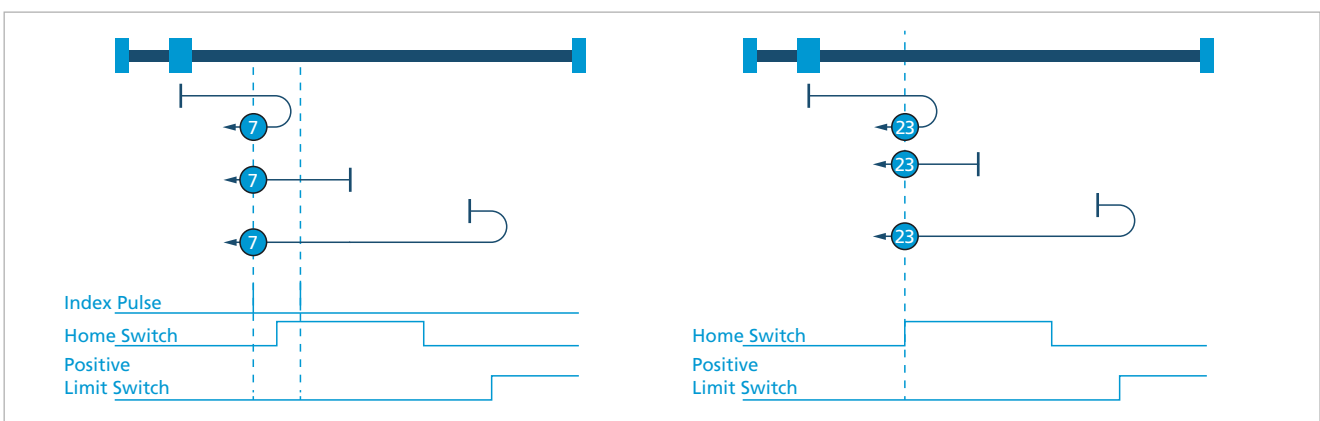


Abb. 14: Homing-Methode 7 und 23

## Funktionsbeschreibung

- Methode 8 und 24:

Homing an steigender Flanke unten. Start in positiver Richtung, wenn Schalter inaktiv.

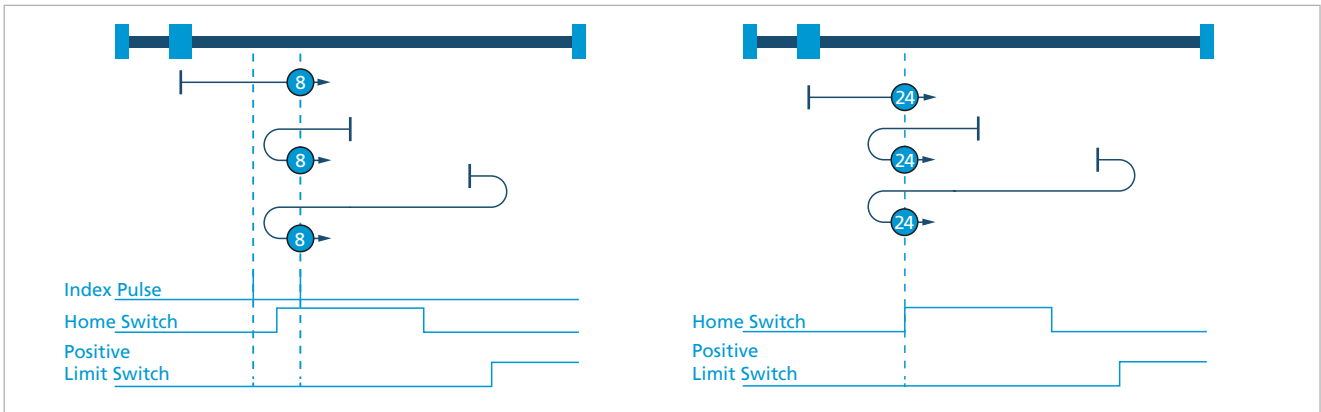


Abb. 15: Homing-Methode 8 und 24

- Methode 9 und 25:

Homing an steigender Flanke oben. Start immer in positiver Richtung.

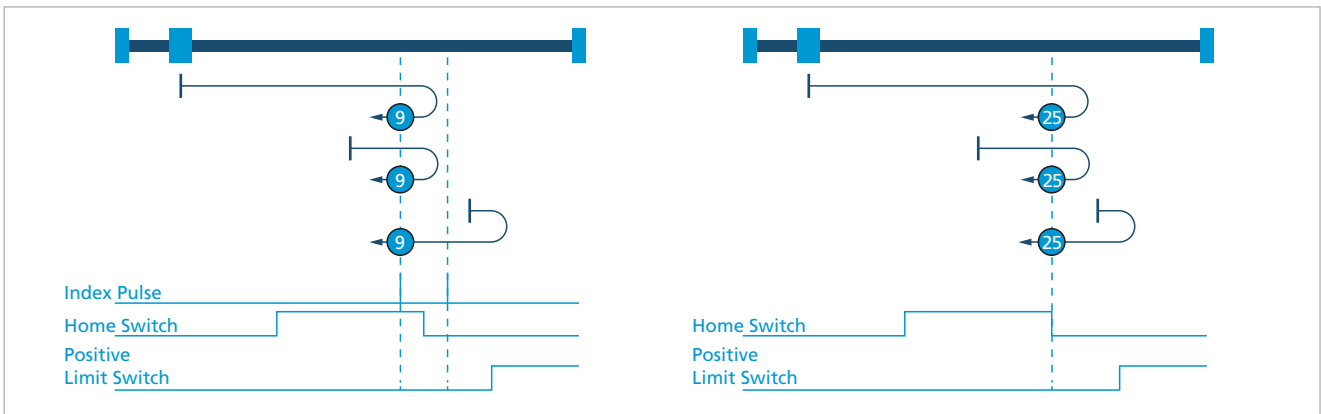


Abb. 16: Homing-Methode 9 und 25

- Methode 10 und 26:

Homing an fallender Flanke oben. Start immer in positiver Richtung.

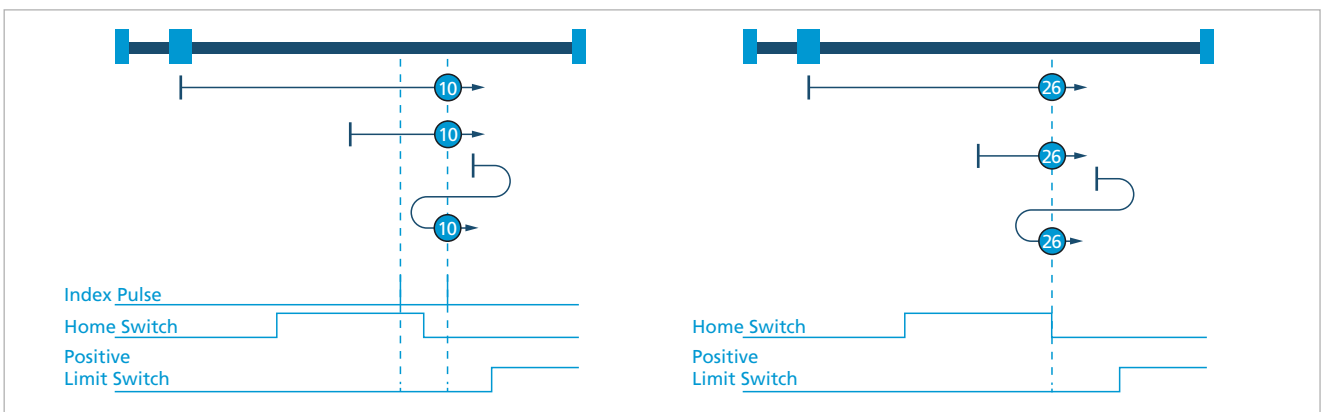


Abb. 17: Homing-Methode 10 und 26

## Funktionsbeschreibung

- Methode 11 und 27:

Homing an fallender Flanke oben. Start in negativer Richtung, wenn Schalter inaktiv.

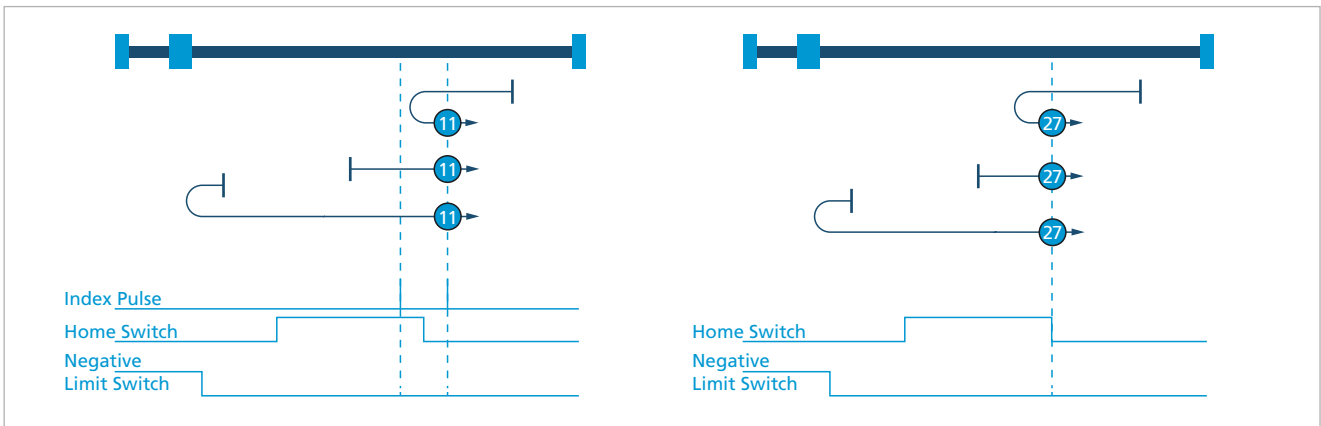


Abb. 18: Homing-Methode 11 und 27

- Methode 12 und 28:

Homing an steigender Flanke oben. Start in negativer Richtung, wenn Schalter inaktiv.

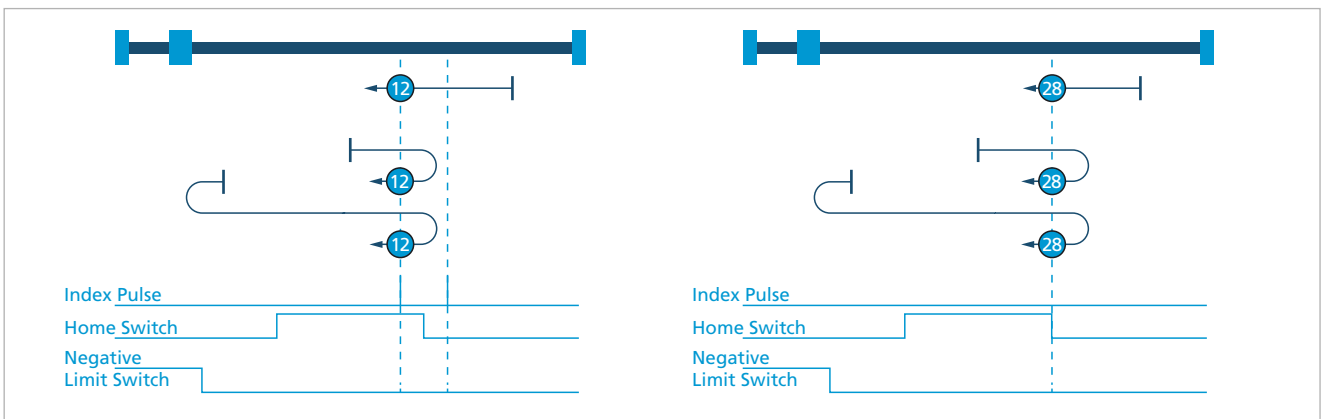


Abb. 19: Homing-Methode 12 und 28

## Funktionsbeschreibung

- Methode 13 und 29:  
Homing an steigender Flanke unten. Start immer in negativer Richtung.

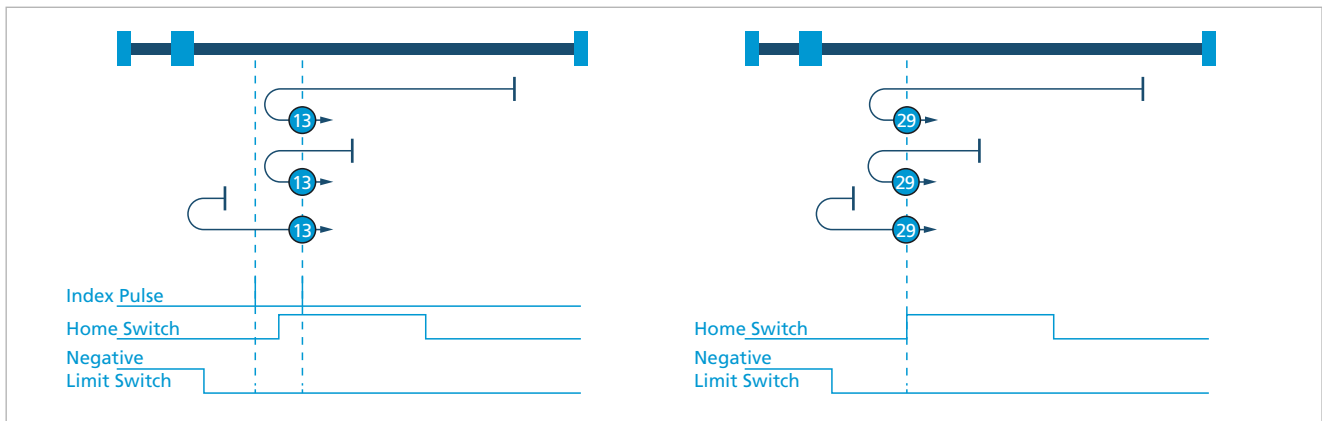


Abb. 20: Homing-Methode 13 und 29

- Methode 14 und 30:  
Homing an fallender Flanke unten. Start immer in negativer Richtung.

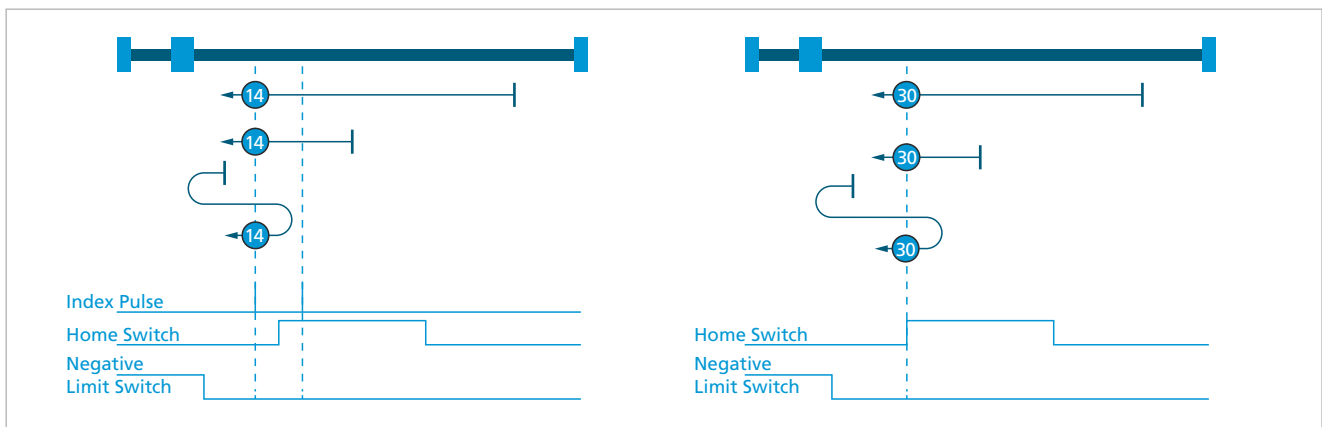


Abb. 21: Homing-Methode 14 und 30

### Methode 33 und 34

Homing am Indeximpuls. Antrieb fährt in negativer (33) oder positiver (34) Richtung bis zum Indeximpuls.

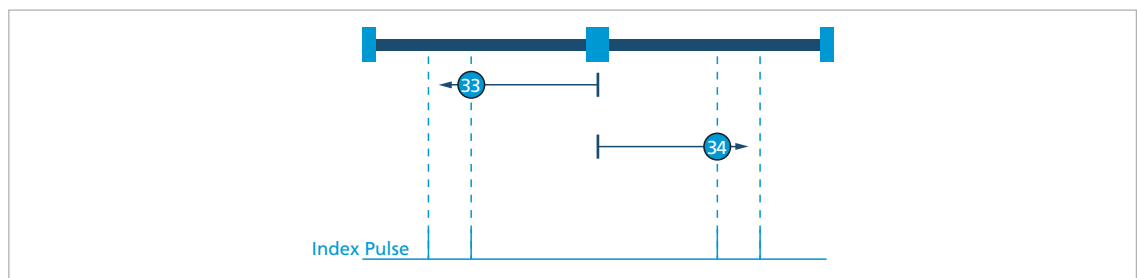


Abb. 22: Homing-Methode 33 und 34

## Funktionsbeschreibung

### Methode 35

Der Positionszähler wird an der aktuellen Position genullt.

**i** Endschalter und Homing-Schalter werden im Drehzahlmodus angefahren, ein Index-Impuls im Positioniermodus. Dabei werden die eingestellten Bereichsgrenzen (0x607D), sofern für die jeweilige Betriebsart aktiviert (0x2338), berücksichtigt.

### Homing Speed (0x6099)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6099	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Switch Seek Velocity	U32	rw	400	Drehzahl bei Schaltersuche [ $\text{min}^{-1}$ ]
	0x02	Homing Speed	U32	rw	100	Drehzahl bei Nullpunktsuche [ $\text{min}^{-1}$ ]

### Homing Acceleration (0x609A)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x609A	0x00	Homing Acceleration	U32	rw	50	Beschleunigung während des Homings [ $1/\text{s}^2$ ]

## 4.5.2 Ablauf einer Homing-Referenzfahrt

- ✓ NMT-Zustand *Operational*
  - ✓ Antriebszustand *Operation Enabled*
  - ✓ Modes of Operation (0x6060) sind auf Homing Mode (6) gesetzt
1. Folgende Objekte mit den gewünschten Werten belegen:
    - Homing Switch (Objekt 0x2310)
    - Homing Method (Objekt 0x6098)
    - Homing Speed (Objekt 0x6099)
    - Homing Acceleration (Objekt 0x609A)
  2. Im Controlword Bit 4 (Homing Operation Start) auf 1 setzen.
- ↻ Antrieb antwortet mit 0 auf Bit 12 und Bit 10 des Statusword.
  - ↻ Antrieb startet die Referenzfahrt.
  - ↻ Wenn die Homing-Position erreicht und die Referenzfahrt abgeschlossen wurde, werden Bit 12 und Bit 10 des Statusword auf 1 gesetzt.

Eine laufende Referenzfahrt kann durch zurücksetzen von Bit 4 im Controlword abgebrochen werden. Dies wird mit gesetztem Bit 10 (Target Reached) und zurückgesetztem Bit 12 (Homing Attained) im Statusword quittiert.

**i** Vor der erneuten Ausführung einer Referenzfahrt muss Bit 4 im Controlword zurückgesetzt werden. Dabei wird Bit 12 im Statusword zurückgesetzt.

## Funktionsbeschreibung

### 4.6 Profile Velocity Mode

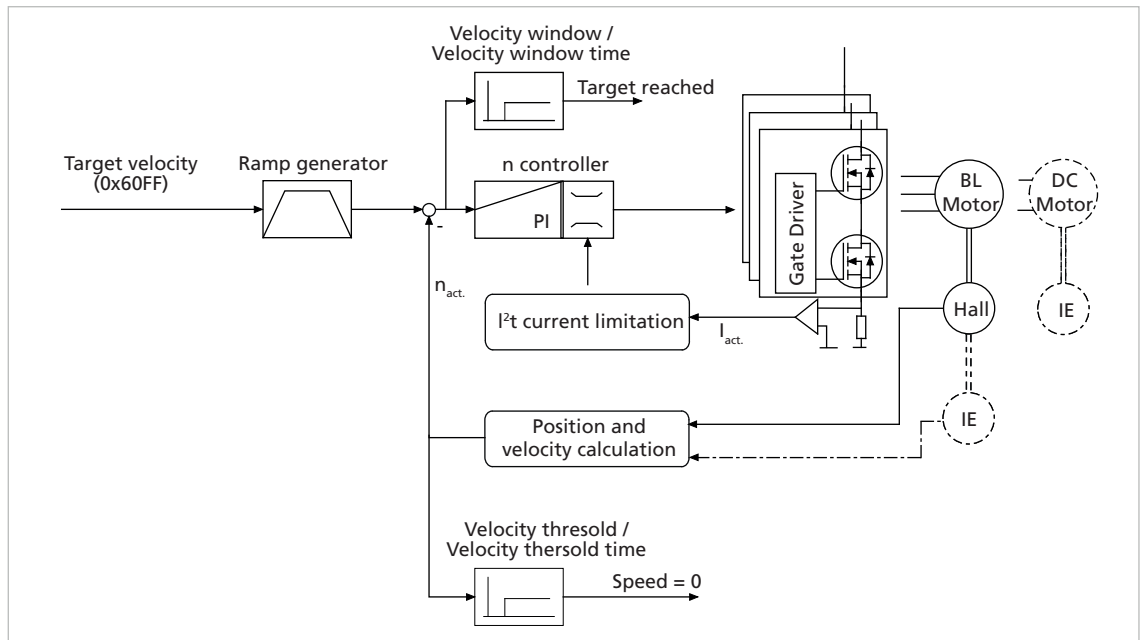


Abb. 23: Reglerstruktur bei Positionsregelung im Profile Velocity Mode

#### 4.6.1 Betriebsart im Überblick

Im Profile Velocity Mode (PV) wird die Drehzahl des Antriebs über einen PI-Regler geregelt. Dadurch wird gewährleistet, dass der Antrieb ohne Abweichung von der Vorgabe betrieben wird, sofern dieser nicht überlastet ist.

Voraussetzungen für den Betrieb des Antriebs im PV Mode:

- Die Betriebsart Profile Velocity Mode ist im Parameter Modes of Operation (0x6060) eingestellt.
- Der Antrieb befindet sich im NMT-Zustand *Operation Enabled*.
- Der Drehzahlregler ist korrekt auf die Anwendung angepasst.

Die Drehzahl wird über das Objekt Target Velocity (0x60FF) im Objektverzeichnis vorgegeben. In der Betriebsart Profile Velocity Mode folgt der Antrieb jedem neu übergebenen Sollwert unmittelbar. Dabei werden die eingestellten Maximalwerte für Beschleunigung, Bremsrampe und Drehzahl berücksichtigt.

#### 4.6.2 Benachrichtigung der übergeordneten Steuerung

Das Erreichen der Soll Drehzahl wird über das Bit 10 (Target Reached) im Statusword des Antriebs signalisiert. Ein stillstehender Antrieb wird über Bit 12 (Speed = 0) gemeldet. Falls der Übertragungstyp für das jeweilige PDO auf 255 eingestellt ist, wird das PDO asynchron, getriggert durch den Zustandswechsel, übertragen.



## Funktionsbeschreibung

### 4.6.3 Grundeinstellungen

Für den Drehzahlregler können über das Objekt Velocity Control Parameter Set (0x2331) die Proportionalverstärkung und der I-Anteil eingestellt werden.

#### Velocity Control Parameter Set (0x2331)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2331	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Proportional Term POR	U16	rw	a)	Proportionalverstärkung des Drehzahlreglers
	0x02	Integral Term I	U16	rw	a)	Integralanteil des Drehzahlreglers

a) *Abhängig von der Konfiguration des Motion Controllers*

Die Abtastrate kann über das Objekt Sampling Rate (0x2330.01) als Vielfaches der internen Abtastrate zwischen 1 und 20 eingestellt werden.

Die interne Abtastrate beträgt 0,2 ms. Bei MCDC und MCBL-AES beträgt die Abtastrate 0,1 ms.

#### Filter Settings (0x2330)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2330	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Sampling Rate	U16	rw	1	Faktor der Abtastrate
	0x02	Gain Scheduling	U16	rw	0	1: Reduzierte Reglerverstärkung im Zielkorridor bei Positionierung

### 4.6.4 Drehzahlwert

Bei BL-Motoren wird die aktuelle Drehzahl über die Auswertung der analogen Hallsignale bestimmt. Alternativ kann auch ein externer Inkrementalencoder für die Drehzahlerfassung verwendet werden.

Der Sensortyp und die Auflösung des externen Encoders werden über das Objekt 0x2351 eingestellt. Bei DC-Motoren wird die Drehzahl immer über den Inkrementalencoder ermittelt.

## Funktionsbeschreibung

### 4.6.5 Zusätzliche Einstellungen

#### 4.6.5.1 Grenzen der Bewegung

Die Software Position Limits (0x607D) können auch für den Drehzahlmodus über das Objekt General Settings (0x2338) aktiviert werden.

#### 4.6.5.2 Rampengenerator

Nach Vorgabe einer neuen Solldrehzahl über das Objekt Target Velocity (0x60FF) wird der Antrieb im Profile Velocity Mode mit der im Objekt Profile Acceleration (0x6083) hinterlegten Beschleunigung auf die neue Drehzahl beschleunigt oder abgebremst. Der Parameter ist in beiden Richtungen gültig.

#### 4.6.5.3 Strombegrenzung

Über die Spitzen- und Dauerstrombegrenzungswerte (Objekt 0x2333) kann der Antrieb vor Überlast geschützt werden.

### 4.6.6 Befehle zur Bewegungssteuerung

Ein Drehzahlsollwert wird über das Objekt Target Velocity (0x60FF) vorgegeben. Sofern sich der Antrieb im Zustand *Operation Enable* befindet (siehe Kap. 4.2, S. 40), wird der Antrieb unmittelbar auf die neue Solldrehzahl beschleunigt.

Über den Parameter Velocity Window (0x606D) wird ein Fenster um die Zieldrehzahl definiert, innerhalb dessen die Solldrehzahl als erreicht signalisiert wird, wenn die Drehzahl für mindestens die über den Parameter Velocity Window Time (0x606E) definierte Zeit innerhalb des Zielfensters verbleibt.

Die erreichte Zieldrehzahl wird im Statusword über das Bit 10 (Target Reached) signalisiert.

Über die Parameter Velocity Threshold (0x606F) wird ein Schwellwert für die Drehzahl definiert, unterhalb dessen der Antrieb als stehend signalisiert wird, wenn die Drehzahl für mindestens die über den Parameter Velocity Threshold Time (0x6070) definierte Zeit unterhalb des Schwellwerts verbleibt.

Stillstand wird im Statusword über das Bit 12 (Speed=0) signalisiert.


#### Target Velocity (0x60FF)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x60FF	0x00	Target Velocity	I32	rw	–	Drehzahlsollwert [ $\text{min}^{-1}$ ]

Die zuletzt eingestellte Solldrehzahl kann über das Objekt Velocity Demand Value (0x606B) abgefragt werden. Der aktuelle Drehzahlwert kann über das Objekt Velocity Actual Value (0x606C) abgefragt werden (siehe Kap. 6.3, S. 110).

### 4.6.7 Zusammengesetzte Bewegungsprofile

Über die Auswertung der Bits 10 (Target Reached) und 12 (Speed = 0) im Statusword können gezielt Drehzahlprofile abgefahren werden. Die Beschleunigung wird über das Objekt Profile Acceleration (0x6083) festgelegt.

 Wenn der Antrieb nicht an den eingestellten Bereichsgrenzen stoppen soll, dürfen die Position Limits für den Velocity Mode (0x2338.02) nicht aktiviert sein.

Sicherstellen, dass die Maximaldrehzahl (0x607F) nicht kleiner als die gewünschte Solldrehzahl ist.

## Funktionsbeschreibung

### 4.7 Cyclic Synchronous Position Mode

#### 4.7.1 Betriebsart im Überblick

Im Cyclic Synchronous Position Mode (CSP) positioniert der Antrieb sofort auf die übergebene Zielposition. Im Gegensatz zum Profile Position Mode ist keine Quittierung über Controlword/Statusword notwendig. Übergebene Positionen werden sofort angefahren.

Voraussetzung dafür ist, dass nur kleine Sollwert-Sprünge vorgegeben werden dürfen, die schnell hintereinander kommen müssen.

Ein typischer Anwendungsfall für diese Betriebsart sind Mehrachs-Steuerungen. Diese müssen jeder Achse und damit jedem Antrieb zu jeder Zeit genau vorgeben, wo er sich befinden muss, um Kollisionen zu vermeiden.

Der Cyclic Synchronous Position Mode findet sich in mehreren FAULHABER Ansteuerungen, z. B. in den integrierten Ansteuerungen der COD-Reihe (z. B. 2232...BX4 COD, 2250...BX4 COD) oder den separaten Ansteuerungen (MCDC 3002 x CO, MCBL 3002 CO). Ob eine vorliegende Ansteuerung tatsächlich den synchronen Positioniermodus CSP unterstützt, lässt sich am Objekt 0x6502 ablesen. Ist das Bit 7 gesetzt, ist CSP implementiert.

Voraussetzungen für den Betrieb des Antriebs im CSP Mode:

- Die Betriebsart Cyclic Synchronous Position Mode ist im Parameter Modes of Operation (0x6060) eingestellt.
- Der Antrieb befindet sich im NMT-Zustand *Operation Enabled*.

Die Reglerstruktur im Cyclic Synchronous Position Mode ist die gleiche wie im Profile Position Mode (siehe Kap. 4.4.1, S. 52).

Die Zielposition wird immer als absoluter Wert angegeben. Relative Positionierungen sind nicht möglich.

Eingestellte Begrenzungen des Antriebs, wie Max Profile Velocity oder Brems- und Beschleunigungsrampen, sind weiterhin aktiv.

#### 4.7.2 Eingänge und Ausgaben

Eingänge aus Sicht der Steuerung:

- Target Position (0x607A)

Ausgaben:

- Motor Current Actual Value (0x6078)
- Velocity Actual Value (0x606C)
- Position Actual Value (0x6064)

## Funktionsbeschreibung

### 4.7.3 Statusword und Controlword

#### Statusword

In der Betriebsart Cyclic Synchronous Position Mode hat das Statusword folgende betriebsarten-spezifische Bits:

Bit	Funktion	Beschreibung
10	Reserved	0: Keine Funktion
12	Drive Follows the Command Value	1: Antrieb folgt dem Betriebswert, Target Position wird als Eingang der Positionsregelung verwendet
13	Following Error	0: Istposition folgt der Vorgabe ohne Schleppfehler 1: Zulässiger Bereich für den Schleppfehler überschritten

#### Controlword

Das Controlword hat in der Betriebsart Cyclic Synchronous Position Mode keine betriebsarten-spezifische Bits.

### 4.7.4 Einstellungen

Folgende Objekte müssen zur Verwendung der Betriebsart eingestellt werden:

- Target Position (0x607A)
- Operation Mode (0x6060 = 8)

Abb. 24 zeigt alle in der Betriebsart wirksamen Objekte. Die zusätzlich abgebildeten Objekte ermöglichen optionale Einstellungen der Betriebsart.

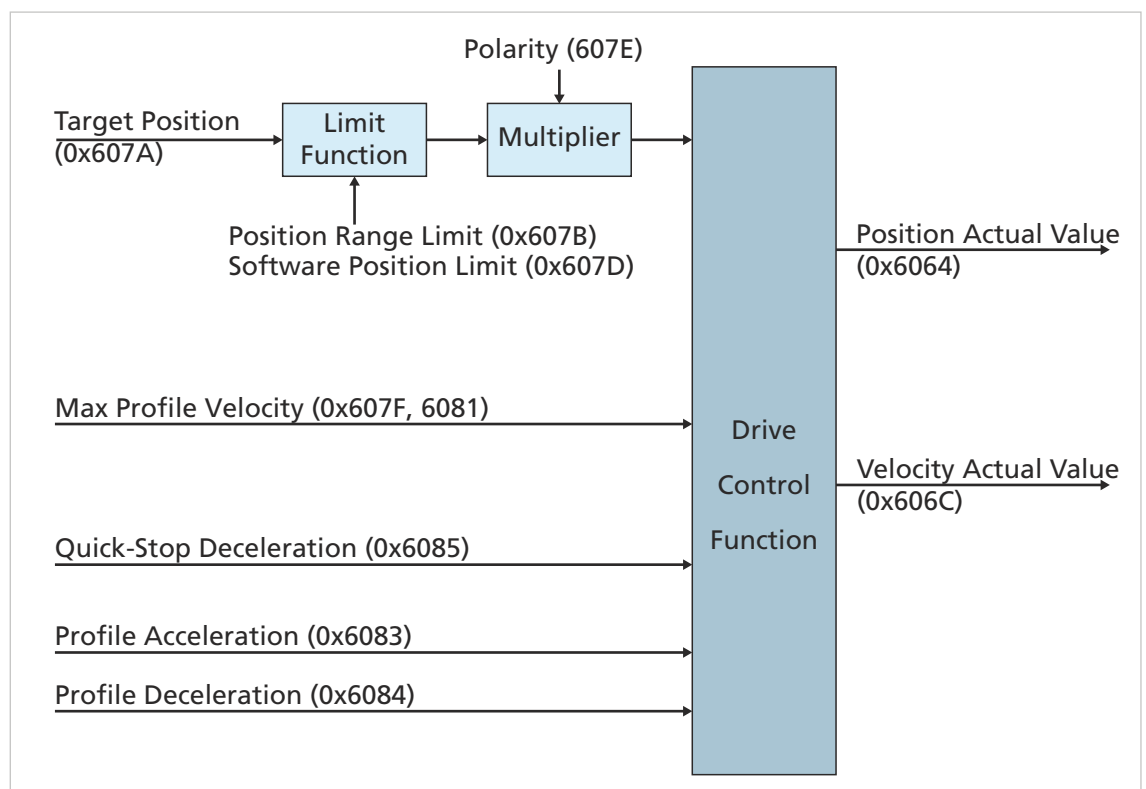


Abb. 24: Übersicht aller wirksamen Objekte in der Betriebsart CSP

## Funktionsbeschreibung

### 4.8 Eingänge/Ausgänge

#### 4.8.1 Endschalteranschlüsse und Schaltpegel

Folgende Anschlüsse können als Referenz- und Endschaltereingänge verwendet werden:

- AnIn
- Fault
- 3. Input
- 4., 5. Input (nur MCDC)

Zusätzlich steht der Nulldurchgang der Hallsensordesignale bei BL-Motoren als Indeximpuls zur Verfügung. Je nach Motortyp (zweipolig oder vierpolig) tritt der Indeximpuls einmal oder zweimal pro Umdrehung auf. An den Fault-Pin kann auch der Indeximpuls eines externen Encoders angeschlossen werden, über den die Istposition exakt genullt werden kann.

Die Anschlüsse AnIn und Fault sind als Interrupteingänge ausgelegt. Dies bedeutet, dass sie flankengetriggert sind. Alle anderen Eingänge sind nicht flankengetriggert. Bei ihnen muss das Signal mindestens 500 µs anliegen, um sicher detektiert werden zu können. Die maximale Reaktionszeit auf Pegeländerungen an allen Eingängen beträgt 500 µs.

##### 4.8.1.1 Konfiguration der digitalen Eingänge

###### Schaltpegel

Über das Objekt Input Threshold Level (0x2316) kann der Schaltpegel aller digitalen Eingänge auf 5V-TTL-kompatibel oder 24V-PLC-kompatibel (Standard) eingestellt werden. Im Datenblatt der eingesetzten Steuerung finden Sie genaue Angaben zu den jeweiligen Schaltschwellen und den zugelassenen Spannungsbereichen.


###### Input Threshold Level (0x2316)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2316	0x00	Input Threshold Level	U8	rw	1	Schaltpegel <ul style="list-style-type: none"> <li>■ 0: 5V-TTL</li> <li>■ 1: 24V-PLC</li> </ul>

###### Endschalter- und Homingschaltereinstellung

Die zur Verfügung stehenden digitalen Eingänge können jeweils als Endschalter oder Homing-Schalter zur Verwendung innerhalb einer DSP402 Homing Method konfiguriert werden. Die oberen und unteren Endschalter dienen zusätzlich als Bereichs-Endschalter, über die nicht hinausgefahren werden kann (Hard-Blocking).

Werden unterer und oberer Endschalter nicht für eine DSP402 Homing Method verwendet, kann deren Schalt-Polarität über den Parameter Switch Polarity festgelegt werden (steigende oder fallende Flanke gültig). Die Homing-Methoden 1, 2, 17 und 18 gehen standardmäßig von einem positiv schaltenden Endschalter aus. Soll hingegen ein negativ schaltender Endschalter verwendet werden, muss die gewünschte Polarität hier entsprechend eingestellt und zusätzlich der Parameter Polarity for Homing Limit auf 1 gesetzt werden.

 Die Eingangskonfiguration kann nicht im Homing Mode geändert werden. Wählen Sie hierfür den Profile Position Mode oder den Profile Velocity Mode.

## Funktionsbeschreibung

### Digital Input Settings (0x2310)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2310	0x00	Number of Entries	U8	ro	6	Anzahl Objekteinträge
	0x01	Negative Limit Switch	U8	rw	0	Auswahl unterer Endschalter
	0x02	Positive Limit Switch	U8	rw	0	Auswahl oberer Endschalter
	0x03	Homing Switch	U8	rw	0x07 / 0x1F <sup>a)</sup>	Homing-Schalter
	0x05	Switch Polarity	U8	rw	0x07 / 0x1F <sup>a)</sup>	Polarität der Endschalter <ul style="list-style-type: none"> <li>▪ 1: Positive Flanke gültig</li> <li>▪ 0: Negative Flanke gültig</li> </ul>
	0x06	Polarity for Homing Limit	U8	rw	0	Polarität der Endschalter auch bei DSP402 Limit Switch Homing Method verwenden

a) BL-Controller/MCDC

Die Einstellungen der digitalen Eingänge erfolgen mit folgender Bitmaske:

Bit:	0	1	2	3	4	5	6	7
Eingang:	Analoger Eingang	Fault-Pin	3. Eingang	4. Eingang (nur MCDC)	5. Eingang (nur MCDC)	–	–	–

### Erläuterungen

Subindex	Name	Erläuterung
0x01	Negative Limit Switch	Hier wird der Eingang angegeben, an dem der untere Endschalter für die Homing-Methoden 1 und 17 oder für eine Hard-Blocking Funktion angeschlossen ist. Bei aktiviertem Endschalter wird der Antrieb gestoppt und lässt sich nur in entgegengesetzter Richtung aus dem Endschalter herausfahren.
0x02	Positive Limit Switch	Hier wird der Eingang angegeben, an dem der obere Endschalter für die Homing-Methoden 2 und 18 oder für eine Hard-Blocking Funktion angeschlossen ist. Bei aktiviertem Endschalter wird der Antrieb gestoppt und lässt sich nur in entgegengesetzter Richtung aus dem Endschalter herausfahren.
0x03	Homing Switch	Hier wird der Eingang angegeben, an dem der Homing-Schalter für die Homing-Methoden 3 bis 14 und 19 bis 30 angeschlossen ist. Switch Polarity (Subindex 0x05) kann hierbei nicht verwendet werden.
0x05	Switch Polarity	Hier kann die Polarität der Hard-Blocking Endschalter eingestellt werden. Soll die Polarität auch bei den Homing-Methoden 1, 2, 17 und 18 geändert werden, muss zusätzlich Subindex 0x06 auf 1 gesetzt sein.
0x06	Polarity for Homing Limit	Hier kann angegeben werden, ob bei den Homing-Methoden 1, 2, 17 und 18 die Polaritätseinstellungen unter Subindex 0x05 verwendet werden sollen. Die Einstellung kann nur generell für alle Eingänge gesetzt werden (keine Bitmasken-Kodierung).

Zur Beschreibung der Homing-Methoden siehe Kap. 4.5.1, S. 57.

Die Endschalterfunktionen für den Fault-Pin werden nur angenommen, wenn dieser über Objekt 0x2315.01 als Referenzeingang (Fault-Pin Function 4) konfiguriert ist. Die Einstellung muss mit SAVE gespeichert werden.

In der Standardkonfiguration sind alle Eingänge als Homing-Eingänge konfiguriert. Ein Homing-Schalter kann an jeden Eingang angeschlossen werden. Es wird empfohlen, nur

## Funktionsbeschreibung

den Homing-Eingang anzugeben, an dem tatsächlich der Referenzschalter angeschlossen ist.

Wird eine Homing-Methode ausgeführt, ohne dass der hierfür benötigte Schalter in Objekt 0x2310 definiert ist, startet die Referenzfahrt nicht.

### 4.8.2 Sonderfunktionen des Fault-Pins

Der Fehleranschluss (Fault-Pin) kann über das Objekt Fault-Pin Settings (0x2315) für unterschiedliche Aufgaben als Ein- oder Ausgang konfiguriert werden:

#### Fault-Pin Settings (0x2315)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2315	0x00	Number of Entries	U8	ro	3	Anzahl Objekteinträge
	0x01	Fault-Pin Function	U8	rw	0	Funktion des Fault-Pins <ul style="list-style-type: none"> <li>▪ 0: Error Output</li> <li>▪ 2: Digital Output</li> <li>▪ 4: Reference Input</li> <li>▪ 5: Position Output</li> </ul>
	0x03	Digital Output Status	U8	rw/ro <sup>a)</sup>	a)	Zustand des Pins in der Funktion Digital Output ändern <ul style="list-style-type: none"> <li>▪ 0: Clear Output</li> <li>▪ 1: Set Output</li> <li>▪ 2: Toggle Output</li> </ul>

a) Abhängig von der Konfiguration des Motion Controllers

#### Fault-Pin Function (0x2315.01)

Wert	Funktion	Beschreibung
0	Error Output	Fault-Pin als Fehlerausgang
2	Digital Output	Fault-Pin als digitaler Ausgang. Der Ausgang wird auf Low Pegel gesetzt.
4	Reference Input	Fault-Pin als Referenz- oder Endschaltereingang.
5	Position Output	Fault-Pin als digitaler Ausgang zur Anzeige der Bedingung „Zielposition erreicht“

In der Funktion als Ausgang ist der Anschluss als Open Collector ausgeführt. Somit wird mit Setzen des Ausganges der Anschluss auf Low Pegel gesetzt.

## Funktionsbeschreibung

### 4.8.2.1 Fault-Pin als Fehlerausgang

Mit der Funktion Error Output wird der Ausgang gesetzt, sobald ein Fehler des FAULHABER Fehlerregisters auftritt und die Errout Mask (0x2321.03) für den entsprechenden Fehler auf 1 gesetzt ist (siehe Kap. 3.10, S. 35).

#### Zusätzliche Einstellungen

##### Verzögerte Signalisierung

Um ein kurzzeitiges Auftreten von Fehlern zum Beispiel während der Beschleunigungsphase auszublenden, kann über das Objekt Error Handling (0x2322.01) eine Fehlerverzögerung (Error Delay) eingestellt werden. Die Fehlerverzögerung gibt an, wie lange ein Fehler anstehen muss, bis er am Fehlerausgang angezeigt wird (siehe Kap. 3.10, S. 35). Die Fehlerverzögerung wirkt bei den Fehlern Continuous Over Current, Deviation und Over Voltage.

##### Beispiel:

Fehler erst nach 2 Sekunden anzeigen:

- Error Delay = 200 (Objekt 0x2322.01)

### 4.8.2.2 Fault-Pin als digitaler Ausgang

Mit der Funktion Digital Output kann der Fehleranschluss als universeller digitaler Ausgang verwendet werden. Über das Objekt 0x2315.03 kann der digitale Ausgang gesetzt oder gelöscht werden:

#### Digital Output Status (0x2315.03)

Wert	Funktion	Beschreibung
0	Clear Output	Digitalen Ausgang auf Low Pegel setzen
1	Set Output	Digitalen Ausgang auf High Pegel setzen
2	Toggle Output	Digitalen Ausgang umschalten

### 4.8.2.3 Fault-Pin als Referenzeingang

Mit der Funktion Reference Input hat der Fault-Pin die Funktion eines digitalen Eingangs und kann als Endschalter oder Homing-Schalter entsprechend Objekt 0x2310 oder zum Anschluss des Index-Impulses eines Inkrementalencoders verwendet werden.

### 4.8.2.4 Fault-Pin als „Position erreicht“-Ausgang

Mit der Funktion Position Output wird der Ausgang gesetzt, wenn im Profile Position Mode die Zielposition erreicht wurde, entsprechend den Bedingungen in Position Window (0x6067) und Position Window Time (0x6068). Beim nächsten Positionier-Startbefehl wird der Ausgang wieder zurückgesetzt.



#### HINWEIS!

##### Schäden an der Elektronik

Das Anlegen einer Spannung am Fault-Pin, während dieser nicht als Eingang konfiguriert ist, kann zu Schäden an der Elektronik führen.

- ▶ Den Fault-Pin als Eingang konfigurieren, bevor Spannung von außen angelegt wird.



## Funktionsbeschreibung

### 4.8.3 Abfrage der Eingangszustände

Der Zustand der digitalen Eingänge kann über das Objekt Digital Input Status (0x2311) als direkter Eingangspegel (Subindex 0x02) oder polaritätsbewertet (Subindex 0x01) gemäß dem Eintrag unter 0x2310.05 abgefragt werden. Die Zustandsanzeige erfolgt entsprechend der Bitmaske von Objekt 0x2310 (siehe Kap. 4.8.1, S. 69).

#### Digital Input Status (0x2311)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2311	0x00	Number of Entries	U8	ro	4	Anzahl Objekteinträge
	0x01	Input Status	U8	ro	0	Logische Zustände der digitalen Eingänge
	0x02	Input Level	U8	ro	0	Physikalische Zustände der digitalen Eingänge

Die angelegte Spannung am analogen Eingang und an den anderen Eingängen kann über die Objekte Analog Input Status (0x2313) und Analog Input Status Raw (0x2314) in Millivolt oder in Digits abgefragt werden. Der analoge Eingang kann somit auch ein Messsignal für die übergeordnete Steuerung liefern.

#### Analog Input Status (0x2313)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2313	0x00	Number of Entries	U8	ro	3/5 <sup>a)</sup>	Anzahl Objekteinträge
	0x01	Input 1 ADC Value	I16	ro	-	Spannung an Eingang 1 [mV] (AnIn)
	0x03	Input 3 ADC Value	I16	ro	-	Spannung an Eingang 3 [mV] (3rd In)
	0x04	Input 4 ADC Value	I16	ro	-	Spannung an Eingang 4 [mV] (nur MCDC)
	0x05	Input 5 ADC Value	I16	ro	-	Spannung an Eingang 5 [mV] (nur MCDC)

a) BL-Controller/MCDC

#### Analog Input Status Raw (0x2314)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2314	0x00	Number of Entries	U8	ro	8	Anzahl Objekteinträge
	0x01	Input 1 ADC value raw	I16	ro	-	Digitalwert an Eingang 1
	0x02	Input 2 ADC value raw	I16	ro	-	Digitalwert an Eingang 2
	0x03	Input 3 ADC value raw	I16	ro	-	Digitalwert an Eingang 3
	0x04	Input 4 ADC value raw	I16	ro	-	Digitalwert an Eingang 4
	0x05	Input 5 ADC value raw	I16	ro	-	Digitalwert an Eingang 5
	0x06	Input 6 ADC value raw	I16	ro	-	Digitalwert an Eingang 6
	0x07	Input 7 ADC value raw	I16	ro	-	Digitalwert an Eingang 7
	0x08	Input 8 ADC value raw	I16	ro	-	Digitalwert an Eingang 8

Über dieses Objekt können die aktuellen Roh-Werte der intern verwendeten digitalen Eingänge ausgelesen werden.

## Funktionsbeschreibung

**i** Die Objekte zur Abfrage der Eingangszustände können in PDOs gemappt werden. Die PDOs können dann über SYNC oder RTR zyklisch abgefragt werden. Das automatische Versenden eines PDOs bei Zustandsänderung ist nicht möglich, da PDOs nur bei Änderung des Statusword automatisch versendet werden können.

### 4.9 Abfrage des Gerätezustands

Der aktuelle Gerätezustand (aktuelle Temperaturen und Temperaturschwellen in °C) kann über das Objekt Device Status (0x2323) abgefragt werden.

#### Device Status (0x2323)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2323	0x00	Number of Entries	U8	ro	4	Anzahl Objekteinträge
	0x01	Housing Temperature	U16	ro	0	Gehäusetemperatur [°C]
	0x02	Internal Temperature	U16	ro	0	Spulen- bzw. MOSFET-Temperatur [°C]
	0x03	Max. Temperature Limit	U16	ro	0	Obere Temperaturschwelle [°C]
	0x04	Min. Temperature Limit	U16	ro	0	Untere Temperaturschwelle [°C]

Die Werte der oberen und unteren Temperaturschwelle zeigen die Ein- und Ausschalt-schwelle der integrierten Übertemperatursicherung (siehe Kap. 4.10.4, S. 80).

## Funktionsbeschreibung

### 4.10 Technische Informationen

#### 4.10.1 Rampengenerator

In allen Betriebsarten wird der Sollwert über den Rampengenerator geführt.

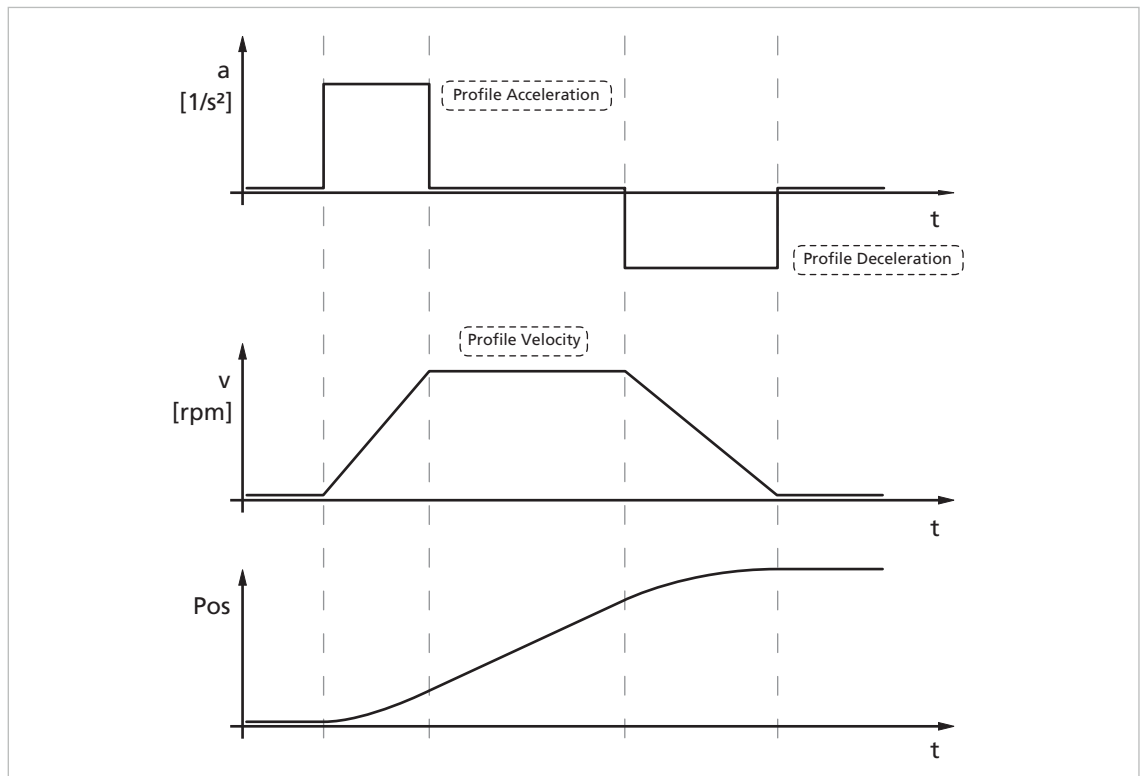


Abb. 25: Grundfunktion des Rampengenerators

Damit können die maximale Beschleunigung (Profile Acceleration), die maximale Verzögerung (Profile Deceleration) und die maximale Drehzahl (Profile Velocity) getrennt anwendungsspezifisch parametrisiert werden.

##### 4.10.1.1 Grundeinstellungen

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6081	0x00	Profile Velocity	U32	rw	a)	Maximaldrehzahl in $\text{min}^{-1}$
0x6083	0x00	Profile Acceleration	U32	rw	30 000	Maximale Beschleunigung [ $1/\text{s}^2$ ]
0x6084	0x00	Profile Deceleration	U32	rw	30 000	Maximale Bremsrate [ $1/\text{s}^2$ ]

a) Abhängig von der Auslieferungskonfiguration des Motion Controllers

# Funktionsbeschreibung

## 4.10.1.2 Rampengenerator im Profile Velocity Mode

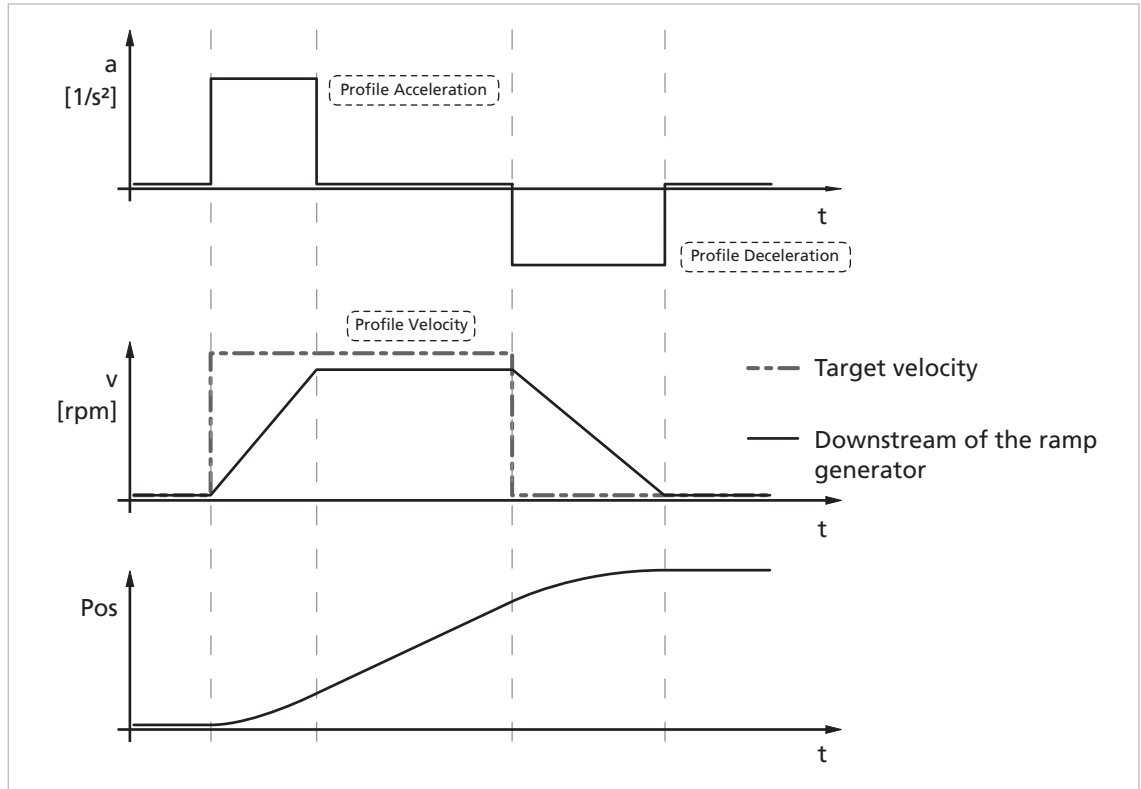


Abb. 26: Eingriff des Rampengenerators im Drehzahlmodus

Im Drehzahlmodus wirkt der Rampengenerator wie ein Filter auf die Solldrehzahl. Der Sollwert wird auf den Profile Velocity Wert begrenzt und Sollwertänderungen entsprechend der Profile Acceleration und Profile Deceleration begrenzt.

## Funktionsbeschreibung

### 4.10.1.3 Rampengenerator im Profile Position Mode

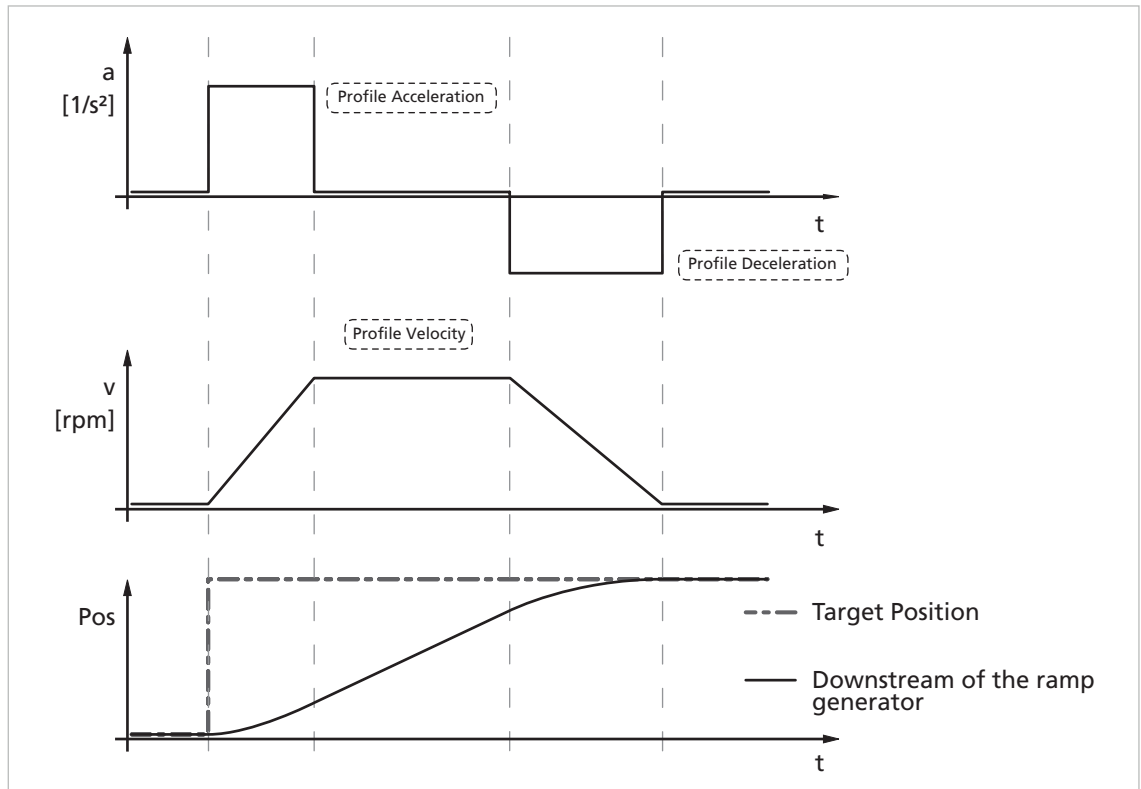


Abb. 27: Eingriff des Rampengenerators im Positionierbetrieb

Im Positionierbetrieb wird über den Positionsregler aus der Differenz zwischen Sollposition und Istposition eine Vorgabegeschwindigkeit ermittelt.

Im Rampengenerator wird die vom Positionsregler ausgegebene Vorgabegeschwindigkeit auf den Profile Velocity Wert begrenzt und die Beschleunigungen werden entsprechend der Beschleunigungsrampe Profile Acceleration begrenzt.

Der Bremsvorgang wird im Positionierbetrieb nicht zeitlich gestreckt, da bereits vor Erreichen der Endposition die Geschwindigkeit so weit reduziert werden muss, dass die Zielposition ohne Überschwingen erreicht werden kann.

Entsprechend der Bewegungsgleichung:

$$2a \cdot s = v^2 \rightarrow v_{\max} = \sqrt{2a \cdot s}$$

a: Beschleunigung [m/s<sup>2</sup>]

v: Geschwindigkeit [m/s]

s: Verbleibende Strecke [m]

muss dazu die maximale Geschwindigkeit  $v_{\max}$  proportional zum verbleibenden Weg begrenzt werden.

Die zulässige bzw. abhängig vom Motor und der Trägheit der Last technisch mögliche Verzögerung wird hier über den Parameter Profile Deceleration eingestellt.

## Funktionsbeschreibung

### 4.10.2 Sinuskommütierung

Die FAULHABER Motion Controller für bürstenlose Motoren zeichnen sich durch eine sogenannte Sinuskommütierung aus. Dies bedeutet, dass das vorgegebene Drehfeld immer ideal zum Rotor steht. Dadurch gelingt es, Momentenschwankungen auf ein Minimum zu reduzieren, auch wenn die Drehzahlen sehr klein sind. Außerdem läuft der Motor dadurch besonders leise.

Die Sinuskommütierung wird durch eine Flat-Top-Modulation erweitert, die eine höhere Aussteuerung ermöglicht. Dadurch sind höhere Leerlaufdrehzahlen möglich.

Über den Parameter Pure Sinus Commutation des Objekts General Settings lässt sich das System so einstellen, dass die Sinuskommütierung im oberen Drehzahlbereich in eine Blockkommütierung übergeht. Durch diese Vollaussteuerung kann der komplette Drehzahlbereich des Motors ausgenutzt werden.

#### General Settings (0x2338)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2338	0x00	Number of Entries	U8	ro	3	Anzahl Objekteinträge
	0x01	Pure Sinus Commutation	U16	rw	1	<ul style="list-style-type: none"> <li>▪ 0: Vollaussteuerung</li> <li>▪ 1: Begrenzung auf Sinusform (nicht MCDC)</li> </ul>
	0x02	Activate Position Limits in Velocity Mode	U16	rw	0	1: Eingestellte Positionierbereichsgrenzen auch im Drehzahlmodus verwenden
	0x03	Activate Position Limits in Position Mode	U16	rw	1	0: Keine Bereichsgrenzen im Positioniermodus

## Funktionsbeschreibung

### 4.10.3 Stromregler und $I^2t$ -Strombegrenzung

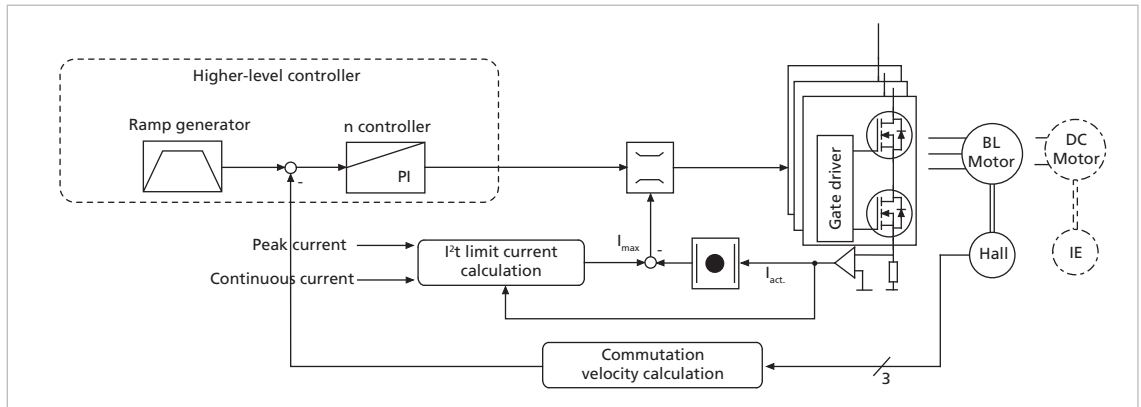


Abb. 28: Eingriff des Strombegrenzungsreglers

Die FAULHABER Motion Controller sind mit einem integralen Stromregler ausgerüstet, der eine Momentenbegrenzung erlaubt.

Der Stromregler arbeitet als Begrenzungsregler. Abhängig von der zurückliegenden Belastung wird durch die  $I^2t$ -Strombegrenzung auf den zulässigen Spitzenstrom oder den Dauerstrom begrenzt. Sobald der Motorstrom den aktuell zulässigen Maximalwert überschreitet, wird über den Stromregler die Spannung begrenzt.

Durch die Ausführung als Strombegrenzungsregler hat die Stromregelung im thermisch entspannten Zustand keinen Einfluss auf die Dynamik der Drehzahlregelung. Das Zeitverhalten dieser Begrenzung ist über den Parameter  $CI$  einstellbar.

Mit den Defaultwerten für  $CI$  wird der Strom nach etwa 5 ms auf den zulässigen Wert begrenzt.

#### Current Control Parameter Set (0x2333)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2333	0x00	Number of Entries	U8	ro	3	Anzahl Objekteinträge
	0x01	Continuous Current Limit	U16	rw	a)	Dauerstrombegrenzung [mA]
	0x02	Peak Current Limit	U16	rw	a)	Spitzenstrombegrenzung [mA]
	0x03	Integral Term CI	U16	rw	a)	Integralanteil des Stromreglers

a) Abhängig von der Konfiguration des Motion Controllers

Bei integrierten Einheiten sind diese Werte bereits voreingestellt. Für externe Steuerungen werden diese Werte durch Auswahl eines Motortyps im Motorassistenten des Motion Managers passend zu Motor und Controller vorbelegt.

#### Arbeitsweise des Stromreglers

Beim Start des Motors wird dem Stromregler der Spitzenstrom als Sollwert vorgegeben. Mit zunehmender Belastung wird der Strom im Motor immer höher, bis er schließlich den Spitzenstrom erreicht. Ab dann tritt der Stromregler in Kraft und begrenzt auf diesen Stromsollwert.

Parallel dazu läuft ein thermisches Strommodell, das aus dem aktuell fließenden Strom eine Modelltemperatur berechnet. Übersteigt diese Modelltemperatur einen kritischen Wert, wird auf den Dauerstrom umgeschaltet und der Motorstrom auf diesen geregelt. Erst wenn

## Funktionsbeschreibung

die Belastung so gering wird, dass die kritische Modelltemperatur unterschritten wird, wird wieder der Spitzenstrom zugelassen.

Das Ziel dieser sogenannten  $I^2t$ -Strombegrenzung ist, den Motor bei geeigneter Wahl des Dauerstroms nicht über die thermisch zulässige Temperatur zu erhitzen. Andererseits sollte kurzzeitig eine hohe Belastung möglich sein, um sehr dynamische Bewegungen realisieren zu können.

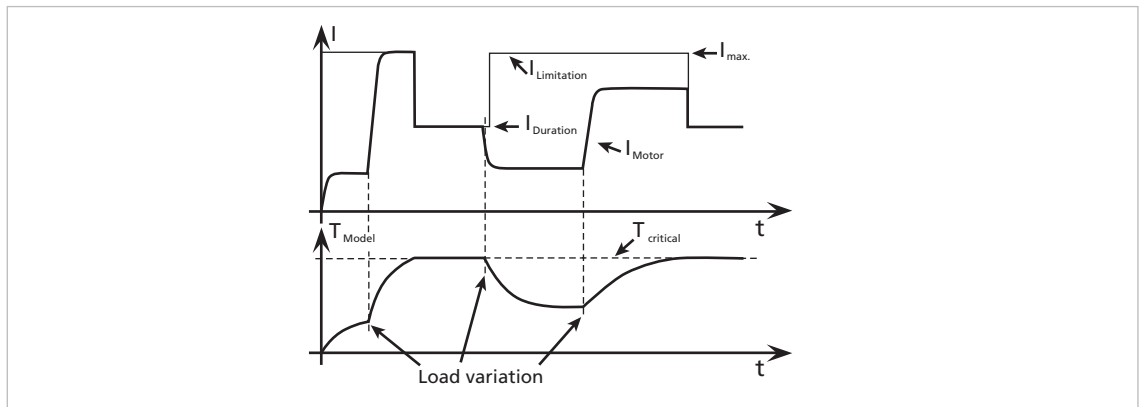


Abb. 29: Funktion der  $I^2t$ -Strombegrenzung

Der aktuell verwendete Strombegrenzungswert (Spitzen- oder Dauerstrom) kann über das Objekt Actual Current Limit abgefragt werden:

### Actual Current Limit (0x2334)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2334	0x00	Actual Current Limit	U16	ro	–	Aktuell verwendeter Strombegrenzungswert

## 4.10.4 Übertemperatursicherung

Überschreitet die MOSFET-Temperatur der externen Controller oder die Spulentemperatur der Antriebe mit integriertem Controller einen vorgegebenen Grenzwert um mehr als eine Sekunde, wird der Motor abgeschaltet.

Über die FAULHABER Error Mask (0x2321) kann die weitere Reaktion auf einen Übertemperaturfehler eingestellt werden (EMCY, Fault-Zustand oder Fehlerausgang).

Um den Motor wieder zu aktivieren, müssen folgende Bedingungen erfüllt sein:

- Temperatur unterhalb eines vorgegebenen Grenzwerts
- Solldrehzahl auf  $0 \text{ min}^{-1}$  eingestellt
- Tatsächliche Motordrehzahl  $< 50 \text{ min}^{-1}$

**i** Bei der Bestimmung der Spulentemperatur wird die Gehäusetemperatur gemessen und über die Strommessung auf die Verlustleistung geschlossen. Über ein thermisches Modell wird aus diesen Größen die MOSFET- bzw. Spulentemperatur berechnet. In den meisten Anwendungsfällen stellt diese Methode einen thermischen Motorschutz dar.



## Funktionsbeschreibung

### 4.10.5 Unterspannungsüberwachung

Unterschreitet die Versorgungsspannung die untere Spannungsschwelle, wird die Endstufe abgeschaltet. Der Motion Controller bleibt weiter aktiv. Liegt die Spannung wieder im zulässigen Bereich, wird die Endstufe sofort wieder eingeschaltet.


### 4.10.6 Überspannungsregelung

Wird der Motor generatorisch angetrieben, erzeugt er Energie. Üblicherweise sind Netzgeräte nicht in der Lage, diese Energie in das Netz zurückzuspeisen. Aus diesem Grund steigt die Versorgungsspannung am Motor und je nach Drehzahl kann es zur Überschreitung der zulässigen Höchstspannung kommen.

Um eine Zerstörung von Bauteilen zu vermeiden, enthalten die FAULHABER Motion Controller für bürstenlose Motoren einen Regler, der beim Überschreiten einer Grenzspannung (32 V) den Polradwinkel verstellt. Die Motion Controller für DC-Motoren enthalten eine Ballastschaltung, die beim Überschreiten einer Grenzspannung (32 V) aktiviert wird. Dadurch wird die erzeugte Energie im Motor umgesetzt und die Spannung der Elektronik bleibt auf 32 V begrenzt. Diese Methode schützt den Antrieb bei generatorischem Betrieb und schnellem Bremsen.

### 4.10.7 Einstellung der Reglerparameter

Um den Regler optimal auf die jeweilige Anwendung anzupassen, müssen die voreingestellten Reglerparameter optimiert werden.

 Der digitale Regler arbeitet mit einer Abtastrate von 200  $\mu\text{s}$ , bei MCDC und MCBL mit 100  $\mu\text{s}$ . Die Abtastrate kann bei Bedarf über den Befehl Sampling Rate (0x2330.01) auf bis zu 2 ms erhöht werden.

#### 4.10.7.1 Standardverhalten

Ohne weitere Einstellungen ist für den Drehzahlregler im Profile Velocity Mode die im Parameter Proportional Term POR eingestellte Verstärkung wirksam.

Im Profile Position Mode wird innerhalb des Zielkorridors die über den Parameter Proportional Term POR eingestellte Verstärkung um den Wert des Parameters Derivative Term PD erhöht. Dadurch kann ein schnelleres Einregeln auf den Stillstand in der Zielposition erreicht werden, ohne den Regler bei den Übergangsvorgängen selbst zu sehr anzuregen. Der Parameter PD muss dazu sorgfältig eingestellt werden und sollte typisch maximal 50% des Basiswerts POR betragen. Andernfalls besteht die Gefahr der Instabilität.

## Funktionsbeschreibung

### 4.10.7.2 Zur Verfügung stehende Reglerparameter

#### Filter Settings (0x2330)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2330	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Sampling Rate	U16	rw	1	Faktor der Abtastrate
	0x02	Gain Scheduling	U16	rw	0	1: Reduzierte Reglerverstärkung im Zielkorridor bei Positionierung

#### Gain Scheduling

Bei aktiviertem Gain Scheduling (0x2330.02) wird die Reglerverstärkung POR im Positionierbetrieb sukzessive reduziert, sobald sich der Antrieb im Zielkorridor (0x6067) befindet. Dadurch kann ein wesentlich „ruhigerer“ Stillstand in der Zielposition erreicht werden. Sobald der Antrieb den Zielkorridor wieder verlässt, wird POR sofort wieder auf den eingestellten Wert erhöht.

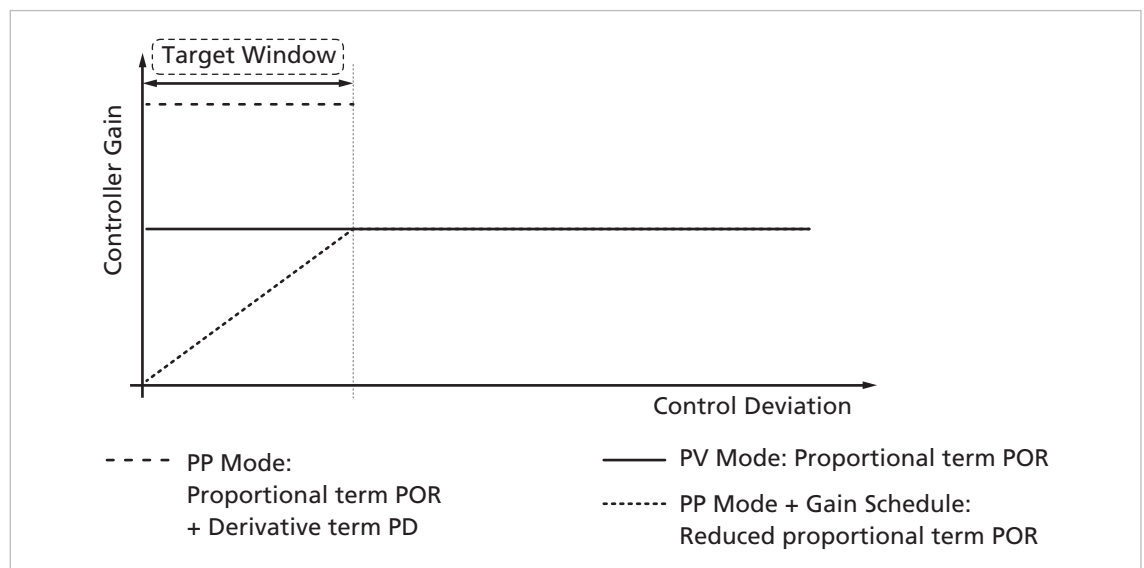


Abb. 30: Reglerverstärkung mit aktiviertem Gain Scheduling

**i** Die Funktion Gain Scheduling wird nur bei Abtastraten mit einem Faktor größer 3 aktiv (Sampling rate > 3).

#### Velocity Control Parameter Set (0x2331)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2331	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Proportional Term POR	U16	rw	a)	Proportionalverstärkung des Drehzahlreglers
	0x02	Integral Term I	U16	rw	a)	Integralanteil des Drehzahlreglers

a) Abhängig von der Konfiguration des Motion Controllers

## Funktionsbeschreibung

### Position Control Parameter Set (0x2332)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2332	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Proportional Term PP	U16	rw	a)	Proportionalverstärkung des Positionsreglers
	0x02	Derivative Term PD	U16	rw	a)	Differenzialanteil des Positionsreglers

a) Abhängig von der Konfiguration des Motion Controllers

Bei integrierten Einheiten sind diese Werte bereits voreingestellt, können allerdings über den Motorassistenten des Motion Managers noch an die zu treibende Last angepasst werden. Für externe Steuerungen werden diese Werte durch Auswahl eines Motortyps im Motorassistenten des Motion Managers passend vorbelegt.

Mit Hilfe des Tools **Regler-Tuning** im Motion Manager können einige Reglerparameter weiter justiert werden, um den Regler optimal an die jeweilige Anwendung anzupassen.

#### 4.10.7.3 Mögliche Vorgehensweise

1. Standardeinstellungen des Motorassistenten durchführen.
2. Drehzahlregler optimieren:
  - Mit dem Tool **Regler-Tuning** Drehzahlsprünge zwischen 1/3 und 2/3 der Maximaldrehzahl durchführen und dabei die Reglerverstärkung POR schrittweise erhöhen, bis der Regler instabil wird.
  - Reglerverstärkung wieder verringern, bis eine sichere Stabilität gegeben ist.

Unter Umständen kann es notwendig sein, den Integralanteil I entsprechend zu optimieren.
3. Positionsregler optimieren:
  - Mit dem Tool **Regler-Tuning** der Anwendung entsprechende Bewegungsprofile vorgeben.
  - Wenn das System mit diesen Einstellungen nicht stabil funktioniert, kann durch Verringern des I-Anteils des Drehzahlreglers oder Verringern des P-Anteils des Positionsreglers Stabilität erreicht werden.
  - Den P-Anteil des Positionsreglers schrittweise bis an die Stabilitätsgrenze des Systems erhöhen.
  - Die Stabilität entweder durch Erhöhung des D-Anteils des Positionsreglers oder durch Verringern des I-Anteils des Drehzahlreglers wieder herstellen.

## 5 Inbetriebnahme

Zur Inbetriebnahme muss die Antriebseinheit über einen CAN-Adapter an einen PC oder an eine übergeordnete Steuerung mit CAN-Schnittstelle angeschlossen werden, um die Grundeinstellungen durchführen zu können.

**i** Der Anschluss der CAN-Schnittstelle ist im Gerätehandbuch beschrieben. Für den Kommunikationsaufbau muss bei allen Teilnehmern die gleiche Übertragungsrate eingestellt werden (siehe Kap. 5.1, S. 84) und die Abschlusswiderstände müssen installiert sein.

### 5.1 Kommunikationseinstellungen

Die FAULHABER-Antriebe werden standardmäßig ohne gültige Knotenadresse (Node-ID = 255) und mit eingestellter automatischer Baudratenerkennung (AutoBaud) ausgeliefert.

**i** Im Netzwerkbetrieb sollte die verwendete Netzwerk-Übertragungsrate fest eingestellt werden.

#### 5.1.1 Einstellung über das CAN-Netzwerk

Für die Einstellung über das CAN-Netzwerk wird der FAULHABER Motion Manager oder ein anderes Konfigurationstool, das das LSS-Protokoll (Layer Setting Service and Protocol) nach CiA 305 unterstützt, benötigt.

**i** Der FAULHABER Motion Manager muss auf einem PC mit unterstütztem CAN-Interface installiert sein.

Es gibt zwei Möglichkeiten zur Einstellung der Kommunikationsparameter:

- Ein einzelner Antrieb ist an der CAN-Schnittstelle des Konfigurations-Tools angeschlossen:

Über den „LSS Switch Mode Global“ wird der Antrieb ohne weitere Angaben in den Konfigurationsmodus versetzt, um Knotennummer und Baudrate einzustellen.

- Der zu konfigurierende Antrieb ist innerhalb eines Netzwerks über die CAN-Schnittstelle an das Konfigurations-Tool angeschlossen:

Über den „LSS Switch Mode Selective“ wird der gewünschte Antrieb durch Eingabe der LSS-Adresse (Vendor-ID, Productcode, Revision Number, Seriennummer) adressiert und in den Konfigurationsmodus versetzt, um Knotennummer und Baudrate einzustellen.

Die FAULHABER-Antriebe der Serie MC V2.5 benötigen folgende Eingaben:

- Vendor-ID: 327
- Produktcode: 3150
- Seriennummer: Siehe Produkt-Aufkleber

Für die Revision-Number kann immer 0.0 übergeben werden, da dieser Wert im Protokoll ignoriert wird.

## Inbetriebnahme

Das LSS-Protokoll unterstützt neben der Einstellung von Knotennummer und Baudrate auch das Auslesen der LSS-Adressen von angeschlossenen Einheiten und das Auslesen der eingestellten Node ID.

Zur LSS-Kommunikation werden die Identifier 0x7E5 (vom Master) und 0x7E4 (vom Slave) verwendet.

Der Motion Controller speichert nach der Konfiguration die eingestellten Parameter in das EEPROM. Sie sind nach Aus- und Einschalten wieder verfügbar.

Für eine detaillierte Beschreibung des LSS-Protokolls wird auf das Dokument CiA 305 verwiesen.

### 5.1.1.1 Einstellung der Knotennummer

- Knotennummern 1 bis 127 können eingestellt werden.
- Die Node ID 255 (0xFF) kennzeichnet den Knoten als nicht konfiguriert. Der Knoten befindet sich nach dem Einschalten im LSS-Init-Status, bis eine gültige Knotennummer übermittelt wird. Nachdem eine gültige Knotennummer an den Knoten übermittelt wurde, wird die NMT-Initialisierung fortgesetzt.

### 5.1.1.2 Einstellung der Baudrate

- Bei aktivierter automatischer Baudratenerkennung (AutoBaud) kann der Antrieb in ein Netzwerk mit beliebiger Übertragungsrate gemäß Tab. 20 eingesetzt werden. Nach spätestens 24 Telegrammen (3 pro Baudrate) auf der Busleitung ist die Baudrate des Netzwerks detektiert. Der Antrieb stellt sich entsprechend der Baudrate des Netzwerks ein.
- Wenn die automatische Baudratenerkennung aktiv ist, können Telegramme solange nicht verarbeitet werden, bis die Baudrate detektiert wurde. Das Hochfahren des Systems dauert bei automatischer Baudratenerkennung entsprechend länger.
- Eine feste Baudrate gemäß Tab. 20 kann durch Angabe des Index 0 bis 4 eingestellt werden.

Tab. 20: Bit-Timing-Parameter

Baudrate	Index
1 000 kBit/s	00
800 kBit/s	01
500 kBit/s	02
250 kBit/s	03
125 kBit/s	04
AutoBaud	255

### 5.1.2 Knotennummer auslesen

Über das Objekt 0x2400.00 kann die aktuelle Einstellung der Baudrate (AutoBaud oder feste Baudrate) ausgelesen werden.

Tab. 21: CAN Baudrate Index und Knotennummer

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2400	0x00	Baudrate Set	U8	ro	0xFF	Eingestellte Baudrate

## 5.2 Grundeinstellungen

Bei den externen Motion Controllern müssen bei der ersten Inbetriebnahme einige Grundeinstellungen vorgenommen werden, um den Controller an den angeschlossenen Motor anzupassen.



### HINWEIS!

#### Zerstörung von Komponenten

- ▶ Die nachfolgend beschriebenen Grundeinstellungen durchführen.

Folgende Grundeinstellungen sind bei externen Motion Controllern notwendig:

- Motortyp bzw. Motordaten des angeschlossenen Motors
- Auflösung eines externen Encoders, falls verwendet
- Strombegrenzungswerte, angepasst an Motortyp und Anwendung
- Reglerparameter, angepasst an Motortyp und Anwendung

Zusätzlich kann mit Hilfe des FAULHABER Motion Managers noch ein Abgleich der Hallsensoren signale für ein besseres Antriebsverhalten vorgenommen werden.

Bei allen Motion Controllern (integriert und extern) muss die Konfiguration noch an die jeweilige Anwendung angepasst werden. Insbesondere sind folgende Grundeinstellungen wichtig:

- Betriebsart
- Strombegrenzungswerte
- Reglerparameter
- Funktion der digitalen Ein-/Ausgänge



### HINWEIS!

#### Zerstörung von Komponenten

- ▶ Bei Verwendung des Fault-Pins als Eingang zuerst die gewünschte Funktion programmieren, bevor von außen Spannung angelegt wird.

Im folgenden Kapitel wird die Konfiguration dieser Parameter mit Hilfe des FAULHABER Motion Managers näher erläutert.

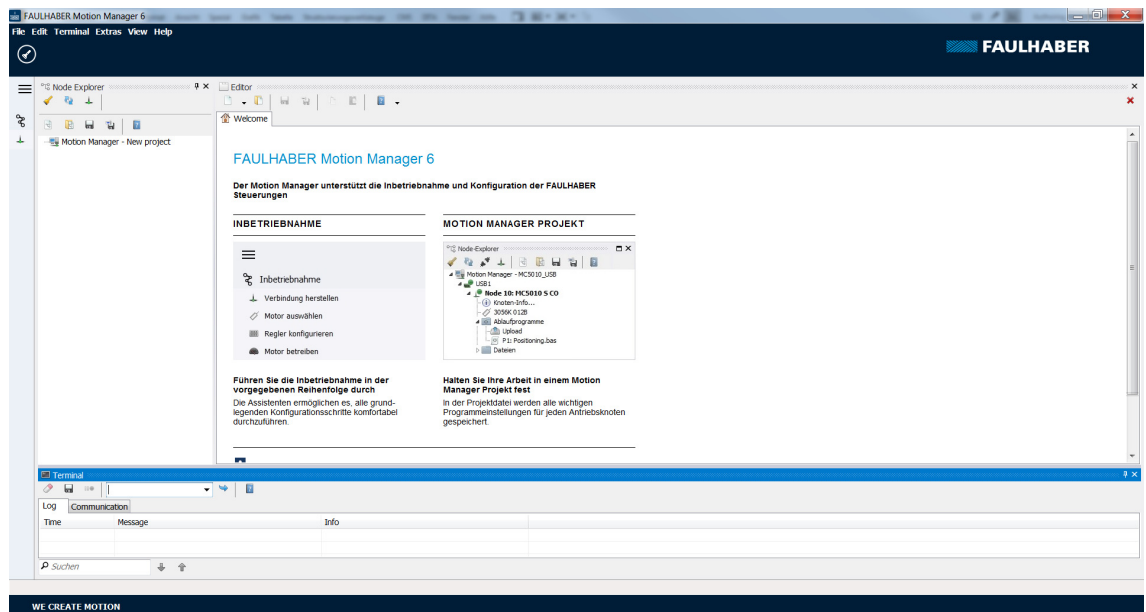
## Inbetriebnahme

### 5.3 Konfiguration mit dem Motion Manager

Die PC-Software FAULHABER Motion Manager bietet eine einfache Möglichkeit, die Antriebseinheit zu konfigurieren und erste Tests und Optimierungen durchzuführen.

Die Software ist für Microsoft Windows verfügbar und kann kostenlos von der FAULHABER Internet-Seite unter [www.faulhaber.com/motionmanager](http://www.faulhaber.com/motionmanager)

heruntergeladen werden.



Motion Controller mit angeschlossenem Motor müssen vor der Inbetriebnahme mit für den Motor geeigneten Werten für die Strombegrenzung und geeigneten Reglerparametern versehen werden.

Zur Auswahl des Motors und der dafür geeigneten Grundparameter steht der Assistent für Motorauswahl zur Verfügung.

Weitere Einstellungen z. B. zur Funktion des Fault Pins können unter dem Menüpunkt **Konfiguration – Antriebsfunktionen** über einen komfortablen Dialog vorgenommen werden (siehe Kap. 5.3.3, S. 89). Der Konfigurationsdialog steht auch in der Schnellzugriffleiste des Motion Managers zur Verfügung.

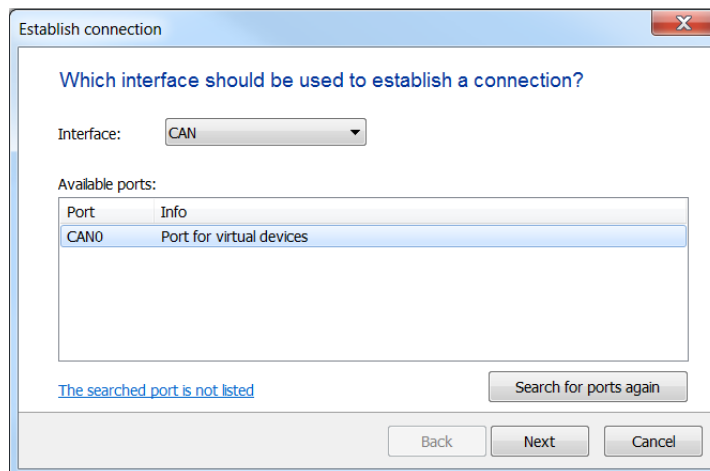
Zusätzlich steht das Tool **Regler-Tuning** zur Verfügung, mit dem die Reglerparameter des Drehzahl- und des Positionsreglers an die Anwendung angepasst werden können.

## Inbetriebnahme

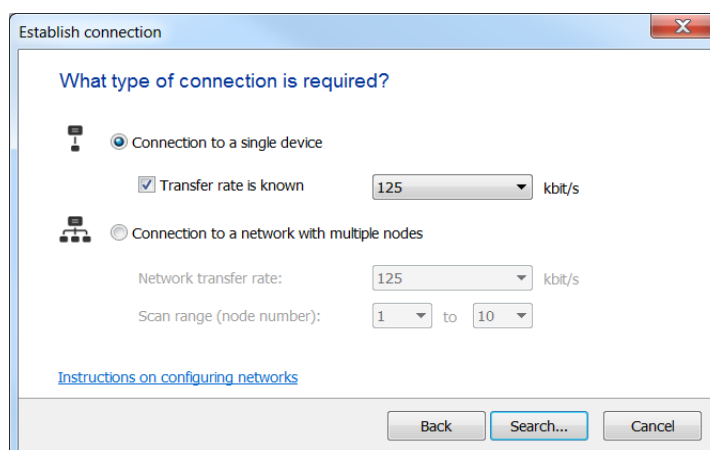
### 5.3.1 Verbindung herstellen

Um mit der vorliegenden Steuerung kommunizieren zu können, muss eine Verbindung vom PC, auf dem der Motion Manager installiert ist, zum Motion Controller hergestellt werden.

Zum Einrichten der Kommunikationsverbindung über eine der unterstützten Schnittstellen steht ein Assistent bereit. Der Assistent erscheint automatisch beim Erstellen eines neuen Projekts. Er ist jederzeit über die Schaltfläche **Verbindung herstellen** in der Schnellzugriff-leiste oder über das Menü **Terminal** aufrufbar.



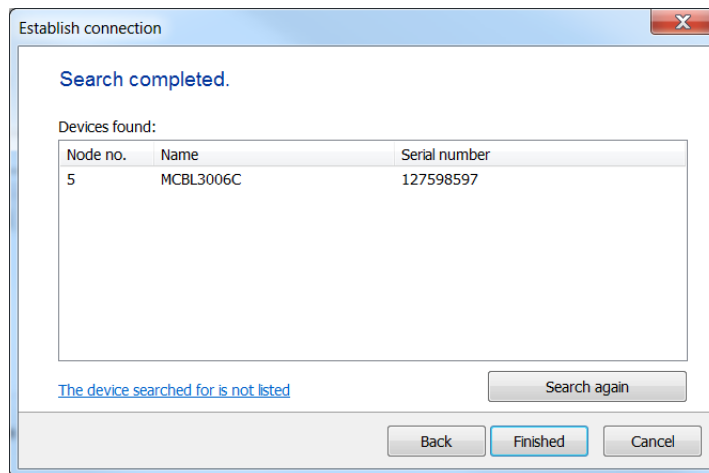
1. Schnittstelle **CAN** auswählen.
2. Unter **Verfügbare Ports** den gewünschten Schnittstellenanschluss eines installierten CAN-Adapters auswählen.
3. Schaltfläche **Weiter** betätigen.



4. Verbindungsart und Baudrate auswählen.  
Informationen zu den unterstützten CAN-Schnittstellen finden sich in der Softwareanleitung und können bei FAULHABER angefragt werden.
5. Schaltfläche **Suchen...** betätigen.  
Geräte, die bereits auf eine passende Baudrate eingestellt sind, werden vom Motion Manager gefunden und angezeigt.



## Inbetriebnahme



### 6. Schaltfläche **Fertig** betätigen.

Noch nicht konfigurierte Geräte erscheinen im Node-Explorer als *Unkonfigurierter LSS-Knoten*. Über **Konfiguration - Verbindungsparameter** kann dann eine gültige Knotennummer eingestellt werden.

### 5.3.2 Motor auswählen

Externe Motion Controller müssen an den angeschlossenen Motor angepasst werden. Hierfür steht der Assistent für Motorauswahl zur Verfügung, der über die Schnellzugriffleiste des Motion Managers oder über das Menü **Konfiguration** unter **Inbetriebnahme - Motor auswählen** aufgerufen werden kann.

Nach Auswahl des verwendeten FAULHABER-Motors aus einer Liste und Einstellung des verwendeten Sensortyps sowie der Eingabe eines Trägheitsfaktors für die zu betreibende Last, werden neben den Motor- und Strombegrenzungswerten auch passende Reglerparameter ermittelt und zum Antrieb übertragen.

Siehe Motion Manager Bedienungsanleitung für die Benutzung des Assistenten für Motorauswahl.

### 5.3.3 Antrieb konfigurieren

Vom Assistenten für Motorauswahl wurden bereits sinnvolle Standardeinstellungen für die ausgewählte Motor-/Sensorkombination eingestellt.

Für die weitere Antriebskonfiguration und die Anpassung an die gewünschte Anwendung steht ein Konfigurationsdialog mit mehreren Seiten in der Schnellzugriffleiste des Motion Managers oder unter dem Menüpunkt **Konfiguration – Antriebsfunktionen...** zur Verfügung.

Alle Einstellungen werden erst an den Antrieb übertragen wenn die Schaltfläche **Senden** betätigt wird. Dabei wird auch der aktuelle Zustand vom Antrieb zurückgelesen und der Dialog dementsprechend aktualisiert. Ungültige Kombinationen von Einstellungen werden dabei korrigiert, da sie vom Antrieb nicht akzeptiert werden.

Dauerhaft im Antrieb gespeichert werden die Einstellungen über die Schaltfläche **SAVE**.

## Inbetriebnahme

**i** Zusätzlich können Einstellungen über den Objekt-Browser durchgeführt werden (Menü **Konfiguration – Objekt-Browser**), der das gesamte Objektverzeichnis darstellt. Über den PDO-Mapping-Dialog kann die Aufteilung der PDO-Daten an die gewünschte Anwendung angepasst werden (Menü **Konfiguration – PDO-Mapping**).

### 5.3.3.1 Grundeinstellungen durchführen

Bei der Inbetriebnahme werden im Reiter **Grundeinstellungen** folgende Parameter eingestellt:

- Betriebsart
- Positionierbereichsgrenzen
- Drehzahl-Bereichsgrenze
- Geber-Einstellungen
- Kommutierungseinstellungen für BL-Motoren

#### **Betriebsarten**

Der FAULHABER Motion Controller unterstützt folgende Betriebsarten:

- Profile Position Mode (Positionsregelung)
- Profile Velocity Mode (Drehzahlregelung)
- Cyclic Synchronous Position Mode (Positionsregelung)
- Homing Mode (Referenzfahrt)

#### **Positionierbereichsgrenzen**

In verschiedenen Betriebsarten kann der Bewegungsbereich überwacht und begrenzt werden. Die Grenzen dieses Bewegungsbereichs können in der Einheit der Istposition angegeben werden.

Über das General Settings Objekt 0x2338.02 können die Bereichsgrenzen auch für den Profile Velocity Mode aktiviert werden. Standardmäßig ist die Überwachung nur im Profile Position Mode und im Homing Mode aktiv.

#### **Gebertyp und Optimierung**

Falls ein an den Motor angebaute Inkrementalencoder ausgewertet werden soll, muss dessen effektive Auflösung bei 4-Flankenauswertung angegeben werden. Bei Verwendung des internen Encoders sind keine weiteren Eingaben notwendig.

Zur Anpassung der Hallsensorsignale auf den angeschlossenen Motor steht für extern angeschlossene BL-Motoren mit analogen Hallsensoren eine Schaltfläche zur Verfügung, über die ein Assistent zur Optimierung auf den angeschlossenen Motor gestartet werden kann.


**i** Vor dem Start der Geberoptimierung sicherstellen, dass sich der Motor frei drehen kann.

#### **Kommutierungseinstellung für BL-Motoren**

Als Standard verwendet der Motion Controller bei BL-Motoren eine reine Sinuskommutierung. Der Motor wird dadurch mit möglichst geringen Verlusten und geräuscharm betrieben.

Alternativ kann bei höheren Drehzahlen über das General Settings Objekt 0x2338.01 auch eine der Blockkommutierung ähnliche Übersteuerung der Ausgangssignale zugelassen werden. Der gesamte Drehzahlbereich des Antriebs kann dadurch genutzt werden.

## Inbetriebnahme

 Beim Wechsel zwischen der reinen Sinuskommutierung und dem Betrieb mit Blockkommutierung im oberen Drehzahlbereich wird die Reglerverstärkung ebenfalls entsprechend erhöht.

### 5.3.3.2 Factor Group

Im Reiter **Factor Group** können Umrechnungsfaktoren von internen auf benutzerdefinierte Positionswerte angegeben werden.

#### Umrechnungsfaktoren für Positionswerte

Die meisten Positionsangaben werden in Benutzereinheiten vorgegeben und angezeigt. Zur Umrechnung von internen Einheiten in Benutzereinheiten stehen die Objekte der Factor Group zur Verfügung. Damit können Faktoren für eine gegebene Getriebeuntersetzung und für einen Vorschubwert angegeben werden. Abhängig von der jeweiligen Geberauflösung wird daraus der Umrechnungsfaktor für Positionswerte bestimmt.

### 5.3.3.3 Reglerparameter einstellen

Im Reiter **Drehzahlregler** und **Positionsregler** des Antriebskonfigurationsdialogs können Änderungen an den standardmäßig eingestellten Regler- und Strombegrenzungsparametern vorgenommen werden.

Zusätzlich steht unter dem Menüpunkt **Konfiguration – Reglerparameter...** noch ein weiterer Dialog zur Verfügung, in dem die Parameter online verändert werden können und das Ergebnis direkt beobachtet bzw. über die Trace-Funktion im Motion Manager aufgezeichnet werden kann.

#### Stromregler

Über den Parameter *Continuous Current Limit* kann der thermisch zulässige Dauerstrom für die Anwendung angegeben werden.

Motoren und auch der Motion Controller sind in gewissen Grenzen überlastbar. Für dynamische Vorgänge können daher auch höhere Ströme zugelassen werden. Der maximale Stromwert wird über den Parameter *Peak Current* angegeben.

Abhängig von der Belastung des Antriebs begrenzt die interne Stromüberwachung den Ausgangsstrom auf den Spitzenstrom (*Peak Current Limit*) oder den zulässigen Dauerstrom (*Continuous Current*).



#### HINWEIS!

##### Zerstörung des Motors

- ▶ Den thermisch zulässigen Dauerstrom (*Continuous Current Limit*) **nicht** über dem thermisch zulässigen Dauerstrom des Motors entsprechend dem Datenblatt einstellen.
- ▶ Den maximalen Spitzenstrom (*Peak Current Limit*) **nicht** über dem maximalen Spitzenausgangsstrom der vorhandenen Elektronik einstellen.

Der Stromregler des Motion Controller arbeitet als Strombegrenzungsregler und hat damit im unbegrenzten Fall keine Auswirkung auf die Dynamik der Drehzahlregelung. Über den Parameter *CI* kann die Geschwindigkeit der Begrenzung eingestellt werden. Bei Verwendung der Standardwerte für den verwendeten Motor wird der Strom nach etwa 5 ms auf den zulässigen Wert begrenzt.

Wurde über den Motorassistenten ein FAULHABER Motor gewählt, sind bereits Parameter eingetragen mit denen der Motor sicher betrieben werden kann.

Weitere Angaben siehe Kap. 4.10.3, S. 79.

## Inbetriebnahme

### Drehzahlregler

Der Drehzahlregler ist als PI-Regler ausgeführt. Folgende Parameter können eingestellt werden:

- Abtastzeit (Sampling Rate) in Vielfachen der Grundabtastrate des Antriebs
- Proportionalverstärkung POR
- Integralanteil I

Wurde über den Motorassistenten ein FAULHABER Motor gewählt, sind bereits Parameter eingetragen mit denen der Motor sicher betrieben werden kann.

Bei zusätzlichen Lasten am Motor muss die Trägheit der Last durch einen höheren Proportionalanteil und ggf. durch eine langsamere Abtastung kompensiert werden. Der Integralanteil kann in den meisten Anwendungen unverändert bleiben.

Weitere Angaben siehe Kap. 4.10.7, S. 81.

### Rampengenerator

Der Rampengenerator begrenzt die Drehzahländerung am Eingang des Drehzahlreglers über die Parameter Profile Acceleration und Profile Deceleration und die maximale Vorgabedrehzahl über den Parameter Profile Velocity.

Die Parameter Profile Acceleration und Profile Velocity sind entsprechend der Anwendung frei wählbar. Über den Parameter Profile Deceleration wird das Bremsverhalten im Positionierbetrieb festgelegt. Für große Lasten muss die Bremsrampe über den Parameter Profile Deceleration begrenzt werden, um ein überschwingungsfreies Einlaufen in die Zielposition zu erreichen.

Weitere Angaben siehe Kap. 4.10.1, S. 75.

### Positionsregler

Der Positionsregler ist als Proportionalregler ausgeführt. Nur innerhalb des Zielkorridors wirkt zusätzlich noch ein D-Anteil.

Der Proportionalanteil errechnet aus der Positionsabweichung in Inkrementen die maximale Vorgabedrehzahl für den unterlagerten Drehzahlregler. Über den Rampengenerator werden die Beschleunigung und die Maximaldrehzahl zusätzlich begrenzt.

Überschwingungsfreies Einlaufen in die Zielposition kann bevorzugt über eine Anpassung der Bremsrampe an die Last erreicht werden. Für gut gedämpftes Einschwingen in der Endposition muss der Parameter PP proportional zur Lastträgheit reduziert werden.

Weitere Angaben siehe Kap. 4.10.7, S. 81.

### Zielkorridor

Die Parameter Position Window und Position Window Time definieren einen Bereich um die Zielposition und eine Verweilzeit in diesem Bereich, bis das Bit „Target Reached“ im Statusword gesetzt wird. Wenn für das PDO, das das Statusword enthält, der Transmission Type auf 255 gesetzt ist (Standardeinstellung), wird die Zielposition über ein asynchron abgesetztes PDO signalisiert. Innerhalb dieses Korridors ist der D-Anteil des Positionsreglers aktiv und der Rampengenerator inaktiv.

## Inbetriebnahme

### 5.3.3.4 Fehlerbehandlung

Im Reiter **Fehlerbehandlung** des Antriebskonfigurationsdialogs kann die Reaktion des Motion Controllers auf bestimmte Fehlerereignisse eingestellt werden.

Grundsätzlich können Fehler über ein Emergency-Telegramm und über den Fehlerausgang signalisiert werden. Außerdem kann eingestellt werden, dass der Antrieb bei bestimmten Fehlern in den DSP402-Fehlerzustand gehen soll.

Der Parameter Error Delay gibt an, wie lange ein Überstrom-, Überspannungs- oder Drehzahlabweichungsfehler anstehen muss, bis er gemeldet wird.

#### Maximal zulässige Drehzahlabweichung

Im Error Handling Objekt kann über den Parameter Deviation eine maximal zulässige Drehzahlabweichung für den Drehzahlregler vorgegeben werden. Wird diese Schranke länger überschritten als mit dem Parameter Error Delay eingestellt, wird je nach Einstellung im Objekt Error Mask ein Fehler über den Fault-Pin oder über eine CANopen Emergency Message signalisiert.

### 5.3.3.5 Ein-/Ausgänge und Homing einstellen

Im Reiter **Ein-/Ausgänge und Homing** des Antriebskonfigurationsdialogs können die Funktionen der digitalen Ein- und Ausgänge festgelegt und Einstellungen für eine Referenzfahrt definiert werden.

#### Eingangspiegel und Flanke

Die Schaltschwellen der digitalen Eingänge sind entweder direkt 5V-TTL kompatibel oder an die Schaltpegel von 24 V SPS Ausgängen angepasst. Die Einstellung erfolgt über den Parameter Input Threshold Level.

Genaue Angaben zu den Schaltschwellen sind im Datenblatt des Antriebs zu finden.

#### Funktion des Fault Pins

Der Fault Pin kann sowohl als Eingang als auch als Ausgang betrieben werden. Die gewünschte Funktion kann über die Optionsschaltflächen **Fault-Pin Function** ausgewählt werden.



#### HINWEIS!

##### Zerstörung des Motors

- ▶ Wenn der Fault Pin als Digitalausgang (ERRROUT / DIGOUT) konfiguriert ist, **keine** Spannung an den Fault Pin anschließen.

Für die Standardfunktion als Fehlerausgang kann über den Parameter Error Delay eine Verzögerungszeit angegeben werden, um die Reaktion auf einzelne kurze Überstromimpulse zu unterdrücken.

In der Funktion Position Output zeigt der Ausgang das Erreichen der Zielposition als Digitalsignal an (Low entspricht „Zielposition ist erreicht“).

In der Funktion Digital Output kann der Ausgang von einer übergeordneten Steuerung über den Parameter Digital Output Status (0x2315.03) gesetzt und gelöscht werden.

In der Funktion Reference Input kann der Fault Pin als Referenzeingang zum Anschluss eines Homing- oder Referenzschalters verwendet werden.

## Inbetriebnahme

---

### Funktion der digitalen Eingänge

Über die angezeigte Eingangsmatrix kann die Verwendung der verfügbaren Digitaleingänge eingestellt werden:

- Endschalter sind Begrenzungsschalter, die bei Aktivierung die jeweilige Bewegungsrichtung blockieren. Über das Polarity-Bit kann eingestellt werden, ob die positive oder die negative Flanke für die Aktivierung gültig sein soll.
- Homing-Schalter sind Referenzschalter zum Nullen der Position mit bestimmten Homing-Methoden. Die Polarität ist hier durch die ausgewählte Homing-Methode vorgegeben.
- Endschalter können ebenfalls wie Homing-Schalter zum Nullen der Position mit bestimmten Homing-Methoden verwendet werden. Die Polarität ist auch hier durch die ausgewählte Homing-Methode vorgegeben. Wenn abweichend zu DSP402 die Polarität geändert werden soll, kann dies über das Polarity-Bit eingestellt werden, wenn zusätzlich die Funktion **Polarity für Homing Mode Endschalter verwenden** aktiviert ist.

### 5.3.4 Datensatz verwalten

#### Parameter sichern

Die Einstellungen eines Antriebs können als Backup oder für die Konfiguration weiterer Antriebe als Datei gespeichert werden.

Der Motion Manager bietet die Möglichkeit, die aktuelle Antriebskonfiguration über den Objekt-Browser auszulesen und als XDC-Datei (XML Device Configuration File) zu speichern.

#### Parameter an den Antrieb übertragen

Im Motion Manager können zuvor gespeicherte XDC-Dateien im CANopen Objekt-Browser geöffnet, bei Bedarf editiert und zum Antrieb übertragen werden.



Den Befehl `SAVE` ausführen, um einen übertragenen Parametersatz dauerhaft im Antrieb zu speichern.

## Inbetriebnahme

---

### 5.3.5 Diagnose

#### 5.3.5.1 Statusanzeige

Die Statusanzeige dient zur laufenden Kontrolle der wesentlichen Betriebszustände.

Interne Zustände, Fehlerflags und der Zustand der digitalen Eingänge werden signalisiert. Zusätzlich werden die intern gemessene Gehäusetemperatur, das Statusword und weitere Istwerte angezeigt.

Die Anzeige wird vom Motion Manager über eine zyklische Abfrage der internen Zustände aktualisiert.

##### Interne Zustände

Folgende teilautonome Zustände des Motion Controllers werden angezeigt:

- Fehlerflags
- Gehäusetemperatur
- Zustände der digitalen Eingänge

Das Flag „Strombegrenzung“ ist gesetzt, wenn der Maximalstrom durch die  $I^2t$ -Überwachung auf den Dauerstrom (*Continuous Current Limit*) gesetzt worden war.

##### Gerätezustand (DSP402)

Die einzelnen Bits des DSP402 Statusword, die aktuelle Istposition und die aktuelle Istdrehzahl werden angezeigt.

#### 5.3.5.2 Trace-Funktion

Als weiteres Diagnose-Werkzeug stellt der Motion Manager das Tool **Grafische Analyse** zur Verfügung, über die interne Parameter grafisch aufgezeichnet werden können. Damit lässt sich das dynamische Verhalten des Antriebs überwachen, was z. B. für die Optimierung der Reglerparameter hilfreich ist.

# Parameterbeschreibung

## 6 Parameterbeschreibung

### 6.1 Kommunikationsobjekte nach CiA 301

#### Device Type

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1000	0x00	Device Type	U32	ro	0x00420192	Angabe des Gerätetyps

Enthält Informationen zum Gerätetyp, aufgeteilt in zwei 16-Bit-Feldern:

- Byte MSB (Most Significant Byte): Additional Information = 0x42 (Servo drive, type specific PDO mapping)
- Byte LSB (Least Significant Byte): Device Profile Number = 0x192 (402d)

#### Error Register

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1001	0x00	Error Register	U8	ro	ja	Fehlerregister

Das Error Register beinhaltet bitcodiert die zuletzt aufgetretenen Fehlerarten.

Für die Beschreibung des Error Registers siehe Kap. 3.6, S. 26.

#### Predefined Error Field (Fehlerspeicher)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1003	0x00	Number of Errors	U8	rw	–	Anzahl gespeicherter Fehler
	0x01	Error Field	U32	ro	–	Letzter Fehler
	0x02	Error Field	U32	ro	–	Vorletzter Fehler

Der Fehlerspeicher enthält die Codierung der zuletzt aufgetretenen Fehler.

- Byte MSB: Error Code, FAULHABER Error
- Byte LSB: Error Code, FAULHABER Error

Die Bedeutung der Fehlercodes ist in Kap. 3.6, S. 26 beschrieben.

Durch Schreiben einer 0 auf Subindex 0 wird der Fehlerspeicher gelöscht.

#### COB-ID SYNC

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1005	0x00	COB ID SYNC	U32	rw	0x80	CAN-Objekt-Identifizier des SYNC-Objekts

#### Manufacturer Device Name

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1008	0x00	Manufacturer Device Name	Vis-String	const	–	Gerätename

Zum Auslesen des Manufacturer's Device Name muss das Segmented-SDO-Protokoll verwendet werden.



## Parameterbeschreibung

### Manufacturer Hardware Version

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1009	0x00	Manufacturer Hardware Version	Vis-String	const	–	Hardwareversion

Zum Auslesen der Manufacturer Hardware Version wird das Segmented-SDO-Protokoll verwendet werden.

### Manufacturer Software Version

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x100A	0x00	Manufacturer Software Version	Vis-String	const	–	Softwareversion

Zum Auslesen der Manufacturer-Software-Version muss das Segmented-SDO-Protokoll verwendet werden.

### Guard Time

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x100C	0x00	Guard Time	U16	rw	0	Überwachungszeit für Node-Guarding

Angabe der Guard Time in Millisekunden. Der Wert 0 schaltet das Node-Guarding aus.

### Life Time Factor

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x100D	0x00	Life Time Factor	U8	rw	0	Zeitfaktor für Node-Guarding

Der Life Time Factor multipliziert mit der Guard Time ergibt die Life Time für das Node Guarding (siehe Kap. 3.8, S. 29). Der Wert 0 schaltet das Node Guarding aus.

### Store Parameters

Tab. 22: Parameter speichern

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1010	0x00	Number of Entries	U8	ro	9	Anzahl Objekteinträge
	0x01	Save All Parameters	U32	rw	1	Speichert alle Parameter
	0x02	Save Comm Parameters	U32	rw	1	Kommunikationsparameter speichern (Objektverzeichnis-Einträge 0x0000 bis 0x1FFF)
	0x03	Save App Parameters	U32	rw	1	Anwendungsparameter speichern (Objektverzeichnis-Einträge 0x2000 bis 0x6FFF)

Das Objekt Store Parameters speichert Konfigurationsparameter in den Flash-Speicher. Ein Lesezugriff liefert Informationen über die Speichermöglichkeiten. Das Schreiben der Signatur "save" auf den entsprechenden Subindex leitet den Speichervorgang ein.

Tab. 23: Signatur "save"

Signature	ISO 8 859 ("ASCII")	hex
MSB	e	65h
	v	76h
	a	61h
LSB	s	73h

## Parameterbeschreibung



### HINWEIS!

Der Flash-Speicher ist für 10 000 Schreibzyklen ausgelegt. Wird dieser Befehl mehr als 10 000 mal ausgeführt, ist die Funktion des Flash-Speichers nicht mehr gewährleistet.

- ▶ Häufiges Speichern vermeiden.
- ▶ Nach 10 000 Speicherzyklen Gerät wechseln.

### Restore Default Parameters

Tab. 24: Wiederherstellen von Parametern

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1011	0x00	Number of Entries	U8	ro	6	Anzahl Objekteinträge
	0x01	Restore all Default Parameters	U32	rw	1	Alle Werkseinstellungen wiederherstellen
	0x02	Restore Comm Default Parameters	U32	rw	1	Werkseinstellungen für Kommunikations-Parameter (0x0000 bis 0x1FFF) wiederherstellen
	0x03	Restore App Default Parameters	U32	rw	1	Werkseinstellungen für Anwendungs-Parameter (ab 0x2000) wiederherstellen
	0x04	Reload User Parameters	U32	rw	1	Vom Benutzer zuletzt gespeicherte Anwendungsparameter (ab 0x2000) wiederherstellen
	0x05	Reload Application Parameters 1	U32	rw	1	Anwendungsparametersatz 1 für den direkten Wechsel
	0x06	Reload Application Parameters 2	U32	rw	1	Anwendungsparametersatz 2 für den direkten Wechsel

Das Objekt Restore Default Parameters lädt Standardkonfigurationsparameter. Die Standardkonfigurationsparameter sind entweder der Auslieferungszustand oder der letzte gespeicherte Zustand. Ein Lesezugriff liefert Informationen über die Restoremöglichkeit. Das Schreiben der Signatur "load" auf den entsprechenden Subindex leitet den Restorevorgang ein:

Tab. 25: Signatur "load"

Signature	ISO 8859 ("ASCII")	hex
MSB	d	64h
	a	61h
	o	6Fh
LSB	l	6Ch



Der Auslieferungszustand darf nur bei abgeschalteter Endstufe geladen werden.

### COB-ID Emergency-Nachricht

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1014	0x00	COB-ID EMCY	U32	rw	0x80 + Node-ID	CAN-Objekt-Identifizier des Emergency Objects

## Parameterbeschreibung

### Consumer Heartbeat Time

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1016	0x00	Number of Entries	U8	ro	1	Anzahl Objekteinträge
	0x01	Consumer Heartbeat Time	U32	rw	0	Heartbeat-Überwachungszeit

- Bit 0 bis 15 enthalten die Consumer Heartbeat Time in Millisekunden. Bei einem Wert von 0 ist die Consumer-Heartbeat-Funktion deaktiviert.
- Bit 16 bis 23 enthalten die Knotennummer, an die die Heartbeat-Nachricht geschickt werden soll (Master Node ID).
- Bit 24 bis 31 sind unbenutzt (reserviert).

### Producer Heartbeat Time

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1017	0x00	Producer Heartbeat Time	U16	rw	0	Heartbeat Sende-Zeitintervall

Das Objekt Producer Heartbeat Time enthält das Producer-Heartbeat-Zeitintervall in Millisekunden. Bei einem Wert von 0 ist die Producer-Heartbeat-Funktion deaktiviert.

### Identity Object

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1018	0x00	Number of Entries	U8	ro	4	Anzahl Objekteinträge
	0x01	Vendor ID	U32	ro	327	Herstellernummer (FAULHABER: 327)
	0x02	Product Code	U32	ro	3 150	Produktkennnummer
	0x03	Revision Number	U32	ro	–	Versionsnummer
	0x04	Serial Number	U32	ro	–	Seriennummer

### Error Behaviour

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1029	0x00	Number of Entries	U8	ro	1	Anzahl Objekteinträge
	0x01	Communication Error	U8	rw	0	Verhalten bei Kommunikationsfehlern 0 = Zustand Pre-Operational 1 = Keine Zustandsänderung 2 = Zustand Stopped

Der Motion Controller wechselt bei schweren Kommunikationsfehlern in den NMT-Zustand *Pre-Operational*. Über den Subindex 1 kann das Verhalten bei schweren Kommunikationsfehlern geändert werden.

### Server SDO Parameter

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1200	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	COB ID Client to Server (rx)	U32	ro	0x600 + Node-ID	CAN-Objekt-Identifizier der Server RxSDO
	0x02	COB ID Server to Client (tx)	U32	ro	0x580 + Node-ID	CAN-Objekt-Identifizier der Server TxSDO

## Parameterbeschreibung

### Receive PDO1 Parameter

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1400	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	COB ID Used by RxPDO1	U32	rw	0x200 + Node-ID	CAN-Objekt-Identifizier der Server RxPDO1
	0x02	Transmission Type	U8	rw	255 (asynchr.)	PDO-Übertragungsart

### Receive PDO2 Parameter

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1401	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	COB ID Used by RxPDO2	U32	rw	0x300 + Node-ID	CAN-Objekt-Identifizier der Server RxPDO2
	0x02	Transmission Type	U8	rw	255 (asynchr.)	PDO-Übertragungsart

### Receive PDO3 Parameter

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1402	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	COB ID Used by RxPDO3	U32	rw	0x400 + Node-ID	CAN-Objekt-Identifizier der Server RxPDO3
	0x02	Transmission Type	U8	rw	255 (asynchr.)	PDO-Übertragungsart

### Receive PDO4 Parameter

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1403	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	COB ID Used by RxPDO4	U32	rw	0x500 + Node-ID	CAN-Objekt-Identifizier der Server RxPDO4
	0x02	Transmission Type	U8	rw	255 (asynchr.)	PDO-Übertragungsart

### Receive PDO1 Mapping

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1600	0x00	Number of Mapped Objects	U8	ro	1	Anzahl gemappter Objekte
	0x01	RxPDO1 Mapping Entry 1	U32	rw	0x60400010	Verweis auf 16-Bit Controlword (0x6040)
	0x02	RxPDO1 Mapping Entry 2	U32	rw	0	
	0x03	RxPDO1 Mapping Entry 3	U32	rw	0	
	0x04	RxPDO1 Mapping Entry 4	U32	rw	0	

## Parameterbeschreibung

### Receive PDO2 Mapping

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1601	0x00	Number of Mapped Objects	U8	ro	2	Anzahl gemappter Objekte
	0x01	RxPDO2 Mapping Entry 1	U32	rw	0x60400010	Verweis auf 16-Bit Controlword (0x6040)
	0x02	RxPDO2 Mapping Entry 2	U32	rw	0x607A0020	Verweis auf 32-Bit Target Position (0x607A)
	0x03	RxPDO2 Mapping Entry 3	U32	rw	0	
	0x04	RxPDO2 Mapping Entry 4	U32	rw	0	

### Receive PDO3 Mapping

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1602	0x00	Number of Mapped Objects	U8	ro	2	Anzahl gemappter Objekte
	0x01	RxPDO3 Mapping Entry 1	U32	rw	0x60400010	Verweis auf 16-Bit Controlword (0x6040)
	0x02	RxPDO3 Mapping Entry 2	U32	rw	0x60FF0020	Verweis auf 32-Bit Target Velocity (0x60FF)
	0x03	RxPDO3 Mapping Entry 3	U32	rw	0	
	0x04	RxPDO3 Mapping Entry 4	U32	rw	0	

### Receive PDO4 Mapping

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1603	0x00	Number of Mapped Objects	U8	ro	2	Anzahl gemappter Objekte
	0x01	RxPDO4 Mapping Entry 1	U32	rw	0x60400010	Verweis auf 16-Bit Controlword (0x6040)
	0x02	RxPDO4 Mapping Entry 2	U32	rw	0x257A0020	Verweis auf 32-Bit Target Position Internal Value (0x257A)
	0x03	RxPDO4 Mapping Entry 3	U32	rw	0	
	0x04	RxPDO4 Mapping Entry 4	U32	rw	0	

### Transmit PDO1 Parameter

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1800	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	COB ID Used by TxPDO1	U32	rw	0x180 + Node-ID	CAN-Objekt-Identifizier der TxPDO1
	0x02	Transmission Type	U8	rw	255 (asynchr.)	PDO-Übertragungsart
	0x05	Event Timer	U16	rw	0	Minimale Zeit zwischen TxPDO-Versendung <ul style="list-style-type: none"> <li>▪ 0: ausgeschaltet</li> <li>▪ 5...65 000: Zeit in Millisekunden</li> </ul>

## Parameterbeschreibung

### Transmit PDO2 Parameter

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1801	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	COB ID used by TxPDO2	U32	rw	0x280 + Node-ID	CAN-Objekt-Identifizier der TxPDO2
	0x02	Transmission Type	U8	rw	255 (asynchr.)	PDO-Übertragungsart
	0x05	Event Timer	U16	rw	0	Minimale Zeit zwischen TxPDO-Versendung <ul style="list-style-type: none"> <li>▪ 0: ausgeschaltet</li> <li>▪ 5...65 000: Zeit in Millisekunden</li> </ul>

### Transmit PDO3 Parameter

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1802	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	COB ID Used by TxPDO3	U32	rw	0x380 + Node-ID	CAN-Objekt-Identifizier der TxPDO3
	0x02	Transmission Type	U8	rw	255 (asynchr.)	PDO-Übertragungsart
	0x05	Event Timer	U16	rw	0	Minimale Zeit zwischen TxPDO-Versendung <ul style="list-style-type: none"> <li>▪ 0: ausgeschaltet</li> <li>▪ 5...65 000: Zeit in Millisekunden</li> </ul>

### Transmit PDO4 Parameter

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1803	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	COB ID Used by TxPDO4	U32	rw	0x480 + Node-ID	CAN-Objekt-Identifizier der TxPDO4
	0x02	Transmission Type	U8	rw	255 (asynchr.)	PDO-Übertragungsart
	0x05	Event Timer	U16	rw	0	Minimale Zeit zwischen TxPDO-Versendung <ul style="list-style-type: none"> <li>▪ 0: ausgeschaltet</li> <li>▪ 5...65 000: Zeit in Millisekunden</li> </ul>

### Transmit PDO1 Mapping

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1A00	0x00	Number of Mapped Objects	U8	rw	1	Anzahl gemappter Objekte
	0x01	TxPDO1 Mapping Entry 1	U32	rw	0x60410010	Verweis auf 16-Bit Statusword (0x6041)
	0x02	TxPDO1 Mapping Entry 2	U32	rw	0	
	0x03	TxPDO1 Mapping Entry 3	U32	rw	0	
	0x04	TxPDO1 Mapping Entry 4	U32	rw	0	

## Parameterbeschreibung

### Transmit PDO2 Mapping

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1A01	0x00	Number of Mapped Objects	U8	rw	2	Anzahl gemappter Objekte
	0x01	TxPDO2 Mapping Entry 1	U32	rw	0x60410010	Verweis auf 16-Bit Statusword (0x6041)
	0x02	TxPDO2 Mapping Entry 2	U32	rw	0x60640020	Verweis auf 32-Bit Position Actual Value (0x6064)
	0x03	TxPDO2 Mapping Entry 3	U32	rw	0	
	0x04	TxPDO2 Mapping Entry 4	U32	rw	0	

### Transmit PDO3 Mapping

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1A02	0x00	Number of Mapped Objects	U8	rw	2	Anzahl gemappter Objekte
	0x01	TxPDO3 Mapping Entry 1	U32	rw	0x60410010	Verweis auf 16-Bit Statusword (0x6041)
	0x02	TxPDO3 Mapping Entry 2	U32	rw	0x606C0020	Verweis auf 32-Bit Velocity Actual Value (0x606C)
	0x03	TxPDO3 Mapping Entry 3	U32	rw	0	
	0x04	TxPDO3 Mapping Entry 4	U32	rw	0	

### Transmit PDO4 Mapping

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x1A03	0x00	Number of Mapped Objects	U8	rw	2	Anzahl gemappter Objekte
	0x01	TxPDO4 Mapping Entry 1	U32	rw	0x60410010	Verweis auf 32-Bit Position Actual Value (0x6064)
	0x02	TxPDO4 Mapping Entry 2	U32	rw	0x606C0020	Verweis auf 32-Bit Velocity Actual Value (0x6077)
	0x03	TxPDO4 Mapping Entry 3	U32	rw	0	
	0x04	TxPDO4 Mapping Entry 4	U32	rw	0	

## Parameterbeschreibung

### 6.2 Herstellerspezifische Objekte

#### Digital Input Settings (0x2310)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2310	0x00	Number of Entries	U8	ro	6	Anzahl Objekteinträge
	0x01	Negative Limit Switch	U8	rw	0	Auswahl unterer Endschalter
	0x02	Positive Limit Switch	U8	rw	0	Auswahl oberer Endschalter
	0x03	Homing Switch	U8	rw	0x07 / 0x1F <sup>a)</sup>	Homing-Schalter
	0x05	Switch Polarity	U8	rw	0x07 / 0x1F <sup>a)</sup>	Polarität der Endschalter <ul style="list-style-type: none"> <li>▪ 1: Positive Flanke gültig</li> <li>▪ 0: Negative Flanke gültig</li> </ul>
	0x06	Polarity for Homing Limit	U8	rw	0	Polarität der Endschalter auch bei DSP402 Limit Switch Homing Method verwenden

a) BL-Controller/MCDC

Einstellungen der digitalen Eingänge gemäß Bitmaske in Kap. 4.8, S. 69.

#### Digital Input Status (0x2311)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2311	0x00	Number of Entries	U8	ro	4	Anzahl Objekteinträge
	0x01	Input Status	U8	ro	0	Logische Zustände der digitalen Eingänge
	0x02	Input Level	U8	ro	0	Physikalische Zustände der digitalen Eingänge

Zustände der digitalen Eingänge gemäß Bitmaske in Kap. 4.8, S. 69.

#### Analog Input Status (0x2313)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2313	0x00	Number of Entries	U8	ro	3/5 <sup>a)</sup>	Anzahl Objekteinträge
	0x01	Input 1 ADC Value	I16	ro	–	Spannung an Eingang 1 [mV] (AnIn)
	0x03	Input 3 ADC Value	I16	ro	–	Spannung an Eingang 3 [mV] (3rd In)
	0x04	Input 4 ADC Value	I16	ro	–	Spannung an Eingang 4 [mV] (nur MCDC)
	0x05	Input 5 ADC Value	I16	ro	–	Spannung an Eingang 5 [mV] (nur MCDC)

a) BL-Controller/MCDC



## Parameterbeschreibung

### Analog Input Status Raw (0x2314)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2314	0x00	Number of Entries	U8	ro	8	Anzahl Objekteinträge
	0x01	Input 1 ADC value raw	I16	ro	-	Digitalwert an Eingang 1
	0x02	Input 2 ADC value raw	I16	ro	-	Digitalwert an Eingang 2
	0x03	Input 3 ADC value raw	I16	ro	-	Digitalwert an Eingang 3
	0x04	Input 4 ADC value raw	I16	ro	-	Digitalwert an Eingang 4
	0x05	Input 5 ADC value raw	I16	ro	-	Digitalwert an Eingang 5
	0x06	Input 6 ADC value raw	I16	ro	-	Digitalwert an Eingang 6
	0x07	Input 7 ADC value raw	I16	ro	-	Digitalwert an Eingang 7
	0x08	Input 8 ADC value raw	I16	ro	-	Digitalwert an Eingang 8

An den intern verwendeten analogen Eingängen eingelesene Rohwerte.

### Fault-Pin Settings (0x2315)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2315	0x00	Number of Entries	U8	ro	3	Anzahl Objekteinträge
	0x01	Fault-Pin Function	U8	rw	0	Funktion des Fault-Pins <ul style="list-style-type: none"> <li>0: Error Output</li> <li>2: Digital Output</li> <li>4: Reference Input</li> <li>5: Position Output</li> </ul>
	0x03	Digital Output Status	U8	rw/ro <sup>a)</sup>	a)	Zustand des Pins in der Funktion Digital Output ändern <ul style="list-style-type: none"> <li>0: Clear Output</li> <li>1: Set Output</li> <li>2: Toggle Output</li> </ul>

a) Abhängig von der Konfiguration des Motion Controllers

### Input Threshold Level (0x2316)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2316	0x00	Input Threshold Level	U8	rw	1	Schaltpegel <ul style="list-style-type: none"> <li>0: 5V-TTL</li> <li>1: 24V-PLC</li> </ul>

### FAULHABER Fehlerregister (0x2320)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2320	0x00	Fault Register	U16	ro	-	FAULHABER Fehlerregister

## Parameterbeschreibung

### Error Mask (0x2321)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2321	0x00	Number of Entries	U8	ro	3	Anzahl Objekteinträge
	0x01	Emergency Mask	U16	rw	0xFFFF	Fehler, für die eine Fehlermeldung verschickt werden
	0x02	Fault Mask	U16	rw	0	Fehler, für die die Zustandsmaschine des Antriebs in den Zustand <i>Fault Reaction Active</i> geht
	0x03	Errout Mask	U16	rw	0x00FF	Fehler, für die der Fehler-Ausgangspin gesetzt wird

Für das FAULHABER Fehlerregister und die Error Mask gilt die in Kap. 3.10, S. 35 beschriebene Fehlerkodierung.

### Error Handling (0x2322)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2322	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Error Delay	U16	rw	200	Fehlervverzögerungszeit in 1/100 s
	0x02	Deviation	U16	rw	30 000	Zulässige Drehzahlabweichung in $\text{min}^{-1}$

Zusätzliche Einstellungen für die Fehlerbehandlung.

### Device Status (0x2323)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2323	0x00	Number of Entries	U8	ro	4	Anzahl Objekteinträge
	0x01	Housing Temperature	U16	ro	0	Gehäusetemperatur [°C]
	0x02	Internal Temperature	U16	ro	0	Spulen- bzw. MOSFET-Temperatur [°C]
	0x03	Max. Temperature Limit	U16	ro	0	Obere Temperaturschwelle [°C]
	0x04	Min. Temperature Limit	U16	ro	0	Untere Temperaturschwelle [°C]

Ausgabe des aktuellen Gerätezustands.

### Filter Settings (0x2330)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2330	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Sampling Rate	U16	rw	1	Faktor der Abtastrate
	0x02	Gain Scheduling	U16	rw	0	1: Reduzierte Reglerverstärkung im Zielkorridor bei Positionierung

Regler-Grundeinstellung.

## Parameterbeschreibung

### Velocity Control Parameter Set (0x2331)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2331	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Proportional Term POR	U16	rw	a)	Proportionalverstärkung des Drehzahlreglers
	0x02	Integral Term I	U16	rw	a)	Integralanteil des Drehzahlreglers

a) Abhängig von der Konfiguration des Motion Controllers

Parameter des Drehzahlreglers.

### Position Control Parameter Set (0x2332)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2332	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Proportional Term PP	U16	rw	a)	Proportionalverstärkung des Positionsreglers
	0x02	Derivative Term PD	U16	rw	a)	Differenzialanteil des Positionsreglers

a) Abhängig von der Konfiguration des Motion Controllers

Parameter des Positionsreglers.

### Current Control Parameter Set (0x2333)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2333	0x00	Number of Entries	U8	ro	3	Anzahl Objekteinträge
	0x01	Continuous Current Limit	U16	rw	a)	Dauerstrombegrenzung [mA]
	0x02	Peak Current Limit	U16	rw	a)	Spitzenstrombegrenzung [mA]
	0x03	Integral Term CI	U16	rw	a)	Integralanteil des Stromreglers

a) Abhängig von der Konfiguration des Motion Controllers

Strombegrenzungswerte und Parameter des Stromreglers.

### Actual Current Limit (0x2334)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2334	0x00	Actual Current Limit	U16	ro	–	Aktuell verwendeter Strombegrenzungswert

## Parameterbeschreibung

### General Settings (0x2338)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2338	0x00	Number of Entries	U8	ro	3	Anzahl Objekteinträge
	0x01	Pure Sinus Comutation	U16	rw	1	<ul style="list-style-type: none"> <li>▪ 0: Vollaussteuerung</li> <li>▪ 1: Begrenzung auf Sinusform (nicht MCDC)</li> </ul>
	0x02	Activate Position Limits in Velocity Mode	U16	rw	0	1: Eingestellte Positionierbereichsgrenzen auch im Drehzahlmodus verwenden
	0x03	Activate Position Limits in Position Mode	U16	rw	1	0: Keine Bereichsgrenzen im Positioniermodus

Allgemeine Einstellungen zu Kommutierung und den Bereichsgrenzen.

Im Profile Position Mode sind die Bereichsgrenzen standardmäßig aktiviert. Falls in der Anwendung keine mechanischen Begrenzungen vorhanden sind, können die Limits auch deaktiviert werden, um unbegrenzte relative Positionierungen durchführen zu können. In diesem Fall wird bei Überschreitung der oberen bzw. unteren Grenze bei 0 weitergezählt.

### Motor Data (0x2350)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2350	0x00	Number of Entries	U8	ro	5	Anzahl Objekteinträge
	0x01	Speed Constant $K_N$	U16	rw	<sup>a)</sup>	Drehzahlkonstante
	0x02	Terminal Resistance RM	U16	rw	<sup>a)</sup>	Anschlusswiderstand
	0x03	Pole Number	U16	rw	2/4 <sup>a)</sup>	Polzahl bei BL-Motoren (nicht MCDC)
	0x05	Thermal Time Constant TW1	U16	rw	<sup>a)</sup>	Thermische Zeitkonstante 1

a) Abhängig von der Auslieferungskonfiguration des Motion Controllers

Datenblattwerte des angeschlossenen Motors.

### Encoder Data (0x2351)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2351	0x00	Number of Entries	U8	ro	3	Anzahl Objekteinträge
	0x01	Sensor Type	U8	rw	0	<ul style="list-style-type: none"> <li>▪ 0: Analog Hall (int. Encoder)</li> <li>▪ 1: Inkrementalencoder (ext.)</li> <li>▪ 10: Inkremental + Hall</li> <li>▪ 104: Absolutencoder AES-4096 (nicht MCDC)</li> </ul>
	0x02	Resolution External Encoder	U32	rw	2 048	4-Flanken-Auflösung eines extern angeschlossenen Inkrementalgebers
	0x03	Resolution Internal Encoder	U32	ro	3 000	Auflösung des internen Hallsensorgebers (nicht MCDC)

Einstellung des zu verwendenden Encoders.

## Parameterbeschreibung

### Velocity Actual Value Unfiltered (0x2361)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2361	0x00	Velocity Actual Value Unfiltered	I16	ro	–	Ungefilterte Ist Drehzahl

### Baudrate Set (0x2400)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2400	0x00	Baudrate Set	U8	ro	0xFF	Eingestellte Baudrate

Über dieses Objekt kann abgefragt werden, welche Baudrate eingestellt ist. Zurückgegeben wird der Index der eingestellten Baudrate oder 0xFF, wenn AutoBaud eingestellt ist.

Baudrate	Index	Baudrate	Index
1000 KBit	0	250 KBit	3
800 KBit	1	125 KBit	4
500 KBit	2	AutoBaud	0xFF

### Position Demand Internal Value (0x2562)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x2562	0x00	Position Demand Internal Value	I32	rw	–	Letzte Sollposition in internen Einheiten

### Target Position Internal Value (0x257A)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x257A	0x00	Target Position Internal Value	I32	rw	–	Zielposition in internen Einheiten

### Position Range Limit Internal Value (0x257B)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x257B	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Minimum Position Range Limit	I32	ro	$-1,8 \cdot 10^6$	Interne untere Positionierbereichsgrenze in internen Einheiten
	0x02	Maximum Position Range Limit	I32	ro	$1,8 \cdot 10^6$	Interne obere Positionierbereichsgrenze in internen Einheiten

### Software Position Limit Internal Value (0x257D)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x257D	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Minimum Position Limit	I32	ro	$-1,8 \cdot 10^6$	Untere Positionierbereichsgrenze in internen Einheiten
	0x02	Maximum Position Limit	I32	ro	$1,8 \cdot 10^6$	Obere Positionierbereichsgrenze in internen Einheiten

## Parameterbeschreibung

### 6.3 Objekte des Antriebsprofils CiA 402

#### Controlword (0x6040)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6040	0x00	Controlword	U16	rw	0	Controlword

Die Bits im Controlword sind in Kap. 4.2, S. 40 beschrieben.

#### Statusword (0x6041)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6041	0x00	Statusword	U16	ro	0	Statusword

Die Bits im Statusword sind in Kap. 4.2, S. 40 beschrieben.

#### Modes of Operation (0x6060)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6060	0x00	Modes of Operation	I8	rw	1	Auswahl der Betriebsart 0: Regler nicht aktiviert 1: Profile Position (Positionsregelung) 3: Profile Velocity (Drehzahlregelung) 6: Homing (Referenzfahrt) 8: Cyclic Synchronous Position (Positionsregelung)

#### Modes of Operation Display (0x6061)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6061	0x00	Modes of Operation Display	I8	ro	1	Anzeige der ausgewählten Betriebsart

Die Bedeutung der Rückgabewerte entspricht den Werten des Objekts 0x6060.

#### Position Demand Value (0x6062)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6062	0x00	Position Demand Value	I32	ro	–	Positionssollwert in benutzerdefinierten Einheiten

#### Position Actual Internal Value (0x6063)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6063	0x00	Position Actual Internal Value	I32	ro	–	Positionswert in internen Einheiten

#### Position Actual Value (0x6064)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6064	0x00	Position Actual Value	I32	ro	–	Positionswert in benutzerdefinierten Einheiten

#### Position Window (0x6067)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6067	0x00	Position Window	U32	rw	32	Korridor um die Zielposition in benutzerdefinierten Einheiten

## Parameterbeschreibung

### Position Window Time (0x6068)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6068	0x00	Position Window Time	U16	rw	48	Mindestverweildauer im Korridor, bis die Zielposition in der Betriebsart PP als erreicht signalisiert wird.

### Velocity Demand Value (0x606B)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x606B	0x00	Velocity Demand Value	I32	ro	–	Solldrehzahl in $\text{min}^{-1}$

### Velocity Actual Value (0x606C)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x606C	0x00	Velocity Actual Value	I32	ro	–	Istdrehzahl in $\text{min}^{-1}$

### Velocity Window (0x606D)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x606D	0x00	Velocity Window	U16	rw	20	Korridor um die Zieldrehzahl in benutzerdefinierten Einheiten

### Velocity Window Time (0x606E)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x606E	0x00	Velocity Window Time	U16	rw	200	Mindestverweildauer im Korridor [ms]

### Velocity Threshold (0x606F)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x606F	0x00	Velocity Threshold	U16	rw	20	Drehzahlschwellwert in $\text{min}^{-1}$

### Velocity Threshold Time (0x6070)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6070	0x00	Velocity Threshold Time	U16	rw	48	Überwachungszeit [ms]. Verlässt die Drehzahl länger als hier angegeben den Korridor, wird die Drehzahl als ungleich 0 gemeldet.

### Current Actual Value (0x6078)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6078	0x00	Current Actual Value	I16	ro	–	Aktuelle Motor-Stromaufnahme in mA

### Target Position (0x607A)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x607A	0x00	Target Position	I32	rw	–	Positionssollwert in benutzerdefinierten Einheiten

## Parameterbeschreibung

### Position Range Limit (0x607B)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x607B	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Min Position Range Limit	I32	ro	$-1,8 \cdot 10^9$	Untere Grenze des Positionsbereichs in benutzerdefinierten Einheiten
	0x02	Max Position Range Limit	I32	ro	$1,8 \cdot 10^9$	Obere Grenze des Positionsbereichs in benutzerdefinierten Einheiten

### Homing Offset (0x607C)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x607C	0x00	Min Range Limit	I32	rw	0	Offset der Null-Position gegenüber der Position des Referenzschalters

### Software Position Limit (0x607D)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x607D	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Min Position Limit	I32	rw	$-1,8 \cdot 10^9$	Untere Grenze des Positionsbereichs
	0x02	Max Position Limit	I32	rw	$1,8 \cdot 10^9$	Obere Grenze des Positionsbereichs

### Polarity (0x607E)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x607E	0x00	Polarity	U8	rw	0	Bit-codiert

Über dieses Objekt kann die Drehrichtung global geändert werden, d. h. Vorgabe- und Istwerte für Position bzw. Drehzahl werden mit  $-1$  multipliziert:

- Bit 7 = 1: negative Drehrichtung im Positionierbetrieb
- Bit 6 = 1: negative Drehrichtung im Drehzahlbetrieb

### Max Profile Velocity (0x607F)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x607F	0x00	Max Profile Velocity	U32	rw	a)	Maximal erlaubte Drehzahl [ $\text{min}^{-1}$ ]

a) Abhängig von der Auslieferungskonfiguration des Motion Controllers

### Profile Velocity (0x6081)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6081	0x00	Profile Velocity	U32	rw	a)	Maximaldrehzahl in $\text{min}^{-1}$

a) Abhängig von der Auslieferungskonfiguration des Motion Controllers

### Profile Acceleration (0x6083)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6083	0x00	Profile Acceleration	U32	rw	30 000	Maximale Beschleunigung [ $1/\text{s}^2$ ]



## Parameterbeschreibung

### Profile Deceleration (0x6084)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6084	0x00	Profile Deceleration	U32	rw	30 000	Maximale Bremsrate [ $1/s^2$ ]

### Quick Stop Deceleration (0x6085)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6085	0x00	Quick Stop Deceleration	U32	rw	30 000	Bremsrampenwert bei Quick Stop [ $1/s^2$ ]

### Position Encoder Resolution (0x608F)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x608F	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Encoder Increments	U32	ro	3 000/ 2 048 <sup>a)</sup>	Encoderinkremente
	0x02	Motor Revolutions	U32	ro	1	Anzahl der Motorumdrehungen bei der in Subindex 0x01 genannten Impulszahl

a) BL-Controller/MCDC

Der Wert der Encoderauflösung wird aus den Einstellungen in Objekt Encoder Data (0x2351) übernommen und kann hier nicht verändert werden.

### Gear Ratio (0x6091)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6091	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Motor Revolutions	U32	rw	1	Anzahl Motorumdrehungen
	0x02	Shaft Revolutions	U32	rw	1	Anzahl Umdrehungen der Abtriebswelle

### Feed Constant (0x6092)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6092	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Feed	U32	rw	3 000 / 2 048 <sup>a)</sup>	Vorschub in Benutzereinheiten
	0x02	Shaft Revolutions	U32	rw	1	Anzahl Umdrehungen der Abtriebswelle

a) BL-Controller/MCDC

Über die Faktoren Gear Ratio und Feed Constant können eine Getriebeübersetzung und ein Vorschubwert zur Umrechnung in Benutzereinheiten angegeben werden (siehe Kap. 4.3, S. 48).

Bei Auslieferung sind die Benutzereinheiten auf die Standard-Encoderauflösung eingestellt (3 000 bzw. 2 048 Inkremente pro Umdrehung).

## Parameterbeschreibung

### Position Factor (0x6093)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6093	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Numerator	U32	ro	1	Zähler des Positionsfaktors
	0x02	Divisor	U32	ro	1	Nenner des Positionsfaktors

$$\text{Internal position} = \frac{\text{Position in user units} \cdot \text{Position factor numerator}}{\text{Position factor denominator}}$$

Der Positionsfaktor wird berechnet aus Position Encoder Resolution (0x608F), Gear Ratio (0x6091) und Feed Constant (0x6092) und kann hier nicht verändert werden (siehe Kap. 4.3, S. 48).

### Homing Method (0x6098)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6098	0x00	Homing Method	I8	rw	20	Homing-Methode

### Homing Speed (0x6099)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6099	0x00	Number of Entries	U8	ro	2	Anzahl Objekteinträge
	0x01	Switch Seek Velocity	U32	rw	400	Drehzahl bei Schaltersuche [min <sup>-1</sup> ]
	0x02	Homing Speed	U32	rw	100	Drehzahl bei Nullpunktsuche [min <sup>-1</sup> ]

### Homing Acceleration (0x609A)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x609A	0x00	Homing Acceleration	U32	rw	50	Beschleunigung während des Homings [1/s <sup>2</sup> ]

### Control Effort (0x60FA)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x60FA	0x00	Control Effort	U32	ro	–	Reglerausgang

### Digital Inputs (0x60FD)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x60FD	0x00	Control Effort	U32	ro	–	Zustand digitaler Eingänge

Dieses Objekt zeigt über folgende Bits an, welcher Schalter ein- bzw. ausgeschaltet ist:

- Bit 0: Negative Limit Switch
- Bit 1: Positive Limit Switch
- Bit 2: Homing Switch

### Target Velocity (0x60FF)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x60FF	0x00	Target Velocity	I32	rw	–	Drehzahlsollwert [min <sup>-1</sup> ]

## Parameterbeschreibung

### Supported Drive Modes (0x6502)

Index	Subindex	Name	Typ	Attr.	Standardwert	Bedeutung
0x6502	0x00	Supported Drive Modes	U32	ro	0x25	Unterstützte Betriebsarten

Das Objekt Supported Drive Modes gibt die von den FAULHABER Motion Control Systems unterstützten Betriebsarten an:

- Bit 0: Profile Position Mode (PP)
- Bit 2: Profile Velocity Mode (PV)
- Bit 5: Homing Mode (HM)
- Bit 7: Cyclic Synchronous Position (CSP)

