

# CANopen®

## CANopen – SD6 Handbuch

de  
10/2023  
ID 442636.02



**STÖBER**

# Inhaltsverzeichnis

<b>1</b>	<b>Vorwort .....</b>	<b>5</b>
<b>2</b>	<b>Benutzerinformationen .....</b>	<b>6</b>
2.1	Aufbewahrung und Weitergabe .....	6
2.2	Beschriebenes Produkt .....	6
2.3	Richtlinien und Normen .....	6
2.4	Aktualität .....	6
2.5	Originalsprache .....	6
2.6	Haftungsbeschränkung .....	7
2.7	Darstellungskonventionen .....	7
2.7.1	Darstellung von Warnhinweisen .....	7
2.7.2	Auszeichnung von Textelementen .....	8
2.7.3	Mathematik und Formeln .....	8
2.8	Marken .....	9
<b>3</b>	<b>Sicherheitshinweise .....</b>	<b>10</b>
<b>4</b>	<b>Netzwerkaufbau .....</b>	<b>11</b>
<b>5</b>	<b>Kommunikationsmodul CA6 .....</b>	<b>12</b>
5.1	Einbau .....	12
<b>6</b>	<b>Anschluss .....</b>	<b>13</b>
6.1	Auswahl geeigneter Kabel .....	13
6.2	Abschlusswiderstände .....	13
6.3	Baudrate und Kabellänge .....	14
6.4	X200: Feldbusanbindung .....	14
<b>7</b>	<b>Was Sie vor der Inbetriebnahme wissen sollten .....</b>	<b>15</b>
7.1	Programmoberfläche DS6 .....	15
7.1.1	Ansicht konfigurieren .....	17
7.1.2	Navigation über sensitive Schaltbilder .....	18
7.2	Bedeutung der Parameter .....	19
7.2.1	Parametergruppen .....	19
7.2.2	Parameterarten und Datentypen .....	20
7.2.3	Parametertypen .....	21
7.2.4	Parameteraufbau .....	21
7.2.5	Parametersichtbarkeit .....	22
7.3	Signalquellen und Prozessdaten-Mapping .....	23
7.4	Nichtflüchtiges Speichern .....	23

<b>8</b>	<b>Inbetriebnahme .....</b>	<b>24</b>
8.1	DS6: Antriebsregler konfigurieren .....	25
8.1.1	Projekt aufsetzen .....	25
8.1.2	Allgemeine CANopen-Einstellungen parametrieren .....	28
8.1.3	PDO-Übertragung konfigurieren .....	29
8.1.4	Konfiguration übertragen und speichern .....	31
8.2	CANopen-Netzwerk in Betrieb nehmen .....	33
<b>9</b>	<b>Monitoring und Diagnose .....</b>	<b>34</b>
9.1	Verbindungsüberwachung .....	34
9.2	LED-Anzeige .....	35
9.2.1	Zustand CANopen .....	35
9.3	Ereignisse .....	36
9.3.1	Ereignis 52: Kommunikation .....	37
9.4	Parameter .....	38
9.4.1	A83   Busadresse   G6   V0 .....	38
9.4.2	A245   CAN-Diagnose   G6   V0 .....	38
9.4.3	A246   CANopen NMT-State   G6   V1 .....	38
<b>10</b>	<b>Mehr zu CANopen? .....</b>	<b>39</b>
10.1	CAN und CANopen .....	39
10.1.1	CANopen – Kommunikation .....	39
10.1.2	Objektverzeichnis .....	40
10.1.3	Netzwerkstruktur .....	40
10.2	CAN-Nachricht .....	41
10.2.1	Predefined Connection Set .....	41
10.2.2	Dynamic Distribution .....	43
10.3	Kommunikationsobjekte .....	44
10.3.1	SYNC: Synchronization Objects .....	45
10.3.2	PDO: Process Data Objects .....	45
10.3.3	SDO: Service Data Objects .....	48
10.3.4	NMT: Network Management Objects .....	53
10.3.5	ERROR CONTROL: Error Control Objects .....	55
10.3.6	EMCY: Emergency Objects .....	57
10.4	Zykluszeiten .....	57
10.5	Aktionen ansteuern und ausführen .....	58
10.6	Feldbusskalierung .....	60
<b>11</b>	<b>Anhang .....</b>	<b>61</b>
11.1	Unterstützte Kommunikationsobjekte .....	61
11.1.1	CiA 301 CANopen: 1000 hex – 1FFFF hex .....	61
11.1.2	Herstellerspezifische Parameter: 2000 hex – 53FF hex .....	66

11.2 EMCY-Nachricht: Fehler-Codes Gerätestörung ..... 68

11.3 Weiterführende Informationen..... 70

11.4 Abkürzungen..... 71

**12 Kontakt ..... 72**

12.1 Beratung, Service, Anschrift ..... 72

12.2 Ihre Meinung ist uns wichtig ..... 72

12.3 Weltweite Kundennähe..... 73

**Glossar ..... 74**

# 1 Vorwort

STÖBER Antriebsregler der 6. Generation bieten der Automatisierungstechnik und dem Maschinenbau trotz immer komplexer werdender Funktionen höchste Präzision und Produktivität.

Für eine zuverlässige Kommunikation zwischen Antriebstechnik und Steuerung über das Feldbussystem CANopen können die Antriebsregler der Baureihe SD6 mit dem Kommunikationsmodul CA6 ausgestattet werden.

Die vorliegende Dokumentation beschreibt eine Kombination der genannten Antriebsregler mit einer Steuerung als CANopen-Master und der zugehörigen Automatisierungssoftware.

## 2 Benutzerinformationen

Diese Dokumentation unterstützt Sie bei der Inbetriebnahme der Baureihe SD6 der 6. STÖBER Antriebsreglergeneration in Verbindung mit übergeordneten Steuerungssystemen über ein CANopen-Netzwerk.

Die Antriebsregler SD6 stellen die CANopen-Funktionalität über ein separates Kommunikationsmodul CA6 zur Verfügung, das Sie, sofern nicht bereits werkseitig montiert, einbauen müssen.

### Fachliche Vorkenntnisse

Um den CANopen-Verbund in Betrieb nehmen zu können, sollten Ihnen die Grundlagen der Netzwerktechnologie CANopen bekannt sein.

### Technische Voraussetzungen

Bevor Sie Ihr CANopen-Netzwerk in Betrieb nehmen, müssen Sie die Antriebsregler verdrahtet und deren korrekte Funktionsweise initial überprüft haben. Folgen Sie hierzu den Anweisungen im Handbuch des Antriebsreglers.

### Gender-Hinweis

Aus Gründen der besseren Lesbarkeit wird auf eine geschlechtsneutrale Differenzierung verzichtet. Entsprechende Begriffe gelten im Sinne der Gleichbehandlung grundsätzlich für alle Geschlechter. Die verkürzte Sprachform beinhaltet also keine Wertung, sondern hat lediglich redaktionelle Gründe.

## 2.1 Aufbewahrung und Weitergabe

Da diese Dokumentation wichtige Informationen zum sicheren und effizienten Umgang mit dem Produkt enthält, bewahren Sie diese bis zur Produktentsorgung unbedingt in unmittelbarer Nähe des Produkts und für das qualifizierte Personal jederzeit zugänglich auf.

Bei Übergabe oder Verkauf des Produkts an Dritte geben Sie diese Dokumentation ebenfalls weiter.

## 2.2 Beschriebenes Produkt

Diese Dokumentation ist verbindlich für:

Antriebsregler der Baureihe SD6 in Verbindung mit der Software DriveControlSuite (DS6) ab V 6.5-K und zugehöriger Firmware ab V 6.5-K.

## 2.3 Richtlinien und Normen

Die für den Antriebsregler und das Zubehör relevanten europäischen Richtlinien und Normen entnehmen Sie der Dokumentation des Antriebsreglers.

## 2.4 Aktualität

Prüfen Sie, ob Ihnen mit diesem Dokument die aktuelle Version der Dokumentation vorliegt. Auf unserer Webseite stellen wir Ihnen die neuesten Dokumentversionen zu unseren Produkten zum Download zur Verfügung:

<http://www.stoeber.de/de/downloads/>.

## 2.5 Originalsprache

Die Originalsprache dieser Dokumentation ist Deutsch; alle anderssprachigen Fassungen sind von der Originalsprache abgeleitet.

## 2.6 Haftungsbeschränkung

Diese Dokumentation wurde unter Berücksichtigung der geltenden Normen und Vorschriften sowie des Stands der Technik erstellt.

Für Schäden, die aufgrund einer Nichtbeachtung der Dokumentation oder aufgrund der nicht bestimmungsgemäßen Verwendung des Produkts entstehen, bestehen keine Gewährleistungs- und Haftungsansprüche. Dies gilt insbesondere für Schäden, die durch individuelle technische Veränderungen des Produkts oder dessen Projektierung und Bedienung durch nicht qualifiziertes Personal hervorgerufen wurden.

## 2.7 Darstellungskonventionen

Damit Sie besondere Informationen in dieser Dokumentation schnell zuordnen können, sind diese durch Orientierungshilfen in Form von Signalwörtern, Symbolen und speziellen Textauszeichnungen hervorgehoben.

### 2.7.1 Darstellung von Warnhinweisen

Warnhinweise sind durch Symbole gekennzeichnet. Sie weisen Sie auf besondere Gefahren im Umgang mit dem Produkt hin und werden durch entsprechende Signalworte begleitet, die das Ausmaß der Gefährdung zum Ausdruck bringen. Darüber hinaus sind nützliche Tipps und Empfehlungen für einen effizienten und einwandfreien Betrieb besonders hervorgehoben.

#### ACHTUNG!

##### Achtung

bedeutet, dass ein Sachschaden eintreten kann,

- wenn die genannten Vorsichtsmaßnahmen nicht getroffen werden.

#### ⚠ VORSICHT!

##### Vorsicht

mit Warndreieck bedeutet, dass eine leichte Körperverletzung eintreten kann,

- wenn die genannten Vorsichtsmaßnahmen nicht getroffen werden.

#### ⚠ WARNUNG!

##### Warnung

mit Warndreieck bedeutet, dass erhebliche Lebensgefahr eintreten kann,

- wenn die genannten Vorsichtsmaßnahmen nicht getroffen werden.

#### ⚠ GEFAHR!

##### Gefahr

mit Warndreieck bedeutet, dass erhebliche Lebensgefahr eintreten wird,

- wenn die genannten Vorsichtsmaßnahmen nicht getroffen werden.

#### Information

Information bedeutet eine wichtige Information über das Produkt oder die Hervorhebung eines Dokumentationsteils, auf den besonders aufmerksam gemacht werden soll.

## 2.7.2 Auszeichnung von Textelementen

Bestimmte Elemente des Fließtexts werden wie folgt ausgezeichnet.

<b>Wichtige Information</b>	Wörter oder Ausdrücke mit besonderer Bedeutung
Interpolated position mode	Optional: Datei-, Produkt- oder sonstige Namen
<u>Weiterführende Informationen</u>	Interner Querverweis
<a href="http://www.musterlink.de">http://www.musterlink.de</a>	Externer Querverweis

### Software- und Display-Anzeigen

Um den unterschiedlichen Informationsgehalt von Elementen, die von der Software-Oberfläche oder dem Display eines Antriebsreglers zitiert werden sowie eventuelle Benutzereingaben entsprechend kenntlich zu machen, werden folgende Darstellungen verwendet.

Hauptmenü Einstellungen	Von der Oberfläche zitierte Fenster-, Dialog-, Seitennamen oder Schaltflächen, zusammengesetzte Eigennamen, Funktionen
Wählen Sie Referenziermethode A	Vorgegebene Eingabe
Hinterlegen Sie Ihre <Eigene IP-Adresse>	Benutzerdefinierte Eingabe
EREIGNIS 52: KOMMUNIKATION	Display-Anzeigen (Status, Meldungen, Warnungen, Störungen)

Tastenkürzel und Befehlsfolgen oder Pfade sind folgendermaßen dargestellt.

[Strg], [Strg] + [S]	Taste, Tastenkombination
Tabelle > Tabelle einfügen	Navigation zu Menüs/Untermenüs (Pfadangabe)

## 2.7.3 Mathematik und Formeln

Zur Darstellung von mathematischen Zusammenhängen und Formeln werden die folgenden Zeichen verwendet.

–	Subtraktion
+	Addition
×	Multiplikation
÷	Division
	Betrag



## 2.8 Marken

Die folgenden Namen, die in Verbindung mit dem Gerät, seiner optionalen Ausstattung und seinem Zubehör verwendet werden, sind Marken oder eingetragene Marken anderer Unternehmen:

CANopen <sup>®</sup> , CiA <sup>®</sup>	CANopen <sup>®</sup> und CiA <sup>®</sup> sind eingetragene Unionsmarken des CAN in AUTOMATION e.V., Nürnberg, Deutschland.
Windows <sup>®</sup> , Windows <sup>®</sup> 7, Windows <sup>®</sup> 10, Windows <sup>®</sup> 11	Windows <sup>®</sup> , das Windows <sup>®</sup> -Logo, Windows <sup>®</sup> XP, Windows <sup>®</sup> 7, Windows <sup>®</sup> 10 und Windows <sup>®</sup> 11 sind eingetragene Marken der Microsoft Corporation in den USA und/ oder anderen Ländern.

Alle anderen, hier nicht aufgeführten Marken, sind Eigentum ihrer jeweiligen Inhaber.

Erzeugnisse, die als Marken eingetragen sind, sind in dieser Dokumentation nicht besonders kenntlich gemacht.

Vorliegende Schutzrechte (Patente, Warenzeichen, Gebrauchsmusterschutz) sind zu beachten.

## 3 Sicherheitshinweise



### WARNUNG!

#### Lebensgefahr bei Nichtbeachtung von Sicherheitshinweisen und Restrisiken!

Bei Nichtbeachtung der Sicherheitshinweise und Restrisiken in der Dokumentation des Antriebsreglers können Unfälle mit schweren Verletzungen oder Tod auftreten.

- Halten Sie die Sicherheitshinweise in der Antriebsregler-Dokumentation ein.
- Berücksichtigen Sie bei der Risikobeurteilung für die Maschine oder Anlage die Restrisiken.



### WARNUNG!

#### Fehlfunktion der Maschine infolge fehlerhafter oder veränderter Parametrierung!

Bei fehlerhafter oder veränderter Parametrierung können Fehlfunktionen an Maschinen oder Anlagen auftreten, die zu schweren Verletzungen oder Tod führen können.

- Beachten Sie die Security-Hinweise in der Antriebsregler-Dokumentation.
- Schützen Sie z. B. die Parametrierung vor unbefugtem Zugriff.
- Treffen Sie geeignete Maßnahmen für mögliche Fehlfunktionen (z. B. Not-Aus oder Not-Halt).

## 4 Netzwerkaufbau

Ein CANopen-Netzwerk besteht in der Regel aus einem CANopen-Master (Steuerung) sowie CANopen-Slaves, d. h. Antriebsreglern der Baureihe SD6.

Der CANopen-Netzwerkaufbau ist generell für die Linientopologie optimiert. Alle Teilnehmer im CANopen-Netzwerk (Master und Slaves) sind über 3 geschirmte Leitungen miteinander verbunden: CAN-High (CAN-H) und CAN-Low (CAN-L) sowie eine zusätzliche Leitung für das Bezugspotenzial (GND) zwischen den Teilnehmern. Um Datenüberlagerungen durch Signalreflexionen an den beiden Busenden zu vermeiden und die Störanfälligkeit somit erheblich zu reduzieren, werden zwischen CAN-High und CAN-Low jeweils Abschlusswiderstände von 120  $\Omega$  zugeschaltet.

Mit der STÖBER Inbetriebnahme-Software DriveControlSuite konfigurieren Sie die Antriebsregler, über die Automatisierungssoftware der Steuerung das gesamte CANopen-Netzwerk.

Nachfolgende Grafik abstrahiert ein CANopen-Netzwerk mit einer Steuerung als CANopen-Master sowie mehreren Antriebsreglern SD6 als CANopen-Slaves.

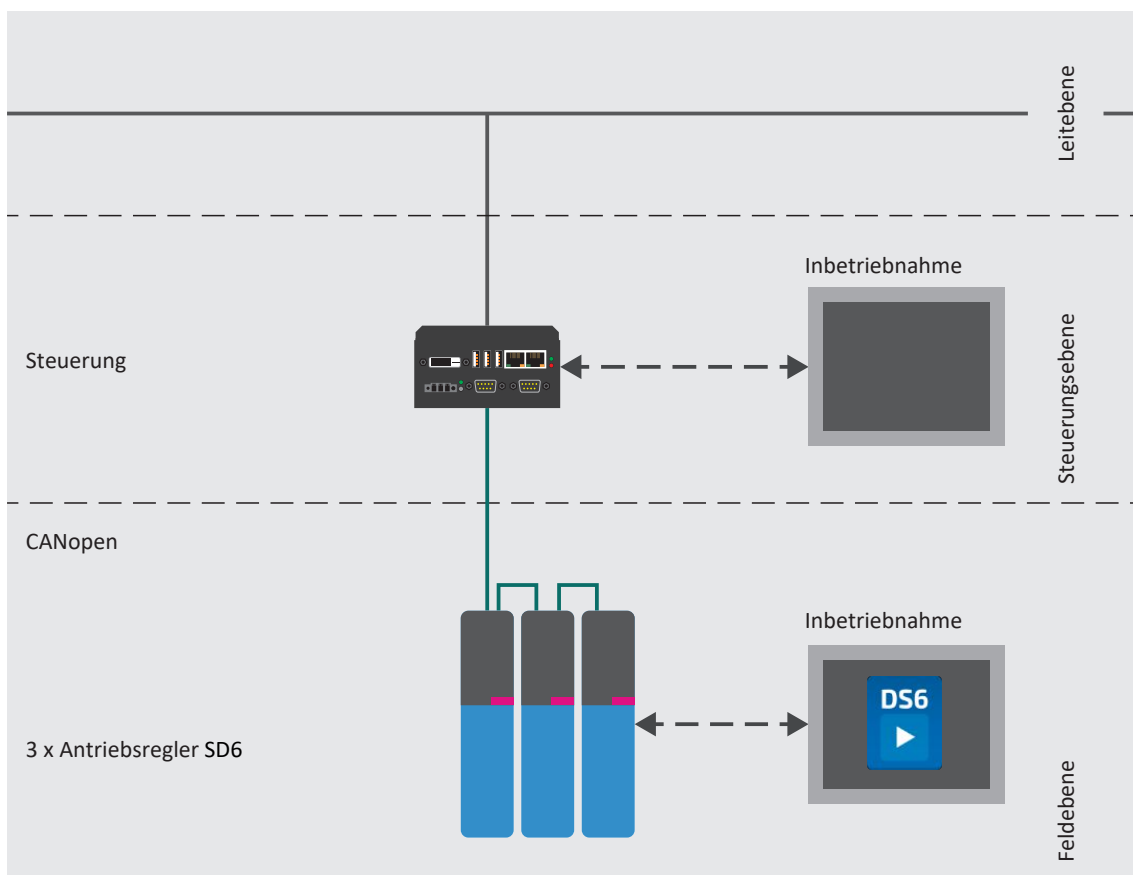


Abb. 1: CANopen: Netzwerkaufbau am Beispiel der Baureihe SD6

## 5 Kommunikationsmodul CA6

Die Antriebsregler der Baureihe SD6 sind untereinander sowie mit dem CANopen-Master über die Kommunikationsmodule CA6 verbunden, die die notwendigen Feldbusschnittstellen zur Verfügung stellen.

Die Kommunikationsmodule CA6 entsprechen dem CANopen-Standard und ermöglichen den zyklischen wie azyklischen Zugriff des CANopen-Masters auf sämtliche relevanten Antriebsparameter und Funktionen.

### 5.1 Einbau

Einbauarbeiten sind ausschließlich bei Spannungsfreiheit erlaubt. Beachten Sie die 5 Sicherheitsregeln.

Beachten Sie beim Einbau die in den technischen Daten angegebenen Mindestfreiräume, um eine Überhitzung des Antriebsreglers zu vermeiden.

Schützen Sie das Gerät bei der Aufstellung oder sonstigen Arbeiten im Schaltschrank gegen herunterfallende Teile (Drahtreste, Adern, Metallteile, usw.). Teile mit leitenden Eigenschaften können innerhalb des Antriebsreglers zu einem Kurzschluss oder Ausfall führen.

Entfernen Sie zusätzliche Abdeckungen vor der Inbetriebnahme, damit es nicht zur Überhitzung des Antriebsreglers kommen kann.



#### **WARNUNG!**

##### **Elektrische Spannung! Lebensgefahr durch Stromschlag!**

- Schalten Sie vor sämtlichen Arbeiten an den Geräten alle Versorgungsspannungen ab!
- Beachten Sie die Entladungszeit der Zwischenkreiskondensatoren von bis zu 6 Minuten. Sie können erst nach dieser Zeitspanne die Spannungsfreiheit voraussetzen.

#### **ACHTUNG!**

##### **Sachschaden durch elektrostatische Entladung!**

Treffen Sie bei der Handhabung offener Leiterplatten geeignete Schutzmaßnahmen, z. B. durch ESD-gerechte Kleidung.

Berühren Sie nicht die Kontaktflächen.

##### **Werkzeug und Material**

Sie benötigen:

- Einen TORX-Schraubendreher TX10
- Die dem Kommunikationsmodul beigegefügte Abdeckung und Schrauben

##### **Einbau**

1. Lösen Sie die Befestigungsschraube der Blindabdeckung auf der Oberseite des Antriebsreglers und entfernen Sie die Abdeckung.
2. Schieben Sie das Kommunikationsmodul an den Führungsschienen in den Antriebsregler.
3. Drücken Sie auf das Modul, sodass die Stiftkontakte in die Buchsenleiste geschoben werden.
4. Setzen Sie die Nasen der dem Kommunikationsmodul beigefügten Abdeckung vorne schräg in die Aussparung.
5. Legen Sie die Abdeckung auf dem Antriebsregler ab, sodass die Nasen unter der Kante liegen.
6. Befestigen Sie die Abdeckung mit beiden Schrauben.

## 6 Anschluss

Um einzelne Antriebsregler untereinander oder mit der Steuerung verbinden zu können, stellt das Kommunikationsmodul CA6 eine Feldbusschnittstelle in Form eines 9-poligen D-Sub-Steckers für den Anschluss an das CAN-Bussystem zur Verfügung.

### 6.1 Auswahl geeigneter Kabel

#### CAN-Bus

CAN-Busse existieren mit unterschiedlichen Mantelmaterialien für verschiedene Anwendungsszenarien und Umgebungsbedingungen. Für die Übertragung von CAN-Signalen ist mindestens ein 3-poliges Kabel mit CAN-High (CAN-H), CAN-Low (CAN-L) und Bezugspotenzial (GND) erforderlich.

Um insbesondere bei hohen Übertragungsraten einen fehlerfreien Betrieb zu gewährleisten, empfehlen wir die Verwendung von Kabeln, die den in ISO 11898-2 genannten Anforderungen entsprechen, beispielsweise:

- Wellenwiderstand: 95 – 140  $\Omega$
- Maximale Betriebskapazität: 60 nF/km
- Leiterwiderstand: 70 m $\Omega$ /m

#### Steckverbinder

Als Steckverbinder am Anfang und am Ende eines CAN-Busses sind 9-polige D-Sub-Stecker mit Metallgehäuse oder Gehäuse aus metallisiertem Kunststoff geeignet; zum Durchschleifen sind Steckergehäuse mit 2 Kabeleinführungen von Vorteil.

#### Schirmung

Eine geeignete Schirmung ist vor allem bei hohen Übertragungsraten unerlässlich. Die Schirmung des CAN-Busses wird unter der Zugentlastung des Steckers montiert; über das Steckergehäuse und den D-Sub-Stecker ist die Schirmung somit mit dem Antriebsregler verbunden. Achten Sie auf eine durchgängige Schirmung über die gesamte Kabellänge hinweg.

### 6.2 Abschlusswiderstände

Im CANopen-Netzwerk müssen am ersten und am letzten Teilnehmer Abschlusswiderstände vorgesehen werden, auch bei geringer Kabellänge. Nachfolgende Grafik abstrahiert ein CANopen-Netzwerk, bei dem der erste Abschlusswiderstand der Steuerung (Master) zugeschaltet ist und der zweite dem letzten teilnehmenden Antriebsregler SD6 (Slave). Um den Abschlusswiderstand am Kommunikationsmodul CA6 zu aktivieren, muss der Schiebeschalter an Anschluss X200 auf ON gesetzt werden.

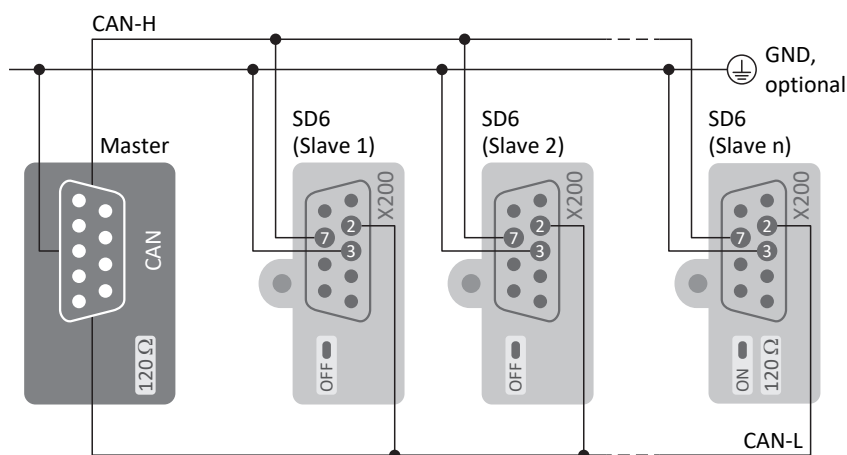


Abb. 2: CANopen-Netzwerk: Abschlusswiderstände

## 6.3 Baudrate und Kabellänge

Die maximal zulässige Ausdehnung eines CANopen-Netzwerks ist abhängig von der Übertragungsgeschwindigkeit, d. h. der Baudrate. Mit geringerer Kabellänge erhöhen sich die Übertragungsgeschwindigkeit sowie die Empfindlichkeit serieller Feldbussysteme.

Nachfolgende Tabelle zeigt die maximal zulässigen Kabellängen über die gesamte Ausdehnung des CANopen-Netzwerks in Abhängigkeit von der jeweils möglichen Baudrate.

Baudrate (kbit/s)	Maximale Kabellänge (m)
10	5000
20	2500
50	1000
100	800
125	500
250	250
500	100
800	< 30, nur mit Sonderkabel $\pm 60$ nF/km
1000	< 10, nur mit Sonderkabel $\pm 60$ nF/km

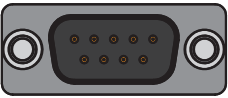
Tab. 1: CAN-Bus: Übertragungsgeschwindigkeit und max. Kabellänge

### Information

Sämtliche Antriebsregler, die innerhalb eines CANopen-Netzwerks via CAN-Bus verbunden sind, müssen dieselbe Baudrate besitzen.

## 6.4 X200: Feldbusanbindung

Um die Antriebsregler an weitere CANopen-Teilnehmer anbinden zu können, stellt das Kommunikationsmodul CA6 einen 9-poligen D-Sub-Stecker zur Verfügung.

Stecker	Pin	Bezeichnung	Funktion
	1	—	—
	2	CAN-L	Leitung CAN-Low
	3	GND	Bezugspotenzial
	4	—	—
	5	—	—
	6	—	—
	7	CAN-H	Leitung CAN-High
	8	—	—
	9	—	—

Tab. 2: Anschlussbeschreibung X200

## 7 Was Sie vor der Inbetriebnahme wissen sollten

Nachfolgende Kapitel ermöglichen Ihnen einen schnellen Einstieg in den Aufbau der Programmoberfläche sowie die zugehörigen Fensterbezeichnungen und liefern Ihnen relevante Informationen rund um Parameter sowie zum generellen Speichern Ihrer Projektierung.

### 7.1 Programmoberfläche DS6

Über die grafische Oberfläche der Inbetriebnahme-Software DriveControlSuite (DS6) können Sie Ihr Antriebsprojekt schnell und effizient projektieren, parametrieren und in Betrieb nehmen. Im Servicefall können Sie mithilfe der DriveControlSuite Diagnoseinformationen wie Betriebszustände, Störungsspeicher und Störungszähler Ihres Antriebsprojekts auswerten.

#### Information

Die Programmoberfläche der DriveControlSuite steht Ihnen in deutscher, englischer und französischer Sprache zur Verfügung. Um die Sprache der Programmoberfläche zu ändern, wählen Sie Menü **Einstellungen > Sprache**.

#### Information

Die Hilfe der DriveControlSuite erreichen Sie in der Menüleiste über Menü **Hilfe > Hilfe zur DS6** oder über die Taste [F1] auf Ihrer Tastatur. Abhängig vom Programmbereich, in dem Sie [F1] drücken, öffnet sich ein thematisch passendes Hilfethema.

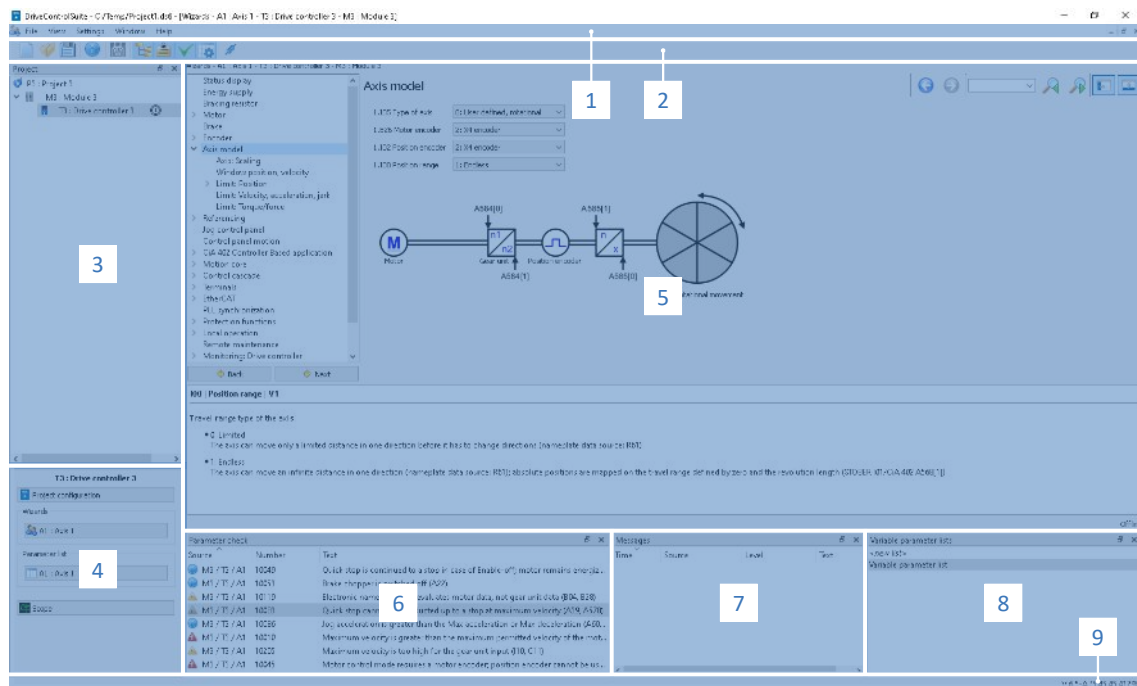


Abb. 3: DS6: Programmoberfläche

Nr.	Bereich	Beschreibung
1	Menüleiste	Über die Menüs <b>Datei</b> , <b>Ansicht</b> , <b>Einstellungen</b> und <b>Fenster</b> können Sie Projekte öffnen und speichern, Programmfenster ein- und ausblenden, die Oberflächensprache sowie Zugriffslevel auswählen und im Arbeitsbereich zwischen verschiedenen Fenstern wechseln.
2	Symbolleiste	Die Symbolleiste ermöglicht Ihnen schnellen Zugriff auf häufig benötigte Funktionen wie das Öffnen und Speichern von Projekten sowie das Ein- und Ausblenden von Fenstern in der Programmoberfläche.
3	Projektbaum	Der Projektbaum bildet die Struktur Ihres Antriebsprojekts in Form von Modulen und Antriebsreglern ab. Wählen Sie zuerst über den Projektbaum ein Element aus, um es über das Projektmenü zu bearbeiten.
4	Projektmenü	Das Projektmenü bietet Ihnen unterschiedliche Funktionen zur Bearbeitung von Projekt, Modul und Antriebsregler an. Das Projektmenü passt sich an das Element an, das Sie im Projektbaum ausgewählt haben.
5	Arbeitsbereich	Im Arbeitsbereich öffnen sich die verschiedenen Fenster, über die Sie ihr Antriebsprojekt bearbeiten können, wie z. B. der Projektierungsdialog, die Assistenten, die Parameterliste oder das Analysewerkzeug Scope.
6	Parameterprüfung	Die Parameterprüfung weist auf Auffälligkeiten und Unstimmigkeiten hin, die bei der Plausibilitätsprüfung der berechenbaren Parameter festgestellt wurden.
7	Meldungen	Die Einträge in den Meldungen protokollieren den Verbindungs- und Kommunikationszustand der Antriebsregler, systemseitig abgefangene Falscheingaben, Fehler beim Öffnen eines Projekts oder Regelverstöße in der grafischen Programmierung.
8	Variable Parameterlisten	Über variable Parameterlisten können Sie beliebige Parameter zur schnellen Übersicht in individuellen Parameterlisten zusammenstellen.
9	Statusleiste	In der Statusleiste finden Sie Angaben zur Software-Version und erhalten bei Prozessen wie dem Laden von Projekten weitere Informationen zur Projektdatei, zu den Geräten sowie zum Fortschritt des Prozesses.







## 7.1.1 Ansicht konfigurieren

Sie können in der DriveControlSuite die Sichtbarkeit und Anordnung von Bereichen und Fenstern ändern, um beispielsweise bei der Arbeit mit kleineren Bildschirmen den verfügbaren Platz im Arbeitsbereich zu optimieren.

### Bereiche ein-/ausblenden

Nutzen Sie die Symbole in der Symbolleiste oder die Einträge im Menü *Ansicht*, um bestimmte Bereiche in der DriveControlSuite nach Bedarf ein- oder auszublenden.

Symbol	Eintrag	Beschreibung
–	Zurücksetzen	Setzt die Ansicht auf Werkeinstellungen zurück.
	Projekt	Blendet das Fenster Projekt (Projektbaum, Projektmenü) ein/aus.
	Meldungen	Blendet das Fenster Meldungen ein/aus.
	Parameterprüfung	Blendet das Fenster Parameterprüfung ein/aus.
	Variable Parameterlisten	Blendet das Fenster Variable Parameterlisten ein/aus.

### Bereiche anordnen und gruppieren

Sie können die einzelnen Bereiche über Drag-and-Drop abdocken und neu anordnen: Wenn Sie ein abgedocktes Fenster an den Rand der DriveControlSuite ziehen, können Sie es dort in einem farblich hervorgehobenen Bereich entweder neben oder auf einem anderen Fenster loslassen, um es neu anzudocken.

Wenn Sie das Fenster auf einem anderen Fenster loslassen, werden die zwei Bereiche in einem Fenster zusammengefügt, in dem Sie über Register zwischen den Bereichen wechseln können.

## 7.1.2 Navigation über sensitive Schaltbilder

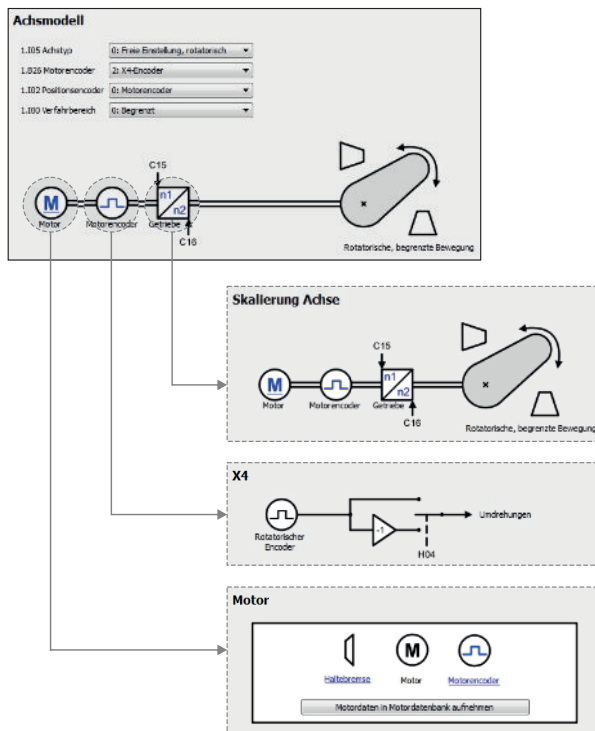


Abb. 4: DriveControlSuite: Navigation über Textlinks und Symbole

Um Ihnen die Bearbeitungsreihenfolgen von Soll- und Istwerten, die Verwendung von Signalen oder bestimmte Anordnungen von Antriebskomponenten grafisch zu verdeutlichen und die Konfiguration zugehöriger Parameter zu erleichtern, werden diese auf den jeweiligen Assistentenseiten des Arbeitsbereichs in Form von Schaltbildern dargestellt.

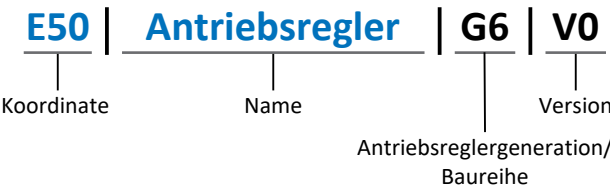
Blau eingefärbte Textlinks oder klickbare Symbole kennzeichnen programminterne Verlinkungen. Diese verweisen auf die jeweils zugehörigen Assistentenseiten und sind somit behilflich, weiterführende Detailseiten mit nur einem Klick zu erreichen.

## 7.2 Bedeutung der Parameter

Über Parameter passen Sie die Funktionen des Antriebsreglers an Ihre individuelle Anwendung an. Parameter visualisieren darüber hinaus aktuelle Istwerte (Istgeschwindigkeit, Istdrehmoment ...) und lösen Aktionen wie z. B. Werte speichern, Phasen testen usw. aus.

### Parameterkennung-Lesart

Eine Parameterkennung setzt sich aus nachfolgenden Elementen zusammen, wobei auch Kurzformen, d. h. die ausschließliche Angabe einer Koordinate oder die Kombination aus Koordinate und Name möglich sind.



### 7.2.1 Parametergruppen

Parameter werden thematisch einzelnen Gruppen zugeordnet. Die STÖBER Antriebsregler der 6. Generation unterscheiden nachfolgende Parametergruppen.

Gruppe	Thema
A	Antriebsregler, Kommunikation, Zykluszeiten
B	Motor
C	Maschine, Geschwindigkeit, Drehmoment/Kraft, Komparatoren
D	Sollwert
E	Anzeige
F	Klemmen, analoge und digitale Ein- und Ausgänge, Bremse
G	Technologie – Teil 1 (applikationsabhängig)
H	Encoder
I	Motion (sämtliche Bewegungseinstellungen)
J	Fahrsätze
K	Steuertafel
L	Technologie – Teil 2 (applikationsabhängig)
M	Profile (applikationsabhängig)
N	Zusatzfunktionen (applikationsabhängig; z. B. erweitertes Nockenschaltwerk)
P	Kundenspezifische Parameter (Programmierung)
Q	Kundenspezifische Parameter, instanzabhängig (Programmierung)
R	Fertigungsdaten von Antriebsregler, Motor, Bremsen, Motoradapter, Getriebe und Getriebemotor
S	Safety (Sicherheitstechnik)
T	Scope
U	Schutzfunktionen
Z	Störungszähler

Tab. 3: Parametergruppen

## 7.2.2 Parameterarten und Datentypen

Neben der thematischen Sortierung in einzelne Gruppen gehören alle Parameter einem bestimmten Datentyp und einer Parameterart an. Der Datentyp eines Parameters wird in der Parameterliste, Tabelle Eigenschaften angezeigt. Die Zusammenhänge zwischen Parameterarten, Datentypen und deren Wertebereich entnehmen Sie nachfolgender Tabelle.

Datentyp	Parameterart	Länge	Wertebereich (dezimal)
INT8	Ganzzahl oder Auswahl	1 Byte (vorzeichenbehaftet)	-128 – 127
INT16	Ganzzahl	2 Byte (1 Wort, vorzeichenbehaftet)	-32768 – 32767
INT32	Ganzzahl oder Position	4 Byte (1 Doppelwort, vorzeichenbehaftet)	-2147483648 – 2147483647
BOOL	Binärzahl	1 Bit (intern: LSB in 1 Byte)	0, 1
BYTE	Binärzahl	1 Byte (vorzeichenlos)	0 – 255
WORD	Binärzahl	2 Byte (1 Wort, vorzeichenlos)	0 – 65535
DWORD	Binärzahl oder Parameteradresse	4 Byte (1 Doppelwort, vorzeichenlos)	0 – 4294967295
REAL32 (Typ single nach IEEE754)	Fließkommazahl	4 Byte (1 Doppelwort, vorzeichenbehaftet)	$-3,40282 \times 10^{38} - 3,40282 \times 10^{38}$
STR8	Text	8 Zeichen	—
STR16	Text	16 Zeichen	—
STR80	Text	80 Zeichen	—

Tab. 4: Parameter: Datentypen, Parameterarten, mögliche Werte

### Parameterarten: Verwendung

- **Ganzzahl, Fließkommazahl**  
Bei allgemeinen Rechenprozessen  
Beispiel: Soll- und Istwerte
- **Auswahl**  
Zahlenwert, dem eine direkte Bedeutung zugeordnet ist  
Beispiel: Quellen für Signale oder Sollwerte
- **Binärzahl**  
Bit-orientierte Parameterinformationen, die binär zusammengefasst werden  
Beispiel: Steuer- und Statusworte
- **Position**  
Ganzzahl in Verbindung mit zugehörigen Einheiten und Nachkommastellen  
Beispiel: Ist- und Sollwerte von Positionen
- **Geschwindigkeit, Beschleunigung, Verzögerung, Ruck**  
Fließkommazahl in Verbindung mit zugehörigen Einheiten  
Beispiel: Ist- und Sollwerte für Geschwindigkeit, Beschleunigung, Verzögerung, Ruck
- **Parameteradresse**  
Referenzierung eines Parameters  
Beispiel: In F40 AO1 Quelle kann beispielsweise E08 Motorgeschwindigkeit parametrisiert werden
- **Text**  
Ausgaben oder Meldungen

## 7.2.3 Parametertypen

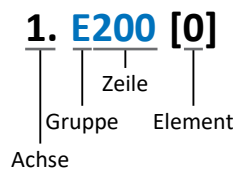
Bei Parametern werden folgende Typen unterschieden.

Parametertyp	Beschreibung	Beispiel
Einfache Parameter	Bestehen aus einer Gruppe und einer Zeile mit einem fest definierten Wert.	A21 Bremswiderstand R: Wert = 100 Ohm
Array-Parameter	Bestehen aus einer Gruppe, einer Zeile und mehreren fortlaufenden (gelisteten) Elementen, die dieselben Eigenschaften, jedoch unterschiedliche Werte besitzen.	A10 Zugriffslevel <ul style="list-style-type: none"> <li>A10[0] Zugriffslevel: Wert = Zugriffslevel über Bedienfeld</li> <li>A10[2] Zugriffslevel: Wert = Zugriffslevel über CANopen und EtherCAT</li> <li>A10[4] Zugriffslevel: Wert = Zugriffslevel über PROFINET</li> </ul>
Record-Parameter	Bestehen aus einer Gruppe, einer Zeile und mehreren fortlaufenden (gelisteten) Elementen, die unterschiedliche Eigenschaften und unterschiedliche Werte besitzen können.	A00 Werte speichern <ul style="list-style-type: none"> <li>A00[0] Starten: Wert = Aktion starten</li> <li>A00[1] Fortschritt: Wert = Aktionsfortschritt anzeigen</li> <li>A00[2] Ergebnis: Wert = Aktionsergebnis anzeigen</li> </ul>

Tab. 5: Parametertypen

## 7.2.4 Parameteraufbau

Jeder Parameter besitzt spezifische Koordinaten, die folgendem Aufbau entsprechen.



- Achse (optional)  
Achse, der ein achsspezifischer Parameter zugeordnet ist; entfällt bei globalen Parametern (Wertebereich: 1 – 4).
- Gruppe  
Gruppe, der ein Parameter thematisch angehört (Wertebereich: A – Z).
- Zeile  
Unterscheidet die Parameter innerhalb einer Parametergruppe (Wertebereich: 0 – 999).
- Element (optional)  
Elemente eines Array- oder Record-Parameters (Wertebereich: 0 – 16000).

## 7.2.5 Parametersichtbarkeit

Die Sichtbarkeit eines Parameters wird primär über das Zugriffslevel gesteuert, das Sie in der DriveControlSuite einstellen, sowie über die Eigenschaften, die Sie für den jeweiligen Antriebsregler projektieren (z. B. Hardware, Firmware und Applikation). Ein Parameter kann außerdem in Abhängigkeit von weiteren Parametern oder Einstellungen ein- bzw. ausgeblendet werden: Beispielsweise werden die Parameter einer Zusatzfunktion erst eingeblendet, sobald Sie die betreffende Zusatzfunktion aktivieren.

### Zugriffslevel

Die Zugriffsmöglichkeiten auf die einzelnen Parameter der Software sind hierarchisch gestaffelt und in einzelne Level unterteilt. Das bedeutet, Parameter können gezielt ausgeblendet und damit verbunden deren Konfigurationsmöglichkeiten ab einer bestimmten Ebene verriegelt werden.

Jeder Parameter besitzt jeweils ein Zugriffslevel für den Lesezugriff (Sichtbarkeit) sowie ein Zugriffslevel für den Schreibzugriff (Editierbarkeit). Folgende Level existieren:

- Level 0  
Elementare Parameter
- Level 1  
Wesentliche Parameter einer Applikation
- Level 2  
Wesentliche Parameter für den Service mit umfangreichen Diagnosemöglichkeiten
- Level 3  
Sämtliche für die Inbetriebnahme und Optimierung einer Applikation notwendigen Parameter

Parameter A10 Zugriffslevel regelt den generellen Zugriff auf Parameter:

- Über das Display des Antriebsreglers SD6 (A10[0])
- Über CANopen oder EtherCAT (A10[2])
- Über PROFINET (A10[3])

### Information

In der DriveControlSuite ausgeblendete Parameter können bei der Kommunikation via Feldbus weder gelesen noch geschrieben werden.

### Hardware

Welche Parameter Ihnen in der DriveControlSuite zur Verfügung stehen wird z. B. dadurch bestimmt, welche Baureihe Sie im Projektierungsdialog für den Antriebsregler wählen oder ob Sie ein Optionsmodul projektieren. Grundsätzlich werden Ihnen nur die Parameter angezeigt, die Sie zur Parametrierung der projektierten Hardware benötigen.

Beispielsweise kann ein Antriebsregler einen Encoder über die Klemme X120 auswerten, sofern das Klemmenmodul XI6 eingebaut wurde. Die zugehörige Auswertung wird über Parameter H120 aktiviert. Dieser Parameter ist jedoch nur dann sichtbar, wenn Klemmenmodul XI6 initial bei der Antriebsprojektierung ausgewählt wurde.

### Firmware

Durch die Weiterentwicklung und Pflege der Funktionen für STÖBER Antriebsregler der 6. Generation werden stets neue Parameter sowie neue Versionen bestehender Parameter in die DriveControlSuite sowie die Firmware implementiert. Die Parameter werden Ihnen in der Software entsprechend der verwendeten DriveControlSuite-Version und der projektierten Firmware-Version des jeweiligen Antriebsreglers angezeigt.

### Applikationen

Applikationen unterscheiden sich generell hinsichtlich Funktionen und deren Ansteuerung. Aus diesem Grund stehen mit jeder Applikation unterschiedliche Parameter zur Verfügung.

## 7.3 Signalquellen und Prozessdaten-Mapping

Die Übertragung von Steuersignalen und Sollwerten in der DriveControlSuite genügt folgenden Prinzipien.

### Signalquellen

Antriebsregler werden entweder über einen Feldbus, einen Mischbetrieb aus Feldbussystem und Klemmen oder ausschließlich über Klemmen angesteuert.

Ob die Steuersignale und Sollwerte der Applikation über einen Feldbus oder über Klemmen bezogen werden, konfigurieren Sie über entsprechende Auswahlparameter, die als Signalquellen bezeichnet werden.

Bei einer Ansteuerung über Feldbus werden Parameter als Quellen für Steuersignale oder Sollwerte ausgewählt, die Teil des anschließenden Prozessdaten-Mappings sein müssen; bei einer Ansteuerung über Klemmen werden die jeweiligen analogen oder digitalen Eingänge direkt angegeben.

### Prozessdaten-Mapping

Wenn Sie mit einem Feldbussystem arbeiten und die Quellparameter für Steuersignale und Sollwerte ausgewählt haben, konfigurieren Sie abschließend die feldbus-spezifischen Einstellungen, z. B. die Belegung der Prozessdatenkanäle für die Übertragung der Empfangs- und Sende-Prozessdaten.

## 7.4 Nichtflüchtiges Speichern

Sämtliche Projektierungen, Parametrierungen und damit verbundene Änderungen an Parameterwerten sind nach der Übertragung an den Antriebsregler wirksam, aber nur flüchtig gespeichert.

### Speichern auf einem Antriebsregler

Um die Konfiguration nichtflüchtig auf einem Antriebsregler zu speichern, haben Sie folgende Möglichkeiten:

- Konfiguration speichern über Assistent Werte speichern:  
Projektmenü > Bereich Assistenten > projektierte Achse > Assistent Werte speichern: Wählen Sie die Aktion Werte speichern
- Konfiguration speichern über die Parameterliste:  
Projektmenü > Bereich Parameterliste > projektierte Achse > Gruppe A: Antriebsregler > A00 Werte speichern: Setzen Sie den Parameter A00[0] auf den Wert 1: Aktiv
- Konfiguration speichern über die Bedieneinheit:  
Antriebsregler mit Bedieneinheit: Halten Sie die Speichertaste 3 Sekunden lang gedrückt

### Speichern auf allen Antriebsreglern innerhalb eines Projekts

Um die Konfiguration nichtflüchtig auf mehreren Antriebsreglern zu speichern, haben Sie folgende Möglichkeiten:

- Konfiguration speichern über die Symbolleiste:  
Symbolleiste > Symbol Werte speichern: Klicken Sie auf das Symbol Werte speichern
- Konfiguration speichern über das Fenster Online-Funktionen:  
Projektmenü > Schaltfläche Online-Verbindung > Fenster Online-Funktionen: Klicken Sie auf Werte speichern (A00)

#### Information

Schalten Sie den Antriebsregler während des Speicherns nicht aus. Wenn während des Speicherns die Versorgungsspannung des Steuerteils unterbrochen wird, startet der Antriebsregler beim nächsten Einschalten ohne lauffähige Konfiguration. In diesem Fall muss die Konfiguration erneut auf den Antriebsregler übertragen und nichtflüchtig gespeichert werden.

## 8 Inbetriebnahme

Nachfolgende Kapitel beschreiben die Inbetriebnahme eines CANopen-Netzwerks, bestehend aus einem CANopen-Master (Steuerung) und mehreren Antriebsreglern SD6 der Firma STÖBER, mithilfe der STÖBER Inbetriebnahme-Software DriveControlSuite.

Um die einzelnen Inbetriebnahmeschritte exakt nachvollziehen zu können, setzen wir folgende **beispielhafte** Systemumgebung voraus:

- Antriebsregler der Baureihe SD6 ab Firmware V 6.5-K
- Inbetriebnahme-Software DriveControlSuite ab Version 6.5-K

in Kombination mit

- Steuerung mit CANopen-Schnittstelle
- Automatisierungssoftware der Steuerung

### Die Inbetriebnahme gliedert sich in folgende Schritte ...

1. DriveControlSuite  
Projektieren Sie sämtliche Antriebsregler (Gerätesteuerung, Applikation, Prozessdaten, Achsmodell, ...), parametrieren Sie die allgemeinen CANopen-Einstellungen sowie die PDO-Übertragung und übertragen Sie im Anschluss Ihre Konfiguration auf die Antriebsregler Ihres CANopen-Netzwerks.
2. CANopen-Netzwerk  
Richten Sie im Anschluss Ihr CANopen-Netzwerk ein und nehmen Sie es mithilfe der Automatisierungssoftware der Steuerung in Betrieb.



## 8.1 DS6: Antriebsregler konfigurieren

Projektieren und konfigurieren Sie sämtliche Antriebsregler Ihres Antriebssystems in der DriveControlSuite (siehe auch [Programmoberfläche DS6 \[► 15\]](#)).

### Information

Führen Sie die im Nachfolgenden beschriebenen Schritte unbedingt in der vorgegebenen Reihenfolge aus!

Einige Parameter stehen in Abhängigkeit zueinander und werden Ihnen erst zugänglich, wenn Sie zuvor bestimmte Einstellungen getroffen haben. Folgen Sie den Schritten in der vorgegebenen Reihenfolge, damit Sie die Parametrierung vollständig abschließen können.

### 8.1.1 Projekt aufsetzen

Um sämtliche Antriebsregler und Achsen Ihres Antriebssystems über die DriveControlSuite konfigurieren zu können, müssen Sie diese im Rahmen eines Projekts erfassen.

#### 8.1.1.1 Antriebsregler und Achse projektieren

Erstellen Sie ein neues Projekt und projektieren Sie den ersten Antriebsregler samt zugehöriger Achse.

#### Neues Projekt anlegen

1. Starten Sie die DriveControlSuite.
2. Klicken Sie im Startbildschirm auf **Neues Projekt erstellen**.
  - ⇒ Das neue Projekt wird angelegt und der Projektierungsdialog für den ersten Antriebsregler öffnet sich.
  - ⇒ Die Schaltfläche **Antriebsregler** ist aktiv.

## Antriebsregler projektieren

1. **Register Eigenschaften:**  
 Stellen Sie die Beziehung zwischen Ihrem Schaltplan und dem zu projektierenden Antriebsregler in der DriveControlSuite her.  
 Referenz: Geben Sie das Referenzkennzeichen (Betriebsmittelkennzeichen) des Antriebsreglers an.  
 Bezeichnung: Benennen Sie den Antriebsregler eindeutig.  
 Version: Versionieren Sie Ihre Projektierung.  
 Beschreibung: Geben Sie gegebenenfalls unterstützende Zusatzinformationen wie die Änderungshistorie der Projektierung an.
2. **Register Antriebsregler:**  
 Wählen Sie die Baureihe und den Gerätetyp des Antriebsreglers.
3. **Register Optionsmodule:**  
 Kommunikationsmodul: Wählen Sie Kommunikationsmodul CA6.  
 Klemmenmodul: Wenn Sie den Antriebsregler im Mischbetrieb, d. h. zusätzlich zu CA6 über analoge und digitale Eingänge steuern, wählen Sie das entsprechende Klemmenmodul.  
 Sicherheitsmodul: Wenn der Antriebsregler Teil eines Sicherheitskreises ist, wählen Sie das entsprechende Sicherheitsmodul.
4. **Register Gerätesteuerung:**  
 Gerätesteuerung: Wählen Sie CiA 402.  
 Prozessdaten Rx: Wählen Sie CANopen Rx + Tx für die Übertragung der CANopen-Prozessdaten.  
 Prozessdaten Tx: Wählen Sie Keine Übertragung.

### Information

Stellen Sie sicher, dass Sie im Register Antriebsregler die korrekte Baureihe projektieren. Die projektierte Baureihe kann nachträglich nicht geändert werden.

## Achse projektieren

1. Klicken Sie auf Achse 1.
2. **Register Eigenschaften:**  
 Stellen Sie die Beziehung zwischen Ihrem Schaltplan und der zu projektierenden Achse in der DriveControlSuite her.  
 Referenz: Geben Sie das Referenzkennzeichen (Betriebsmittelkennzeichen) der Achse an.  
 Bezeichnung: Benennen Sie die Achse eindeutig.  
 Version: Versionieren Sie Ihre Projektierung.  
 Beschreibung: Geben Sie gegebenenfalls unterstützende Zusatzinformationen wie beispielsweise die Änderungshistorie der Projektierung an.
3. **Register Applikation:**  
 Wählen Sie CiA 402.
4. **Register Motor:**  
 Wählen Sie den Motortyp, den Sie über diese Achse betreiben. Wenn Sie mit Motoren von Fremdanbietern arbeiten, geben Sie die zugehörigen Motordaten zu einem späteren Zeitpunkt an.
5. Bestätigen Sie mit OK.

### 8.1.1.2 Sicherheitstechnik einrichten

Wenn der Antriebsregler Teil eines Sicherheitskreises ist, müssen Sie im nächsten Schritt die Sicherheitstechnik gemäß der im zugehörigen Handbuch beschriebenen Inbetriebnahmeschritte einrichten (siehe [Weiterführende Informationen](#) [► 70]).

### 8.1.1.3 Weitere Antriebsregler und Module anlegen

In der DriveControlSuite sind innerhalb eines Projekts alle Antriebsregler über Module gruppiert. Wenn Sie Ihrem Projekt einen neuen Antriebsregler hinzufügen, weisen Sie diesen immer einem bestehenden Modul zu. Gruppieren Sie beispielsweise Antriebsregler in einem Modul, wenn diese sich im selben Schaltschrank befinden oder gemeinsam denselben Maschinenteil betreiben.

#### Antriebsregler anlegen

1. Wählen Sie im Projektbaum Ihr Projekt P1 > Modul M1 > Kontextmenü **Neuen Antriebsregler anlegen**.  
⇒ Der Antriebsregler wird im Projektbaum angelegt und der Projektierungsdialog öffnet sich.
2. Projektieren Sie den Antriebsregler wie in **Antriebsregler und Achse projektieren** beschrieben.
3. Wiederholen Sie die Schritte für alle weiteren Antriebsregler, die Sie projektieren möchten.

#### Modul anlegen

1. Wählen Sie im Projektbaum Ihr Projekt P1 > Kontextmenü **Neues Modul anlegen**.  
⇒ Das Modul wird im Projektbaum angelegt.
2. Projektieren Sie das Modul wie in **Modul projektieren** [► 27] beschrieben.
3. Wiederholen Sie die Schritte für alle weiteren Module, die Sie projektieren möchten.

### 8.1.1.4 Modul projektieren

Benennen Sie Ihr Modul eindeutig, geben Sie das Referenzkennzeichen an und hinterlegen Sie optional Zusatzinformationen wie Version und Änderungshistorie des Moduls.

1. Markieren Sie im Projektbaum das Modul und klicken Sie im Projektmenü auf **Projektierung**.  
⇒ Der Projektierungsdialog für das Modul öffnet sich.
2. Stellen Sie die Beziehung zwischen Ihrem Schaltplan und dem Modul in der DriveControlSuite her.  
**Referenz:** Geben Sie das Referenzkennzeichen (Betriebsmittelkennzeichen) des Moduls an.  
**Bezeichnung:** Benennen Sie das Modul eindeutig.  
**Version:** Versionieren Sie das Modul.  
**Beschreibung:** Geben Sie gegebenenfalls unterstützende Zusatzinformationen wie beispielsweise die Änderungshistorie des Moduls an.
3. Bestätigen Sie mit **OK**.

### 8.1.1.5 Projekt projektieren

Benennen Sie Ihr Projekt eindeutig, geben Sie das Referenzkennzeichen an und hinterlegen Sie optional Zusatzinformationen wie Version und Änderungshistorie des Projekts.

1. Markieren Sie im Projektbaum das Projekt und klicken Sie im Projektmenü auf **Projektierung**.  
⇒ Der Projektierungsdialog für das Projekt öffnet sich.
2. Stellen Sie die Beziehung zwischen Ihrem Schaltplan und dem Projekt in der DriveControlSuite her.  
**Referenz:** Geben Sie das Referenzkennzeichen (Betriebsmittelkennzeichen) des Projekts an.  
**Bezeichnung:** Benennen Sie das Projekt eindeutig.  
**Version:** Versionieren Sie das Projekt.  
**Beschreibung:** Geben Sie gegebenenfalls unterstützende Zusatzinformationen wie beispielsweise die Änderungshistorie des Projekts an.
3. Bestätigen Sie mit **OK**.

## 8.1.2 Allgemeine CANopen-Einstellungen parametrieren

Parametrieren Sie die Feldbuskalierung sowie Baudrate Ihres CANopen-Netzwerks und definieren Sie die Node-ID für den jeweiligen Antriebsregler. Um die Kommunikation zwischen Steuerung und Antriebsregler im CANopen-Netzwerk zu überwachen, parametrieren Sie entweder die Guarding- oder die Heartbeat-Funktion.

- ✓ Sie haben das Kommunikationsmodul CA6 mit den Prozessdaten CANopen Rx + Tx projektiert.
- 1. Markieren Sie im Projektbaum den betreffenden Antriebsregler und klicken Sie im Projektmenü > Bereich Assistent auf die gewünschte projektierte Achse.
- 2. Wählen Sie Assistent CANopen.
- 3. A213 Feldbuskalierung:  
Belassen Sie den Default-Wert auf 1: Rohwert (Werte werden unverändert durchgereicht).
- 4. A82 CAN-Baudrate:  
Wählen Sie die Baudrate in Abhängigkeit von der Kabellänge (siehe [Baudrate und Kabellänge](#) [► 14]).
- 5. A83 Busadresse:  
Definieren Sie die Busadresse (Node-ID) des Antriebsreglers (Adressbereich: 1 – 127).
- 6. A203 Guard Time, A204 Life Time Factor:  
Um die Kommunikation zwischen Steuerung und Antriebsregler über die Guarding-Funktion (Node Guarding/ Life Guarding) zu überwachen, parametrieren Sie die Guard Time und den Life Time Factor.
  - 6.1. A203 Guard Time:  
Definieren Sie die Zeitspanne zwischen 2 aufeinanderfolgenden RTRs der Steuerung (Wertebereich: 1 – 4000 ms; 0 = inaktiv).
  - 6.2. A204 Life Time Factor:  
Definieren Sie die Anzahl der ausbleibenden RTRs der Steuerung, ab der der Antriebsregler in den Zustand Störung wechselt (Wertebereich: 1 – 255; 0 = inaktiv).
- 7. A210 Producer Heartbeat Time:  
Um die Kommunikation zwischen Steuerung und Antriebsregler über die Heartbeat-Funktion zu überwachen, definieren Sie die Zeitspanne zwischen 2 aufeinanderfolgenden Heartbeat-Nachrichten des Antriebsreglers (Wertebereich: 1 – 65535 ms; 0 = inaktiv).

### Information

Um die Kommunikation zwischen Steuerung und Antriebsregler im CANopen-Netzwerk zu überwachen, können Sie entweder die Guarding-Funktion oder die Heartbeat-Funktion nutzen. Da Guarding und Heartbeat dieselbe COB-ID und somit dieselbe Priorität haben, kann im CANopen-Netzwerk zeitgleich nur eine der beiden Überwachungsfunktionen genutzt werden, d. h. die jeweils andere Funktion muss deaktiviert werden.

Nähere Informationen zu den Überwachungsfunktionen finden Sie unter [ERROR CONTROL: Error Control Objects](#) [► 55].

## 8.1.3 PDO-Übertragung konfigurieren

Die PDO-Kommunikation ermöglicht pro Übertragungsrichtung (RxPDO, TxPDO) den gleichzeitigen Betrieb von bis zu 4 unabhängigen PDO-Kanälen, von denen jeder jeweils 1 PDO mit bis zu 6 Kommunikationsparametern übertragen kann. Die Zuordnung der Parameter zu den PDO-Kanälen ist frei konfigurierbar, solange die resultierende Datenlänge pro Kanal 8 Byte nicht überschreitet.

Zur Laufzeitoptimierung empfiehlt STÖBER die Verwendung der Voreinstellungen: Per Default sind 2 PDO-Kanäle pro Übertragungsrichtung aktiv, da es die meisten Anwendungsfälle abdeckt und die Laufzeitauslastung reduziert.

### 8.1.3.1 RxPDO anpassen

- ✓ Sie haben die allgemeinen CANopen-Einstellungen konfiguriert.
- 1. Markieren Sie im Projektbaum den betreffenden Antriebsregler und klicken Sie im Projektmenü > Bereich Assistent auf die gewünschte projektierte Achse.
- 2. Wählen Sie Assistent CANopen > Empfangs-Prozessdaten RxPDO.
- 3. Überprüfen Sie die Voreinstellungen und/oder konfigurieren Sie die Prozessdaten Ihren Anforderungen entsprechend.
  - 3.1. A221[0] COB-ID:  
COB-ID des 1. RxPDO entsprechend des Predefined Connection Set. Ändern Sie die COB-ID ausschließlich dann, wenn Sie vom Standard abweichen möchten (siehe [Dynamic Distribution](#) [► 43]).
  - 3.2. A221[1] Transmission Type:  
Definieren Sie die Übertragungsart für das 1. RxPDO.
  - 3.3. A225[0] 1. mapped Parameter – A225[5] 6. mapped Parameter:  
Definieren Sie die Zielparameter, deren Werte der Antriebsregler mittels des 1. RxPDO von der Steuerung empfängt. Die jeweilige Position (1. – 6.) gibt Auskunft über die Empfangsreihenfolge.
  - 3.4. Resultierende Datenlänge:  
Gesamtlänge der zu übertragenden Parameter des ersten RxPDO. Der Wert darf 8 Byte nicht überschreiten. Um den Wert einzuhalten, ändern Sie gegebenenfalls die Art oder Anzahl der zu übertragenden Parameter in diesem Kanal.
- 4. Wenn Sie weitere RxPDO-Kanäle nutzen möchten, wiederholen Sie das Vorgehen für den jeweiligen RxPDO-Kanal.

### 8.1.3.2 TxPDO anpassen

- ✓ Sie haben die allgemeinen CANopen-Einstellungen konfiguriert.
- 1. Markieren Sie im Projektbaum den betreffenden Antriebsregler und klicken Sie im Projektmenü > Bereich Assistent auf die gewünschte projektierte Achse.
- 2. Wählen Sie Assistent CANopen > Sende-Prozessdaten TxPDO.
- 3. Überprüfen Sie die Voreinstellungen und/oder konfigurieren Sie die Prozessdaten Ihren Anforderungen entsprechend.
  - 3.1. A229[0] COB-ID:  
COB-ID des 1. TxPDO entsprechend des Predefined Connection Set. Ändern Sie die COB-ID ausschließlich dann, wenn Sie vom Standard abweichen möchten (siehe [Dynamic Distribution](#) [► 43]).
  - 3.2. A229[1] Transmission Type:  
Definieren Sie die Übertragungsart für das 1. TxPDO.
  - 3.3. A229[2] Inhibit Time:  
Definieren Sie den zeitlichen Mindestabstand für den Versand von 2 aufeinanderfolgenden TxPDO.
  - 3.4. A229[3] Event Timer  
Wenn Sie den asynchronen Transmission Type 254 gewählt haben, definieren Sie die Zeit, nach deren Ablauf das 1. TxPDO gesendet wird, auch wenn keine RxPDO empfangen wurden.
  - 3.5. A233[0] – A233[5] 1. mapped Parameter – 6. mapped Parameter:  
Definieren Sie die Quellparameter, deren Werte der Antriebsregler mittels des 1. TxPDO an die Steuerung sendet. Die jeweilige Position (1. – 6.) gibt Auskunft über die Sendereihenfolge.
  - 3.6. Resultierende Datenlänge:  
Gesamtlänge der zu übertragenden Elemente des ersten TxPDO. Der Wert darf 8 Byte nicht überschreiten. Um den Wert einzuhalten, ändern Sie gegebenenfalls die Art oder Anzahl der zu übertragenden Elemente in diesem Kanal.
- 4. Wenn Sie weitere TxPDO-Kanäle nutzen möchten, wiederholen Sie das Vorgehen für den jeweiligen TxPDO-Kanal.

## 8.1.4 Konfiguration übertragen und speichern

Um die Konfiguration auf einen oder mehrere Antriebsregler zu übertragen und zu speichern, müssen Sie Ihren PC und die Antriebsregler über das Netzwerk verbinden.



### WARNUNG!

#### Personen- und Sachschaden durch Achsbewegung!

Wenn eine Online-Verbindung der DriveControlSuite zum Antriebsregler besteht, können Änderungen der Konfiguration zu unerwarteten Achsbewegungen führen.

- Ändern Sie die Konfiguration nur, wenn Sie Blickkontakt zur Achse haben.
- Stellen Sie sicher, dass sich keine Personen oder Gegenstände im Verfahrbereich befinden.
- Bei Zugriff über Fernwartung muss eine Kommunikationsverbindung zwischen Ihnen und einer Person vor Ort mit Blickkontakt zur Achse bestehen.

### Information

Bei der Suche werden via IPv4-Limited-Broadcast alle Antriebsregler innerhalb der Broadcast-Domain ausfindig gemacht.

Voraussetzungen für das Auffinden eines Antriebsreglers im Netzwerk:

- Netzwerk unterstützt IPv4-Limited-Broadcast
- Alle Antriebsregler und der PC sind im selben Subnetz (Broadcast-Domain)

### 8.1.4.1 Konfiguration übertragen

Die Schritte für die Übertragung der Konfiguration variieren in Abhängigkeit von der Sicherheitstechnik.

#### Antriebsregler ohne Sicherheitsmodul SE6

- ✓ Sie haben die vordefinierten Testbewegungsgrößen auf Plausibilität verifiziert.
  - ✓ Die Antriebsregler sind eingeschaltet.
1. Markieren Sie im Projektbaum das Modul, unter dem Sie Ihre Antriebsregler erfasst haben, und klicken Sie im Projektmenü auf **Online-Verbindung**.
    - ⇒ Der Dialog **Verbindung hinzufügen** öffnet sich. Alle via IPv4-Limited-Broadcast gefundenen Antriebsregler werden angezeigt.
  2. Register **Direktverbindung** > **Spalte IP-Adresse**:  
 Aktivieren Sie die betreffenden IP-Adressen und bestätigen Sie Ihre Auswahl mit **OK**.
    - ⇒ Das Fenster **Online-Funktionen** öffnet sich. Sämtliche Antriebsregler, die über die zuvor ausgewählten IP-Adressen angeschlossen sind, werden angezeigt.
  3. Wählen Sie den Antriebsregler, auf den Sie eine Konfiguration übertragen möchten und ändern Sie die Auswahl der Übertragungsart von **Lesen** in **Senden**.
  4. Ändern Sie die Auswahl **Neuen Antriebsregler anlegen**:  
 Wählen Sie die Konfiguration, die Sie an den Antriebsregler übertragen möchten.
  5. Wiederholen Sie die Schritte 3 und 4 für alle weiteren Antriebsregler, auf die Sie eine Konfiguration übertragen möchten.
  6. Register **Online**:  
 Klicken Sie auf **Online-Verbindung herstellen**.
    - ⇒ Die Konfigurationen werden an die Antriebsregler übertragen.

### Antriebsregler mit Sicherheitsmodul SE6

- ✓ Sie haben die vordefinierten Testbewegungsgrößen auf Plausibilität verifiziert.
- ✓ Die Antriebsregler sind eingeschaltet.
- 1. Markieren Sie im Projektbaum das Modul, unter dem Sie Ihre Antriebsregler erfasst haben, und klicken Sie im Projektmenü auf **Online-Verbindung**.
  - ⇒ Der Dialog **Verbindung hinzufügen** öffnet sich. Alle via IPv4-Limited-Broadcast gefundenen Antriebsregler werden angezeigt.
- 2. Register **Direktverbindung** > Spalte **IP-Adresse**:  
Aktivieren Sie die betreffenden IP-Adressen und bestätigen Sie Ihre Auswahl mit **OK**.
  - ⇒ Das Fenster **Online-Funktionen** öffnet sich. Sämtliche Antriebsregler, die über die zuvor ausgewählten IP-Adressen angeschlossen sind, werden angezeigt.
- 3. Wählen Sie den Antriebsregler, auf den Sie eine Konfiguration übertragen möchten und ändern Sie die Auswahl der Übertragungsart von **Lesen** in **Senden**.
- 4. Ändern Sie die Auswahl **Neuen Antriebsregler anlegen**:  
Wählen Sie die Konfiguration, die Sie an den Antriebsregler übertragen möchten.
- 5. Wiederholen Sie die Schritte 3 und 4 für alle weiteren Antriebsregler, auf die Sie eine Konfiguration übertragen möchten.
- 6. Register **Online**:  
Klicken Sie auf **Online-Verbindung herstellen**.
  - ⇒ Die Konfigurationen werden an die Antriebsregler übertragen.
  - ⇒ Sie werden über einen Dialog aufgefordert, das Konfigurations-Tool PASmotion zu öffnen.
- 1. Bestätigen Sie den Dialog mit **Ja**.
  - ⇒ PASmotion öffnet sich.
- 2. Navigieren Sie in der Projektverwaltung von PASmotion zum Sicherheitsmodul des Antriebsreglers und öffnen Sie dieses mit einem Doppelklick.
  - ⇒ Der Dialog zur Kennwortabfrage öffnet sich.
- 3. Geben Sie das Kennwort ein und bestätigen Sie mit **OK**.
  - ⇒ Der Assistent zur Gerätesynchronisierung öffnet sich.
  - ⇒ Gerätekonfiguration und Konfiguration werden automatisch gegeneinander geprüft.
- 4. Stimmen die Konfigurationen überein, klicken Sie nach abgeschlossener Gerätesynchronisierung auf **Fertig**.
- 5. Optional: Stimmen die Konfigurationen nicht überein, klicken Sie nach abgeschlossener Gerätesynchronisierung auf **Weiter**.
  - 5.1. Bestätigen Sie die Produktionsnummer des Sicherheitsmoduls und klicken Sie auf **Weiter**.
  - 5.2. Geben Sie das Kennwort für die Konfiguration auf dem Sicherheitsmodul ein und klicken Sie auf **Weiter**.
  - 5.3. Klicken Sie auf **Upload**, um die Gerätekonfiguration in das Projekt zu übertragen.
  - 5.4. Klicken Sie nach erfolgreicher Übertragung auf **Fertig**.
- 6. Beenden Sie PASmotion.
  - ⇒ Die Sicherheitskonfiguration wird an die ausgewählten Antriebsregler übertragen.



### 8.1.4.2 Konfiguration speichern

- ✓ Sie haben die Konfiguration erfolgreich übertragen.
- 1. Fenster Online-Funktionen:  
Klicken Sie auf Werte speichern (A00).
  - ⇒ Das Fenster Werte speichern (A00) öffnet sich.
- 2. Klicken Sie auf Aktion starten.
  - ⇒ Die Konfiguration wird nichtflüchtig auf den Antriebsreglern gespeichert.
- 3. Schließen Sie das Fenster Werte speichern (A00).

#### Information

Damit die Konfiguration auf dem Antriebsregler wirksam wird, ist ein Neustart beispielweise beim erstmaligen Speichern der Konfiguration auf dem Antriebsregler erforderlich sowie bei Änderungen an der Firmware oder am Prozessdaten-Mapping.

### Antriebsregler neu starten

- ✓ Sie haben die Konfiguration nichtflüchtig auf dem Antriebsregler gespeichert.
- 1. Fenster Online-Funktionen:  
Klicken Sie auf Neu starten (A09).
  - ⇒ Das Fenster Neu starten (A09) öffnet sich.
- 2. Wählen Sie, welche der verbundenen Antriebsregler Sie neu starten möchten.
- 3. Klicken Sie auf Aktion starten.
- 4. Bestätigen Sie den Sicherheitshinweis mit OK.
  - ⇒ Das Fenster Neu starten (A09) schließt sich.
- ⇒ Die Feldbuskommunikation und die Verbindung zwischen DriveControlSuite und Antriebsreglern werden unterbrochen.
- ⇒ Die gewählten Antriebsregler starten neu.

## 8.2 CANopen-Netzwerk in Betrieb nehmen

Richten Sie im Anschluss Ihr CANopen-Netzwerk ein und nehmen Sie es mithilfe der Automatisierungssoftware der Steuerung in Betrieb.

## 9 Monitoring und Diagnose

Zur Überwachung sowie im Störfall stehen Ihnen unterschiedliche, nachfolgend beschriebene Monitoring- und Diagnosemöglichkeiten zur Verfügung.

### 9.1 Verbindungsüberwachung

Um einen Kommunikationsausfall erkennen zu können, aktivieren Sie eine der beiden Überwachungsfunktionen für die Kommunikation im CANopen-Netzwerk: Guarding (Node Guarding/Life Guarding) oder Heartbeat (siehe [Allgemeine CANopen-Einstellungen parametrieren](#) [► 28]).

Bei der Überwachung der Kommunikation im CANopen-Netzwerk via Guarding überwachen sich Steuerung und Antriebsregler gegenseitig, bei der Überwachung via Heartbeat überwacht die Steuerung den Antriebsregler. Nähere Informationen zur jeweiligen Überwachungsfunktion finden Sie unter [Guarding](#) [► 55] oder [Heartbeat](#) [► 56].

Wenn Sie die Guarding-Funktion nutzen und der Antriebsregler innerhalb einer gewissen Zeit keine RTR von der Steuerung empfängt, wertet der Antriebsregler dies als Kommunikationsfehler (Life Guarding Event) und löst im NMT-Zustand Operational die Störung 52: Kommunikation mit der Ursache 1: CAN Life Guarding Event aus. Das Zeitintervall, in dem der Antriebsregler ein RTR von der Steuerung erwartet, wird als Life Time bezeichnet und ist das Produkt aus Guard Time und Life Time Factor ( $\text{Life Time} = \text{A203} \times \text{A204}$ ).

# 9.2 LED-Anzeige

Die Antriebsregler verfügen über Diagnose-Leuchtdioden, die den Zustand der Feldbuskommunikation sowie die Zustände der physikalischen Verbindung visualisieren.

## 9.2.1 Zustand CANopen

2 Leuchtdioden auf der Geräteoberseite des Antriebsreglers geben Auskunft über die Verbindung zwischen CANopen-Master und -Slave sowie über den Zustand des Datenaustauschs. Dieser kann zusätzlich in Parameter A245 ausgelesen werden.

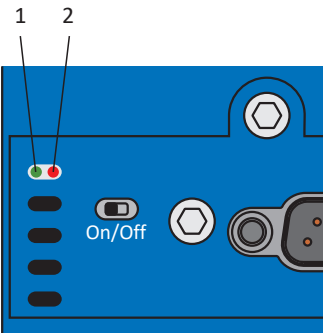


Abb. 5: Leuchtdioden für den CAN-Zustand

- 1 Grün: Run
- 2 Rot: Error

Grüne LED	Verhalten	NMT-Zustand	Beschreibung
	Aus	Initialisation	Keine Kommunikation zwischen CANopen-Master und -Slave; CAN-Bus wird initialisiert, die Konfiguration startet und gespeicherte Werte werden geladen
	Blinken	Pre-Operational	Keine PDO-Kommunikation zwischen CANopen-Master und -Slave; CAN-Bus ist aktiv, der Antriebsregler kann via SDO für Betrieb parametrisiert werden
	1-faches Blinken	Stopped	Kommunikationsdienste sind gestoppt (Ausnahme: Guarding, Heartbeat)
	Ein	Operational	Normalbetrieb: CAN-Bus ist aktiv, alle Kommunikationsdienste sind in Betrieb

Tab. 6: Bedeutung der grünen LED (Run)

Rote LED	Verhalten	Fehler	Beschreibung
	Aus	Kein Fehler, keine Warnung	Kein Fehler
	1-faches Blinken	Kommunikationsfehler im Zustand Operational (Warning Level)	Anschluss und Schirmung prüfen und gegebenenfalls korrigieren
	2-faches Blinken	Life Guarding Event	Guarding-Konfiguration der Steuerung prüfen
	3-faches Blinken	SYNC Error	SYNC-Konfiguration der Steuerung prüfen
	Ein	Bus-off	Keine Netzwerkverbindung

Tab. 7: Bedeutung der roten LED (Error)

## 9.3 Ereignisse

Der Antriebsregler verfügt über ein System zur Selbstüberwachung, das anhand von Prüfregelein das Antriebssystem vor Schaden schützt. Bei Verletzung der Prüfregelein wird ein entsprechendes Ereignis ausgelöst. Auf manche Ereignisse wie beispielsweise das Ereignis Kurz-/Erdschluss haben Sie als Anwender keinerlei Einflussmöglichkeit. Bei anderen können Sie Einfluss auf die Auswirkungen und Reaktionen nehmen.

Mögliche Auswirkungen sind:

- **Meldung:** Information, die von der Steuerung ausgewertet werden kann
- **Warnung:** Information, die von der Steuerung ausgewertet werden kann und nach Ablauf einer definierten Zeitspanne zu einer Störung wird, sofern die Ursache nicht behoben wurde
- **Störung:** Sofortige Reaktion des Antriebsreglers; das Leistungsteil wird gesperrt und die Achsbewegung nicht mehr durch den Antriebsregler gesteuert oder die Achse wird durch einen Schnellhalt oder eine Notbremsung zum Stillstand gebracht

### ACHTUNG!

#### Sachschaden durch Unterbrechung von Schnellhalt oder Notbremsung!

Tritt während der Ausführung eines Schnellhalts oder einer Notbremsung eine Störung auf oder wird STO aktiv, wird der Schnellhalt oder die Notbremsung unterbrochen. In diesem Fall kann die Maschine durch die unkontrollierte Achsbewegung beschädigt werden.

Ereignisse, deren Ursachen sowie geeignete Maßnahmen sind nachfolgend gelistet. Ist die Fehlerursache behoben, können Sie den Fehler in der Regel direkt quittieren. Ist stattdessen ein Neustart des Antriebsreglers erforderlich, finden Sie einen entsprechenden Hinweis in den Maßnahmen.

### Information

Um Steuerungsprogrammierern das Einrichten der Benutzerschnittstelle (HMI) zu erleichtern, finden Sie eine Liste der Ereignisse und deren Ursachen im STÖBER Download-Center unter <http://www.stoeber.de/de/downloads/>.

### 9.3.1 Ereignis 52: Kommunikation

Der Antriebsregler geht **in Störung**, wenn:

- A29 = 0: Inaktiv bei Gerätesteuerung Drive Based  
oder
- A540 = 0: Disable drive motor coasting bei Gerätesteuerung CiA 402

Reaktion:

- Das Leistungsteil wird gesperrt und die Achsbewegung nicht mehr durch den Antriebsregler gesteuert
- Die Bremsen fallen bei inaktivem Lüft-Override ein (F06)

Der Antriebsregler geht **mit einem Schnellhalt in Störung**, wenn:

- A29 = 1: Aktiv bei Gerätesteuerung Drive Based  
oder
- A540 = 2: Slow down on quick stop ramp bei Gerätesteuerung CiA 402

Reaktion:

- Die Achse wird durch einen Schnellhalt gestoppt
- Während des Schnellhalts bleiben die Bremsen gelüftet
- Am Ende des Schnellhalts wird das Leistungsteil gesperrt und die Achsbewegung nicht mehr durch den Antriebsregler gesteuert
- Die Bremsen fallen bei inaktivem Lüft-Override ein (F06)

Ursache		Prüfung und Maßnahme
1: CAN Life Guarding Event	Fehlende Remote Transmission Request (RTR)	Guarding-Einstellungen für CANopen-Master und Antriebsregler prüfen und gegebenenfalls korrigieren (A203, A204)
3: CAN Bus Off	Timing-Fehler	Anschluss prüfen und gegebenenfalls korrigieren; Kabel prüfen und gegebenenfalls tauschen; Baudrate in CANopen-Master und Antriebsregler prüfen und gegebenenfalls korrigieren (A82)

Tab. 8: Ereignis 52 – Ursachen und Maßnahmen

## 9.4 Parameter

Folgende Diagnoseparameter stehen Ihnen bei der Kommunikation via CANopen zur Verfügung.

### 9.4.1 A83 | Busadresse | G6 | V0

Busadresse des Antriebsreglers im CANopen-Netzwerk (Node-ID).

Entspricht dem Kommunikationsobjekt Node-ID nach CiA 301; Objekt 100B hex.

### 9.4.2 A245 | CAN-Diagnose | G6 | V0

Diagnoseinformationen des Antriebsreglers im CANopen-Netzwerk.

- Bit [0] – [2]: NMT-Zustand des Antriebsreglers im CANopen-Netzwerk (Network Management, NMT)  
000 bin = 0: Inaktiv; 001 bin = 1: Initialisation - Reset App.; 010 bin = 2: Initialisation - Reset Communication; 011 bin = 3: Initialisation - Boot-Up; 100 bin = 4: Pre-Operational; 101 bin = 5: Stopped; 110 bin = 6: Operational
- Bit [3]: Warning Level  
Kommunikationsfehler im Zustand Operational
- Bit [4]: Bus-off  
Anschlussfehler oder falsche Baudrate
- Bit [5]: Toggle-Bit SDO-Kanal 1  
Zustand wechselt mit jedem auf SDO-Kanal 1 empfangenen Frame
- Bit [6]: Speicherauslastung SDO-Kanal 1  
0 = Auslastung < 50 %; 1 = Auslastung ≥ 50 %
- Bit [7]: Toggle-Bit PDO-Kanal 1  
Zustand wechselt mit jedem auf PDO-Kanal 1 empfangenen Operational-Frame
- Bit [8]: Speicherauslastung PDO-Kanal 1  
0 = Auslastung < 50 %; 1 = Auslastung ≥ 50 %
- Bit [9]: Rote LED (Error)  
0 = Kein Fehler, keine Warnung; 1-faches Blinken = Warning Level; 2-faches Blinken = Life Guarding Event; 3-faches Blinken = SYNC Error; 1 = Bus-off
- Bit [10]: Grüne LED (Run)  
0 = Init; Blinken = Pre-Operational; 1-faches Blinken = Stopped; 1 = Operational
- Bit [11]: Fehler PDO-SYNC-Verhältnis
- Bit [12] – [15]: Reserviert

Sofern nicht anders angegeben, gilt: 0 = inaktiv; 1 = aktiv.

### 9.4.3 A246 | CANopen NMT-State | G6 | V1

NMT-Zustand des Antriebsreglers im CANopen-Netzwerk (Network Management, NMT).

- 0: Inaktiv
- 1: Initialisation - Reset App.
- 2: Initialisation - Reset Communication
- 3: Initialisation - Boot-Up
- 4: Pre-Operational
- 5: Stopped
- 6: Operational

## 10 Mehr zu CANopen?

Nachfolgende Kapitel fassen die wesentlichen Begriffe, Dienste und Beziehungen rund um CANopen zusammen.

### 10.1 CAN und CANopen

#### CAN

CAN (Controller Area Network) ist ein serielles Bussystem und industrieller Standard für Echtzeitanforderungen in der Automatisierungstechnik. Aufgrund seiner Echtzeit-Fähigkeit, seiner hohen Störresistenz sowie seiner guten Verfügbarkeit bedienen CAN-Bussysteme eine Vielzahl von Anwendungsbereichen und kommen vor allem dort zum Einsatz, wo hohe Übertragungsgeschwindigkeiten und eine einfache, kostengünstige Installation gefordert sind.

CAN wurde ursprünglich von der Firma Robert Bosch GmbH initiiert und wird bis zum aktuellen Zeitpunkt von der Organisation CAN in AUTOMATION e.V. (CiA) unterstützt. CAN ist eine offene Technologie, die seit 1993 in der Norm ISO 11898-1 standardisiert ist.

Die Aufgaben eines CAN-Netzwerks sind in sogenannten Schichten (Layer) definiert. Das eigentliche CAN-Protokoll entspricht Schicht 2 des ISO/OSI-Referenzmodells, der Sicherungsschicht (Data Link Layer). Das CAN-Protokoll verwaltet in dieser Schicht die Zugriffe der Teilnehmer, die über den CAN-Bus miteinander verbunden sind und ermöglicht so das Zusammenspiel der Technik unterschiedlicher Hersteller. Im CAN-Netzwerk werden Nachrichten entsprechend ihres CAN-Identifiers (CAN-ID) mit einer bestimmten Priorität übertragen.

#### CANopen

CANopen ist ein auf CAN basierendes Kommunikationsprotokoll für die Vernetzung in der Automatisierungstechnik. Es definiert die grundlegenden Kommunikationsmechanismen und die Funktionalität der Teilnehmer im CANopen-Netzwerk.

CANopen erweitert CAN um die Anwendungsschicht (Application Layer), die Schicht 7 des ISO/OSI-Referenzmodells entspricht und die im Kontext von CAN auch als CAL (CAN Application Layer) bezeichnet wird.

CANopen wird ebenfalls von der CiA gepflegt, wurde erstmals 1996 in der Norm EN 50325-4 standardisiert und später in die Normen IEC 61800-7-201 und IEC 61800-7-301 überführt.

#### 10.1.1 CANopen – Kommunikation

CANopen gruppiert Teilnehmer im CANopen-Netzwerk anhand ihrer Eigenschaften in Geräteklassen und stellt je Gerätekategorie ein eigenes Geräteprofil zur Verfügung, z. B. das Geräteprofil CiA 402 für die Ansteuerung von elektrischen Antrieben. Das Geräteprofil definiert die Funktionalität und den Aufbau des Objektverzeichnisses und ermöglicht so beispielsweise den standardisierten Zugriff auf die Kommunikationsobjekte (COB) bzw. Parameter der Gerätekategorie.

Dem Geräteprofil liegt ein Kommunikationsprofil zugrunde, z. B. das Kommunikationsprofil CiA 301, das die Inhalte der Kommunikationsobjekte definiert und somit die grundlegenden Kommunikationsmechanismen zwischen den Teilnehmern im CANopen-Netzwerk festlegt.

## 10.1.2 Objektverzeichnis

Das Objektverzeichnis beschreibt den vollständigen Funktionsumfang eines Teilnehmers im CANopen-Netzwerk entsprechend seines Geräteprofils. Im Objektverzeichnis werden beispielsweise standardisierte Datentypen, Kommunikationsobjekte und Geräteprofilobjekte aufgeführt. Neben den standardisierten Objekten gibt es auch einen Bereich für herstellerspezifische Objekte bzw. Parameter.

Über Servicedaten-Objekte (SDO) kann auf die Einträge im Objektverzeichnis zugegriffen werden, um die Geräteeigenschaften des Teilnehmers zu konfigurieren. Die Einträge im Objektverzeichnis können über den Index (Reihenadresse, 16 Bit) und bei Array- bzw. Record-Parametern über einen zusätzlichen Subindex (Spaltenadresse, 8 Bit) adressiert werden.

Index (hex)	Objekt
0000	Nicht verwendet
0001 – 001F	Statische Datentypen
0020 – 003F	Komplexe Datentypen
0040 – 005F	Herstellerspezifische Datentypen
0060 – 007F	Profilspezifische, statische Datentypen
0080 – 009F	Profilspezifische, komplexe Datentypen
00A0 – 0FFF	Reserviert
1000 – 1FFF	Kommunikationsprofil (CiA 301 und 302)
2000 – 5FFF	Herstellerspezifische Parameter
6000 – 9FFF	Parameter aus standardisierten Profilen (CiA 4xx)
A000 – AFFF	Netzwerkvariablen
B000 – FFFF	Reserviert

Tab. 9: Objektverzeichnis – Struktur

## 10.1.3 Netzwerkstruktur

Die Topologie eines CANopen-Netzwerks ist in der Regel linienförmig. Theoretisch können bis zu 127 Teilnehmer angesprochen werden, physikalisch sind 64 Teilnehmer im CANopen-Netzwerk möglich. Bei einer Baudrate von beispielsweise 50 kBaud eine Kabellänge von 1 km möglich, bei einer Baudrate von 1 MBaud ist eine Kabellänge von 25 m realistisch (siehe auch [Baudrate und Kabellänge](#) [► 14]). Generell stehen 9 Baudraten (10 kBaud – 1 MBaud) zur Verfügung (Parameter: A82).



## 10.2 CAN-Nachricht

Die Kommunikation über das CAN-Protokoll erfolgt in Form von Nachrichten, deren standardisierter Aufbau als Frame bezeichnet wird. Der CAN-Bus unterscheidet 4 Arten von Frames:

- Daten-Frame; für den Transport von Daten an einen anderen Teilnehmer
- Remote-Frame; für die Anforderung von Daten eines anderen Teilnehmers
- Error-Frame; für einen bekannten Übertragungsfehler
- Overload-Frame; als Zwangspause zwischen dem Versand von Daten- und Remote-Frames

### Aufbau einer CAN-Nachricht

Die nachfolgende Grafik zeigt den Aufbau einer CAN-Nachricht.

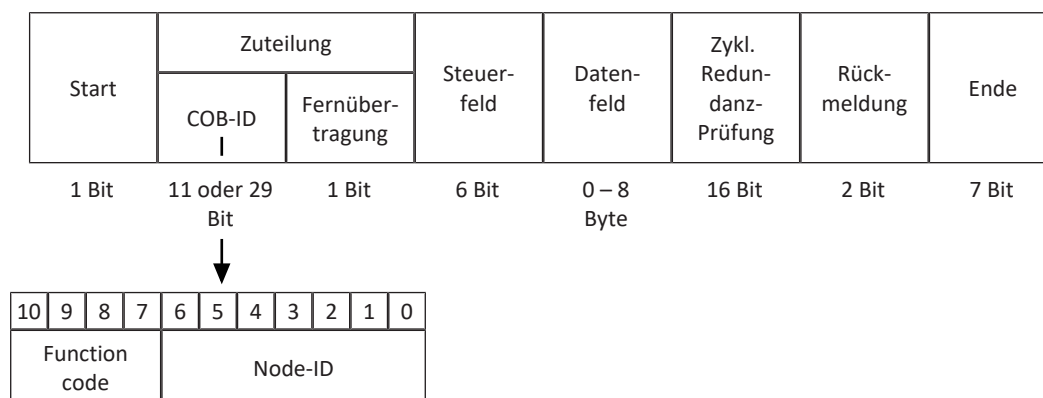


Abb. 6: CAN-Nachricht: Aufbau

### COB-ID

Die COB-ID (Communication Object Identifier) bestimmt die Priorität einer Nachricht im CANopen-Netzwerk (niedrige COB-ID = hohe Priorität). Die COB-ID setzt sich aus dem Function Code (Funktion des Kommunikationsobjekts) und der Node-ID (Busadresse des Teilnehmers) zusammen und kann über SDO-Zugriffe geändert werden. Eine COB-ID besteht standardmäßig aus 11 Bit (Base Frame Format), von denen Bit 7 – 10 mit dem Function Code belegt sind und Bit 0 – 6 mit der Node-ID, wodurch 2032 verschiedene logische Adressen kodiert und 2048 verschiedene Informationen versendet werden können.

Für die Vergabe der COB-ID existieren 2 Mechanismen: Beim Predefined Connection Set sind sämtliche Kommunikationsobjekte mit standardisierten Function Codes vorbelegt, bei Dynamic Distribution können für bestimmte Anwendungsfälle die vorbelegten Function Codes überschrieben werden.

### 10.2.1 Predefined Connection Set

Bei der Vergabe von COB-IDs über das Predefined Connection Set sind sämtliche Kommunikationsobjekte mit standardisierten Function Codes vorbelegt. Die Node-ID des jeweiligen Antriebsreglers wird im Anschluss addiert, wodurch sich eine eindeutige COB-ID ergibt. Diese Vorbelegung ermöglicht die einfache Inbetriebnahme eines CANopen-Netzwerks mit einer Steuerung und bis zu 127 Antriebsreglern (bzw. bei aktiven SDO-Kanälen 2 – 4 mit bis zu 31 Antriebsreglern) und deckt somit die meisten Anwendungsfälle ab.

Das Predefined Connection Set ist für sämtliche Kommunikationsobjekte (Broadcast-Objekte und Peer-to-Peer-Objekte) per Default aktiv, sofern für das jeweilige Kommunikationsobjekt keine manuelle COB-ID vergeben wurde.

Nachfolgende Tabellen zeigen die COB-ID der Kommunikationsobjekte entsprechend des Predefined Connection Set.

**Broadcast-Objekte**

Objekt	Function Code	COB-ID		Kommunikationsparameter (Index)	Priorität
NMT	0000	0 hex	0	—	Höchste
SYNC	0001	80 hex	128	1005	
TIME	0010	100 hex	256	—	

**Peer-to-Peer-Objekte**

Bei Verwendung des Predefined Connection Sets muss die Node-ID des betreffenden Teilnehmers zwischen 1 und 127 liegen bzw. bei aktiven SDO-Kanälen 2 – 4 zwischen 1 und 31 (Node-ID: A83).

Objekt	Function Code	COB-ID			Kommunikationsparameter		Priorität
					Index	DS6	
EMCY	0001	80 hex + Node-ID	81 hex – FF hex	129 – 255	1014 hex, 1015 hex	A207	Hoch
TxPDO1	0011	180 hex + Node-ID	181 hex – 1FF hex	385 – 511	1800 hex	A229[0]	
RxPDO1	0100	200 hex + Node-ID	201 hex – 27F hex	513 – 639	1400 hex	A221[0]	
TxPDO2	0101	280 hex + Node-ID	281 hex – 2FF hex	641 – 767	1801 hex	A230[0]	
RxPDO2	0110	300 hex + Node-ID	301 hex – 37F hex	769 – 895	1401 hex	A222[0]	
TxPDO3	0111	380 hex + Node-ID	381 hex – 3FF hex	897 – 1023	1802 hex	A231[0]	
RxPDO3	1000	400 hex + Node-ID	401 hex – 47F hex	1025 – 1151	1402 hex	A223[0]	
TxPDO4	1001	480 hex + Node-ID	481 hex – 4FF hex	1153 – 1279	1803 hex	A232[0]	
RxPDO4	1010	500 hex + Node-ID	501 hex – 5FF hex	1281 – 1407	1403 hex	A224[0]	
TxSDO1	1011	580 hex + Node-ID	581 hex – 59F hex	1409 – 1439	1200 hex	—	
RxSDO1	1100	600 hex + Node-ID	601 hex – 61F hex	1537 – 1567	1200 hex	—	
TxSDO2	1011	5A0 hex + Node-ID	5A1 hex – 5BF hex	1441 – 1471	1201 hex	A218[1]	
RxSDO2	1100	620 hex + Node-ID	621 hex – 63F hex	1539 – 1599	1201 hex	A218[0]	
TxSDO3	1011	5C0 hex + Node-ID	5C1 hex – 5DF hex	1473 – 1503	1202 hex	A219[1]	
RxSDO3	1100	640 hex + Node-ID	641 hex – 65F hex	1601 – 1631	1202 hex	A219[0]	
TxSDO4	1011	5E0 hex + Node-ID	5E1 hex – 5FF hex	1505 – 1535	1203 hex	A220[1]	
RxSDO4	1100	660 hex + Node-ID	661 hex – 67F hex	1633 – 1663	1203 hex	A220[0]	Niedrig
ERROR CONTROL	1110	700 hex + Node-ID	701 hex – 77F hex	1793 – 1919	1016 hex, 1017 hex	—	

## 10.2.2 Dynamic Distribution

Bei der Vergabe von COB-IDs über Dynamic Distribution können Sie die COB-IDs für bestimmte Kommunikationsobjekte manuell vergeben, entweder direkt über die DriveControlSuite oder über eine Steuerung, die als Distributor (DBT) die COB-IDs dynamisch vergibt. Der Steuerung muss dazu jederzeit ein Prozessdatenabbild aller Antriebsregler im CANopen-Netzwerk vorliegen.

Durch die manuelle Vergabe von COB-IDs via Dynamic Distribution können Sie die Priorität von Kommunikationsobjekten beeinflussen und so besondere Anwendungsfälle abdecken, um z. B. komplexe CANopen-Netzwerke mit unterschiedlichen Teilnehmern und Aufgaben zu optimieren. Da die COB-IDs eindeutig sein müssen, erfordert Dynamic Distribution mehr Planung als die Nutzung des Predefined Connection Set.

### Dynamic Distribution aktivieren

Grundsätzlich kann die COB-ID der folgenden Kommunikationsobjekte via Dynamic Distribution überschrieben werden:

- PDO 1 – 4 (RxPDO, TxPDO) (COB-ID: A221[0] – A224[0], A229[0] – A232[0])
- SDO 2 – 4 (RxSDO, TxSDO) (COB-ID: A218[0] – A220[0], A218[1] – A220[1])
- SYNC (COB-ID: A200)
- EMCY (COB-ID: A207)

Um eine COB-ID mithilfe der DriveControlSuite zu überschreiben, definieren Sie über den entsprechenden Parameter eine eindeutige COB-ID für das gewünschte Kommunikationsobjekt des jeweiligen Antriebsreglers. Anschließend übertragen Sie die Konfiguration auf den jeweiligen Antriebsregler und speichern diese nichtflüchtig auf dem Gerät (Parameter: A00).

Um eine COB-ID mithilfe der Steuerung zu überschreiben, muss die NMT-Zustandsmaschine des jeweiligen Antriebsreglers sich im NMT-Zustand Pre-Operational befinden. Die geänderten COB-ID werden für PDO beim Wechsel in den NMT-Zustand Initialising initialisiert und für SDO beim Wechsel in den NMT-Zustand Reset Communication. Damit die Änderungen auch bei Neustart der Antriebsregler wirksam bleiben, müssen diese nichtflüchtig gespeichert werden (Parameter: A00).

## 10.3 Kommunikationsobjekte

CANopen sieht grundsätzlich vor, Werte von Kommunikationsparametern über Kommunikationsobjekte zu lesen oder zu schreiben. Ein Schreibdienst wird als Domain Download, ein Lesedienst als Domain Upload bezeichnet.

Im CANopen-Netzwerk sind folgende Kommunikationsobjekten für die Datenübertragung von wesentlicher Bedeutung:

- Synchronization Objects (Synchronisationsobjekte, SYNC)  
... für die zeitliche Rasterung und Synchronisation der Teilnehmer
- Process Data Objects (Prozessdaten-Objekte, PDO)  
... für die Übertragung von Echtzeitdaten der Teilnehmer (Ist- und Sollwerte)
- Service Data Objects (Servicedaten-Objekte, SDO)  
... für den Zugriff auf das Objektverzeichnis der Teilnehmer zur Gerätekonfiguration
- Network Management Objects (Netzwerkmanagementobjekte, NMT)  
... für die Steuerung und Überwachung der NMT-Zustände der Teilnehmer
- Error Control Objects (Überwachungsobjekte, ERROR CONTROL)  
... für die Überwachung der Kommunikation der Teilnehmer (Guarding, Heartbeat)
- Emergency Objects (Fehlerobjekte, EMCY)  
... für die Überwachung der Gerätezustände der Teilnehmer
- Time Stamp Objects (Zeitstempelobjekte, TIME)  
... für die Übertragung der aktuellen Uhrzeit (Lokalzeit, keine definierte Zeitzone)

### Information

Kommunikationsmodul CA6 unterstützt keine Time Stamp Objects (TIME).

Bei den Kommunikationsobjekten sind Broadcast-Objekte von Peer-to-Peer-Objekten zu unterscheiden: Broadcast-Objekte werden von der Steuerung an alle Teilnehmer gleichzeitig versendet (NMT, SYNC, TIME). Peer-to-Peer-Objekte können in beide Senderichtungen ausgetauscht werden (SDO, PDO, EMCY, ERROR CONTROL).

### Information

In der DriveControlSuite ausgeblendete Parameter können bei der Kommunikation via Feldbus weder gelesen noch geschrieben werden.

### 10.3.1 SYNC: Synchronization Objects

SYNC-Objekte sind Broadcast-Objekte, die der zeitlichen Rasterung und Synchronisation der Teilnehmer im CANopen-Netzwerk dienen. SYNC-Objekte ermöglichen synchrone Transmission Types für PDO-Nachrichten, indem PDOs mit Bezug auf SYNC-Objekte empfangen oder gesendet werden können und so z. B. mehrere Eingänge parallel eingelesen oder Achsen synchronisiert werden können. SYNC-Nachrichten enthalten generell keine Daten.

#### Information

Bei gleichzeitiger Nutzung von CAN-Bus und IGB-Motionbus kann die CANopen-Kommunikation nicht mit der Steuerung synchronisiert werden.

### 10.3.2 PDO: Process Data Objects

Prozessdaten-Objekte sind Peer-to-Peer-Objekte, die der Übertragung zeitkritischer Echtzeitdaten der Teilnehmer dienen, wie z. B. Soll- und Istwerte bzw. Steuer- und Statusinformationen wie Sollpositionen, Verfahrgeschwindigkeiten oder Beschleunigungsvorgaben.

PDO ermöglichen den gleichzeitigen Zugriff auf mehrere Kommunikationsparameter, die über das Objektverzeichnis des jeweiligen Teilnehmers definiert sind. Bei der PDO-Übertragung werden keine Objekte adressiert, sondern die Werte der Kommunikationsparameter werden direkt an den jeweiligen Teilnehmer übertragen.

Das Prozessdaten-Mapping (PDO-Mapping) definiert, welche Kommunikationsparameter gesendet und empfangen werden. Beim Prozessdaten-Mapping ist frei wählbar, welche Kommunikationsparameter in welchem PDO versendet bzw. empfangen werden.

PDO werden über Prozessdatenkanäle (PDO-Kanäle) generell mit hoher Priorität übertragen. Aus Sicht des jeweiligen Teilnehmers unterscheidet man Empfangs-PDO (Receive-PDO, RxPDO) von Sende-PDO (Transmit-PDO, TxPDO).

Die Übertragungsart (Transmission Type) definiert, ob die PDO ereignis- oder zeitgesteuert und zyklisch oder azyklisch oder nur auf Anfrage eines anderen Teilnehmers gesendet werden (Remote Transmission Request, RTR). PDO werden nur übertragen, wenn sich der Teilnehmer im NMT-Zustand Operational befindet. Die Zuordnung sowie die Priorität von PDO-Nachrichten wird über die COB-ID definiert.

Für Informationen zur Skalierung siehe [Feldbuskalierung](#) [► 60].

#### 10.3.2.1 PDO-Mapping

Das Prozessdaten-Mapping (PDO-Mapping) definiert, welche Kommunikationsparameter gesendet und empfangen werden. Die Kommunikationsparameter aus dem Objektverzeichnis eines Teilnehmers werden dazu auf die jeweiligen PDO-Kanäle abgebildet.

Die PDO-Kommunikation ermöglicht pro Übertragungsrichtung (RxPDO, TxPDO) den gleichzeitigen Betrieb von bis zu 4 unabhängigen PDO-Kanälen, von denen jeder jeweils 1 PDO mit bis zu 6 Kommunikationsparametern übertragen kann. Die Zuordnung der Parameter zu den PDO-Kanälen ist frei konfigurierbar, solange die resultierende Datenlänge pro Kanal 8 Byte nicht überschreitet.

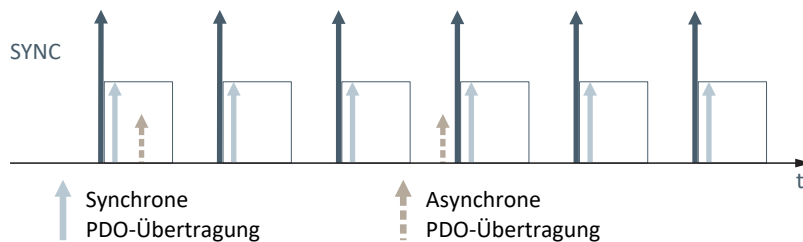
Zur Laufzeitoptimierung empfiehlt STÖBER die Verwendung der Voreinstellungen: Per Default sind 2 PDO-Kanäle pro Übertragungsrichtung aktiv, da es die meisten Anwendungsfälle abdeckt und die Laufzeitauslastung reduziert.

### 10.3.2.2 Transmission Type

Die Übertragungsart (Transmission Type) definiert, ob PDO zeitgesteuert (synchron) in Bezug auf SYNC-Objekte oder ereignisgesteuert (asynchron) übertragen werden. Die synchrone PDO-Übertragung kann zyklisch oder azyklisch erfolgen, woraus sich die folgenden Übertragungsarten ergeben:

- Synchron (zyklisch, azyklisch)
- Asynchron

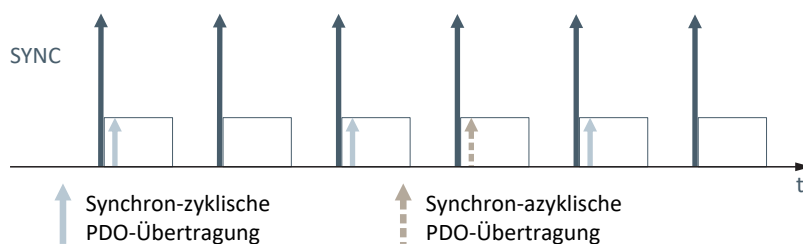
#### Synchrone und asynchrone PDO-Übertragung



Bei synchroner PDO-Übertragung werden die PDO-Nachrichten zeitgesteuert, d. h. abhängig von SYNC-Objekten bearbeitet. Beispielsweise kann ein RxPDO bei jedem oder bei jedem fünften SYNC-Objekt empfangen werden.

Bei asynchroner PDO-Übertragung werden die PDO-Nachrichten ereignisgesteuert, d. h. unabhängig von SYNC-Objekten bearbeitet. Beispielsweise kann ein RxPDO sofort empfangen und ein TxPDO sofort versendet werden, oder wenn die Übertragung durch einen Event Timer oder durch eine RTR ausgelöst wurde.

#### Synchron-zyklische und synchron-azyklische PDO-Übertragung



Bei synchroner PDO-Übertragung können die PDO-Nachrichten entweder zyklisch oder azyklisch übertragen werden. Zyklische PDO-Nachrichten werden synchron zu einer bestimmte Anzahl an SYNC-Objekten übertragen. Azyklische PDO-Nachrichten werden synchron zu einem SYNC-Objekt und ereignisgesteuert übertragen, wenn ein geräteinternes Ereignis auftritt, z. B. der Empfang von RxPDO.

#### Remote Transmission Requests (RTR)

Bei ereignisgesteuerter (asynchroner) PDO-Übertragung kann der Versand von PDO durch die Anfrage eines anderen Teilnehmers im CANopen-Netzwerk ausgelöst werden. Beispielsweise sendet bei der Überwachung der Teilnehmer im CANopen-Netzwerk via Guarding (Node Guarding/Life Guarding) die Steuerung in regelmäßigen Abständen Remote Transmission Requests (RTR) an die Teilnehmer im CANopen-Netzwerk, um den NMT-Zustand des jeweiligen Antriebsreglers anzufragen (siehe [Guarding](#) [► 55]).

#### TxPDO: Inhibit Time und Event Timer

Jeder TxPDO-Kanal eine Sendeverzögerungszeit (Inhibit Time), die unabhängig vom Transmission Type den zeitlichen Mindestabstand für den Versand von 2 aufeinanderfolgenden TxPDO festlegt.

Bei asynchroner PDO-Übertragung kann über einen Zeitgeber (Event Timer) je TxPDO-Kanal die Zeit definiert werden, nach deren Ablauf TxPDO gesendet werden, wenn für die ereignisgesteuerte PDO-Übertragung sonst kein Ereignis aufgetreten ist. Beispielsweise wird bei Transmission Type 254 der Versand von TxPDO durch den Empfang von RxPDO ausgelöst, spätestens jedoch nach Ablauf des Event Timers, wenn vorher keine neuen RxPDO empfangen werden.

### Transmission Type definieren

Der Transmission Type kann je Kanal und Senderichtung über den jeweiligen Parameter definiert werden (RxPDO: A221[1] – A224[1]; TxPDO: A229[1] – A232[1]).

Wert	Zyklisch	Azyklisch	Synchron	Asynchron	Beschreibung
0	–	✓	✓	–	RxPDO werden mit dem nächsten SYNC übernommen; TxPDO werden mit dem nächsten SYNC versendet
1 – 240	✓	–	✓	–	Wert = Anzahl der SYNC, bevor ein RxPDO empfangen und übernommen und ein TxPDO versendet wird
241 – 253					Reserviert
254	–	–	–	✓	RxPDO werden bei Empfang übernommen; TxPDO werden direkt im Anschluss versendet
255	–	–	–	✓	Reserviert

Tab. 10: Transmission Type

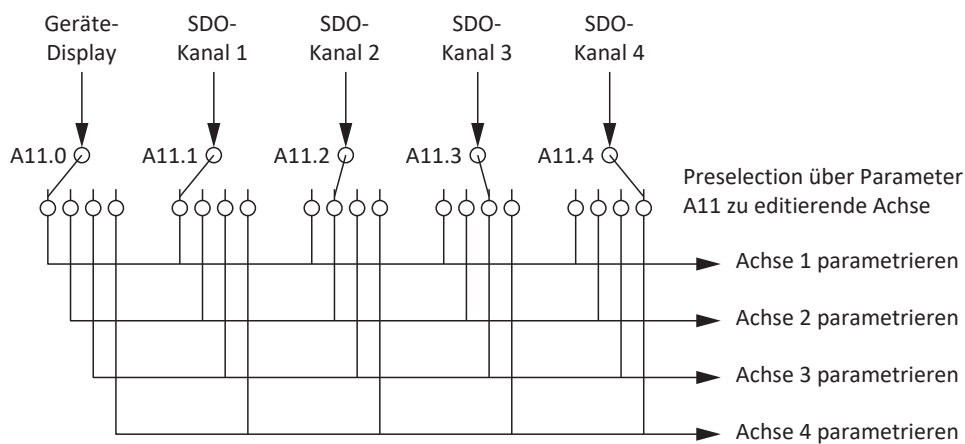
### 10.3.3 SDO: Service Data Objects

Servicedaten-Objekte sind Peer-to-Peer-Objekte, die der Übertragung zeitlich unkritischer Daten dienen und den Zugriff auf Einträge im Objektverzeichnis eines Teilnehmers ermöglichen, um dessen Geräteeigenschaften zu konfigurieren.

Eine SDO-Übertragung besteht aus Perspektive des Antriebsreglers immer mindestens aus einer RxSDO-Nachricht und einer TxSDO-Nachricht. In der RxSDO-Nachricht wählt die Steuerung über Index und Subindex einen Eintrag aus dem Objektverzeichnis des Antriebsreglers aus, um die Geräteeigenschaften zu konfigurieren. Mit einer TxSDO-Nachricht quittiert der Antriebsregler anschließend den Zugriff auf das Objektverzeichnis.

Für die SDO-Übertragung sind 4 unabhängige Kanäle verfügbar. SDO-Kanal 1 ist immer aktiv, die zugehörigen COB-IDs entsprechen dem Predefined Connection Set und können nicht verändert werden (RxSDO1 = 600 hex + Node-ID; TxSDO1 = 580 hex + Node-ID). SDO-Kanäle 2 – 4 sind per Default deaktiviert, die zugehörigen COB-IDs können bei Bedarf geändert werden.

Antriebsregler SD6 ermöglicht den Betrieb von bis zu 4 logischen Achsen, je Achse kann genau ein SDO-Kanal genutzt werden. Die Achsen werden nicht über Index und Subindex adressiert, sondern je SDO-Kanal über Parameter A11.



Abhängig von der Übertragungsart können via SDO grundsätzlich Daten beliebiger Länge übermittelt werden:

- Expedited Transfer  
... für die Übertragung von bis zu 4 Byte in einer einzigen Nachricht
- Segmented Transfer  
... für die Übertragung von mehr als 4 Byte verteilt auf mehrere Nachrichten

Für Informationen zur Skalierung siehe [Feldbuskalierung](#) [► 60].

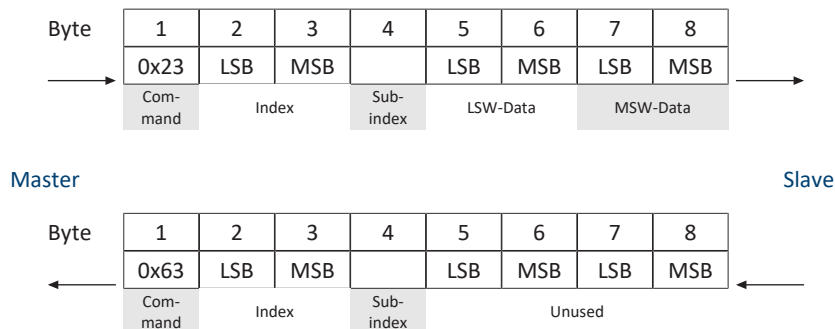


### 10.3.3.1 Expedited Transfer

Bei der SDO-Übertragung via Expedited Transfer (beschleunigte Übertragung) können bis zu 4 Byte Daten in einer einzigen Nachricht übertragen werden. Die Daten sind nach dem Intel-Format (Little-Endian) angeordnet, d. h., das kleinstwertige Byte wird an der Anfangsadresse gespeichert und zuerst übermittelt (vgl. Big-Endian oder Motorola-Format, bei dem die höchstwertige Komponente zuerst gesendet wird).

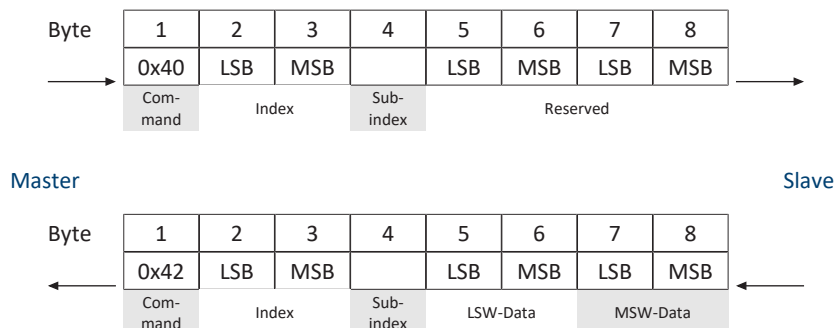
#### Parameter schreiben (Initiate Domain Download Request)

Die Steuerung (Master) veranlasst durch einen Initiate Domain Download Request den Schreibprozess eines Kommunikationsparameters. Die Anfrage wird durch eine Initiate Domain Download Response des Antriebsreglers (Slave) positiv quittiert.



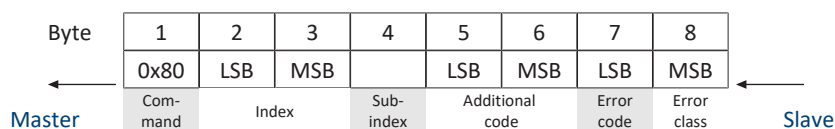
#### Parameter lesen (Initiate Domain Upload Request)

Die Steuerung (Master) veranlasst durch einen Initiate Domain Upload Request den Lesevorgang eines Kommunikationsparameters. Die Anfrage wird durch eine Initiate Domain Upload Response des Antriebsreglers (Slave) positiv quittiert.



#### Fehlermeldung (Abort Domain Transfer)

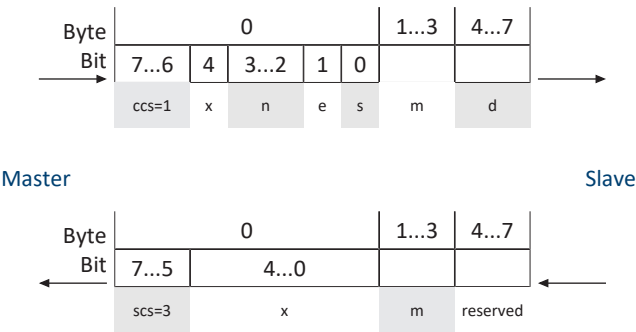
Ein Antriebsregler (Slave) beantwortet über einen Abort Domain Transfer die Requests Parameter schreiben oder Parameter lesen negativ (siehe SDO-Übertragung: Fehlercodes).



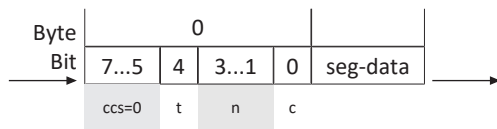
10.3.3.2 Segmented Transfer

Bei der SDO-Übertragung via Segmented Transfer (segmentierte Übertragung) können mehr als 4 Byte Daten verteilt auf mehrere Nachrichten übertragen werden. In einer ersten Initiate-Nachricht (Initiate SDO Download) wird die Gesamtzahl der zu übertragenden Byte übermittelt, im Anschluss folgen die Segmente (Download SDO Segment) mit je 1 Byte Steuer- und Protokollinformationen und bis zu 7 Byte Nutzdaten.

Initiate SDO Download Protocol

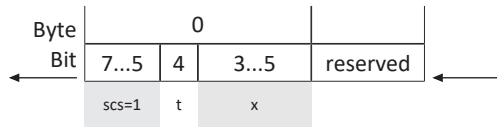


### Download SDO Segment Protocol



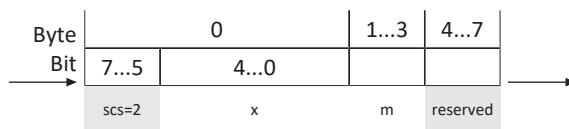
Master

Slave



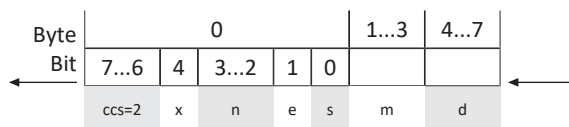
ccs	Client command specifier	0 = Download segment request
scs	Server command specifier	1 = Download segment response
n	Number of byte	Anzahl der Byte in "Segment data", die keine Nutzdaten enthalten. n = 0: Keine Angabe zu ungenutzten Daten
seg-data	Segment data	7 Byte Nutzdaten
c	Continue	<ul style="list-style-type: none"> <li>0 = Weitere Segmente folgen</li> <li>1 = Letztes Segment</li> </ul>
t	Toggle Bit	t = 0 bei Segment 1; muss bei jedem Segment wechseln. Identische Werte bei Request und Response.
x	Unused	x = 0

### Initiate SDO Upload Protocol



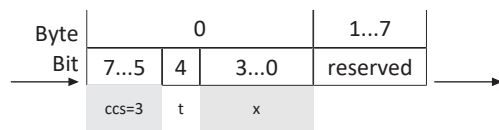
Master

Slave



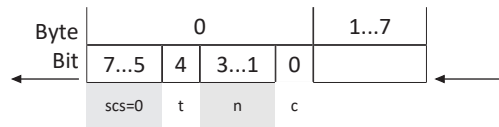
ccs	Client command specifier	2 = Initiate upload request
scs	Server command specifier	2 = Initiate upload response
n	Number of byte	Anzahl der Byte in "Data", die keine Nutzdaten enthalten. Wenn e = 0, s = 1, dann n = gültig, ansonsten n = 0
e	Transfer type	<ul style="list-style-type: none"> <li>0 = Normal transfer</li> <li>1 = Expedited transfer</li> </ul>
s	Size indicator	<ul style="list-style-type: none"> <li>0 = Wird nicht angezeigt</li> <li>1 = Wird angezeigt</li> </ul>
m	Multiplexor	= Index + Subindex
d	Data	<ul style="list-style-type: none"> <li>Wenn e = 0, s = 0, dann d = reserviert</li> <li>Wenn e = 0, s = 1, dann d = Anzahl der zu übertragenden Byte</li> <li>Wenn e = 1, s = 1, dann d = 4-n</li> </ul>
x	Unused	x = 0

## Upload SDO Segment Protocol



Master

Slave



ccs	Client command specifier	3 = Upload segment request
scs	Server command specifier	0 = Upload segment response
n	Number of byte	Anzahl der Byte in "Segment data", die keine Nutzdaten enthalten. n = 0: Keine Angabe zu ungenutzten Daten
seg-data	Segment data	7 Byte Nutzdaten
c	Continue	<ul style="list-style-type: none"> <li>0 = Weitere Segmente folgen</li> <li>1 = Letztes Segment</li> </ul>
t	Toggle Bit	t = 0 bei Segment 1; muss bei jedem Segment wechseln. Identische Werte bei Request und Response.
x	Unused	x = 0

## Beispiele

Segment Download mit 16 Byte Daten; Inhalt: 01, 02, 03 ... 10 hex

Client: IDReq:	21	idx	x	10 00 00 00	(ccs = 1, e = 0 = normal, s = 1 -> data = no of bytes)
Server: IDRes:	60	idx	x	00 00 00 00	
Client: DSegReq:	00	01 02 03 04 05 06 07			(ccs = 0, t = 0, n = 0, c = 0 -> all data bytes are used)
Server: DSegRes:	20	00 00 00 00 00 00 00			
Client: DSegReq:	10	08 09 0A 0B 0C 0D 0E			(ccs = 0, t = 1, n = 0, c = 0 -> all data bytes are used)
Server: DSegRes:	30	00 00 00 00 00 00 00			
Client: DSegReq:	0b	0F 10 00 00 00 00 00			(ccs = 0, t = 0, n = 5, c = 1 -> 5 data bytes are unused)
Server: DSegRes:	20	00 00 00 00 00 00 00			

Segment Upload mit 16 Byte Daten, Inhalt: 01, 02, 03 .. 10 hex

Client: IDUReq:	40	idx	x	00 00 00 00	(ccs = 2, rest = 0)
Server: IDURes:	41	idx	x	10 00 00 00	(scs = 2, x = 0, e = 0, s = 1 -> data contains no of bytes to be uploaded)
Client: USegReq:	60	00 00 00 00 00 00 00			(ccs = 3, t = 0)
Server: USegRes:	00	01 02 03 04 05 06 07			(scs = 0, t = 0, n = 0, c = 0 -> all data bytes are used)
Client: USegReq:	70	00 00 00 00 00 00 00			(ccs = 3, t = 1)
Server: USegRes:	10	08 09 0A 0B 0C 0D 0E			(scs = 0, t = 1, n = 0, c = 0 -> all data bytes are used)
Client: USegReq:	60	00 00 00 00 00 00 00			(ccs = 3, t = 0)
Server: USegRes:	0b	0F 10 00 00 00 00 00			(scs = 0, t = 0, n = 5, c = 1 -> 5 data bytes are unused)

10.3.4 NMT: Network Management Objects

Netzwerkmanagementobjekte (NMT-Objekte) sind Broadcast-Objekte, die der Steuerung und Überwachung der NMT-Zustände der Teilnehmer im CANopen-Netzwerk dienen. Jeder Teilnehmer im CANopen-Netzwerk besitzt eine Network Management State Machine (NMT-Zustandsmaschine), die sich in verschiedenen Zuständen befinden kann.

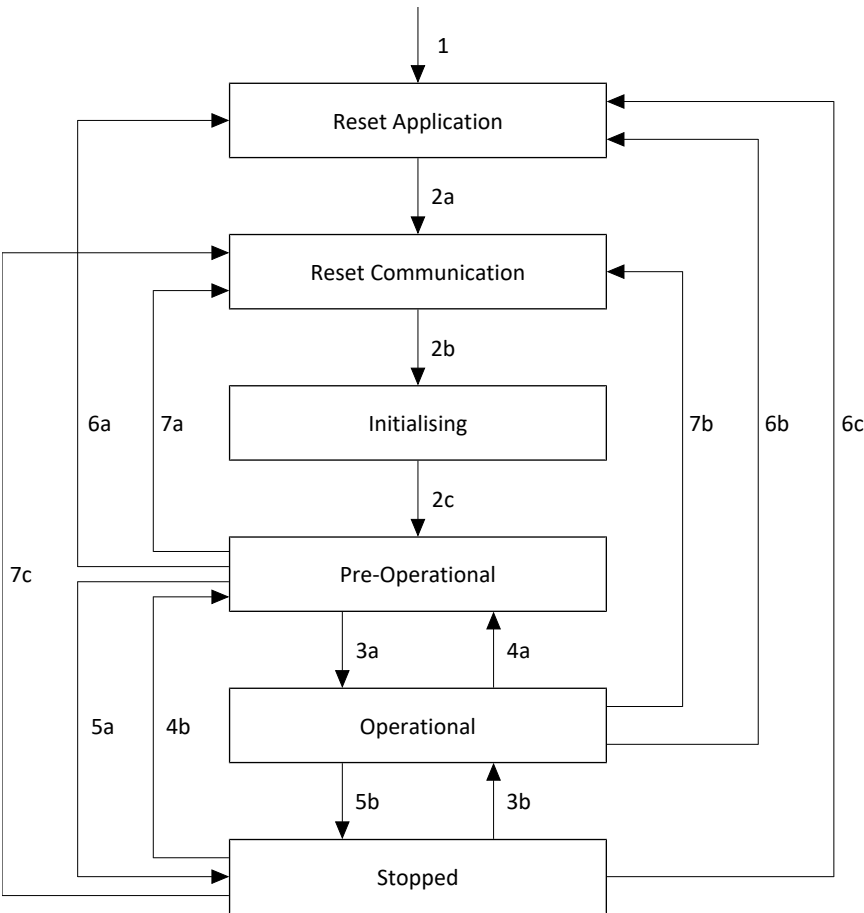


Abb. 7: Network Management State Machine: Zustände und Zustandswechsel

Zustände

Zustand		Beschreibung
Initialisation	Reset Application	Parameter des Geräteprofils und herstellerspezifische Parameter werden geladen
	Reset Communication	Parameter des Kommunikationsprofils werden geladen
	Initialising	CAN-Bus wird initialisiert
Pre-Operational		CAN-Bus ist aktiv, Antriebsregler kann via SDO für Betrieb parametrieren (PDO: inaktiv)
Operational		CAN-Bus ist aktiv, alle Kommunikationsdienste sind in Betrieb (PDO: aktiv)
Stopped		Kommunikationsdienste sind gestoppt (Ausnahme: Guarding, Heartbeat)

## Boot-Up

Das Einschalten der Versorgungsspannung startet den Geräteanlauf inklusive Initialisierung der NMT-Zustandsmaschine. Beim ersten Zustandswechsel von Initialisation zu Pre-Operational sendet der Antriebsregler einmalig eine Boot-Up-Nachricht an die Steuerung um zu signalisieren, dass er bereit für den Empfang von NMT-Kommandos ist.

Die Boot-Up-Nachricht besteht aus einer COB-ID und 1 Datenbyte. Die COB-ID der Boot-Up-Nachricht ist 1792 bzw. 700 hex + Node-ID, der Inhalt des Datenbyte ist 0. Da die Boot-Up-Nachricht nur einmalig beim Geräteanlauf gesendet wird, ist die COB-ID danach frei für andere Aufgaben und wird für die Überwachung der NMT-Zustände der Teilnehmer im CANopen-Netzwerk mittels Guarding oder Heartbeat (siehe [ERROR CONTROL: Error Control Objects \[► 55\]](#)).

## Zustandswechsel

Mithilfe von NMT-Objekten kann zwischen den Zuständen der NMT-Zustandsmaschine gewechselt werden. NMT-Objekte bestehen aus einer COB-ID und 2 Datenbyte. Die COB-ID für NMT-Objekte ist 0. Byte 1 des NMT-Objekts enthält den sogenannten Command Specifier (cs) und Byte 2 die Node-ID des Teilnehmers, wobei Node-ID 0 sämtliche Antriebsregler adressiert.

COB-ID	Command Specifier (cs)	Node-ID
0	1	0

Nr.	Beschreibung	Command Specifier (cs)
1	Versorgungsspannung einschalten	–
2a, 2b, 2c	Weiterschalten nach abgeschlossener Initialisierung	–
3a, 3b	Kommando NMT_Start_Remote_Node empfangen	1
4a, 4b	Kommando NMT_Enter_Pre_Operational empfangen	128
5a, 5b	Kommando NMT_Stop_Remote_Node empfangen	2
6	Kommando NMT_Reset_Node empfangen	129
7	Kommando NMT_Reset_Communication empfangen	130

## Verfügbarkeit der Kommunikationsobjekte

Nachfolgende Tabelle zeigt die Verfügbarkeit der Kommunikationsobjekte in Abhängigkeit vom NMT-Zustand.

Objekt	Initialisation	Pre-Operational	Operational	Stopped
Boot-Up	✓	–	–	–
PDO	–	–	✓	–
SDO	–	✓	✓	–
NMT	–	✓	✓	✓
SYNC	–	✓	✓	–
EMCY	–	✓	✓	–

## 10.3.5 ERROR CONTROL: Error Control Objects

Error Control Objects sind Broadcast-Objekte, die der Überwachung der NMT-Zustände der Teilnehmer im CANopen-Netzwerk dienen. CANopen bietet 2 Error Control Services für die Überwachung der NMT-Zustände, die auf der periodischen Übertragung von Nachrichten basieren: Guarding (Node Guarding/Life Guarding) und Heartbeat. Da Guarding und Heartbeat dieselbe COB-ID haben (1792 bzw. 700 hex + Node-ID), kann nur einer der beiden Dienste gleichzeitig verwendet werden.

### 10.3.5.1 Guarding

Bei der Überwachung der Kommunikation im CANopen-Netzwerk via Guarding (Node Guarding/Life Guarding) überwachen sich Steuerung und Antriebsregler gegenseitig. Um die Guarding-Funktion zu nutzen, muss die Heartbeat-Funktion deaktiviert werden (A210 = 0 ms).

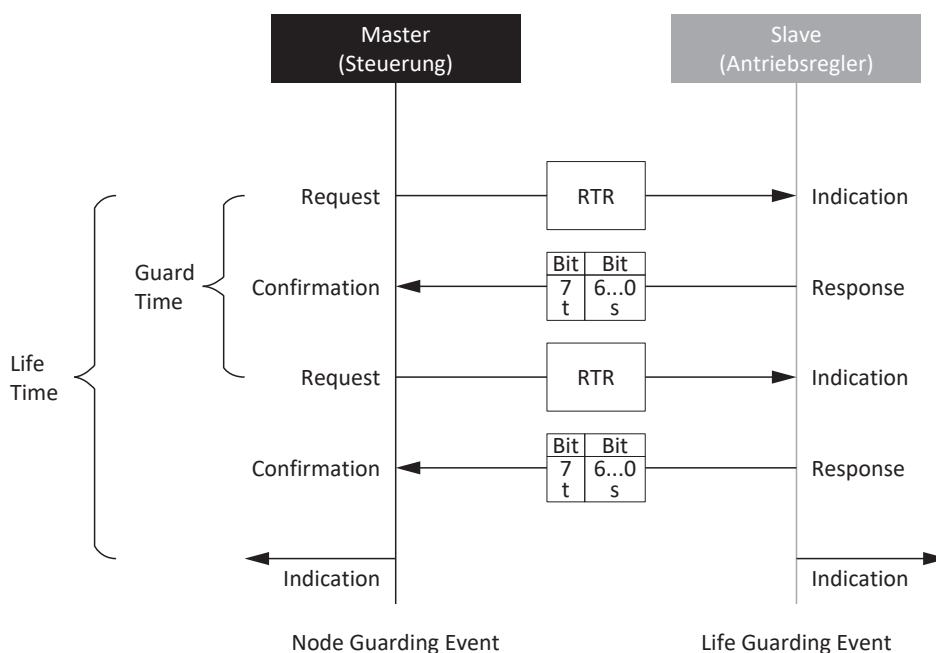


Abb. 8: Protokoll Guarding (Node Guarding/Life Guarding)

s = NMT-Zustand

t = Toggle Bit

#### Node Guarding

Bei der Überwachung via Node Guarding sendet die Steuerung in regelmäßigen Abständen Remote Transmission Requests (RTR) an sämtliche Antriebsregler im CANopen-Netzwerk, um den NMT-Zustand des jeweiligen Teilnehmers abzufragen. Das Zeitintervall, in dem die Steuerung RTRs an die Antriebsregler sendet, wird über die Guard Time definiert (Parameter: A203).

Der Antriebsregler erkennt die Anfrage der Steuerung und beantwortet diese mit einem Daten-Frame von 1 Byte Länge, von denen Bit 0 – 6 den NMT-Zustand angeben, während Bit 7 mit jeder Nachricht den Zustand wechselt (Toggle-Bit).

Die Steuerung empfängt das Daten-Frame und überprüft den NMT-Zustand des Antriebsreglers (Confirmation). Wenn die Steuerung von einem Antriebsregler kein Daten-Frame als Antwort auf die Anfrage via RTR erhält oder wenn die Informationen im Daten-Frame (NMT-Zustand, Toggle-Bit) nicht mit den erwarteten Informationen übereinstimmen, wertet die Steuerung dies als Kommunikationsfehler (Node Guarding Event).

## Life Guarding

Bei der Überwachung via Life Guarding erwartet ein Antriebsregler in regelmäßigen Abständen RTRs von der Steuerung. Wenn der Antriebsregler innerhalb einer gewissen Zeit keine RTR von der Steuerung empfängt, wertet der Antriebsregler dies als Kommunikationsfehler (Life Guarding Event) und wechselt in den Gerätezustand Störung.

Die Anzahl der ausbleibenden RTRs, ab der der Antriebsregler in Störung geht, wird über den Life Time Factor definiert (Parameter: A204).

Das Zeitintervall, in dem der Antriebsregler RTRs von der Steuerung erwartet, wird als Life Time bezeichnet und ist das Produkt aus Guard Time und Life Time Factor (Life Time = A203 × A204).

### 10.3.5.2 Heartbeat

Bei der Überwachung der Kommunikation im CANopen-Netzwerk via Heartbeat überwacht die Steuerung die Antriebsregler. Um die Heartbeat-Funktion zu nutzen, muss die Guarding-Funktion deaktiviert werden (A203 = 0 ms oder A204 = 0).

Der Vorteil der Heartbeat-Funktion gegenüber der Guarding-Funktion ist, dass keine RTRs zwischen Steuerung und Antriebsreglern ausgetauscht werden, wodurch die Auslastung des CANopen-Netzwerks erheblich sinkt.

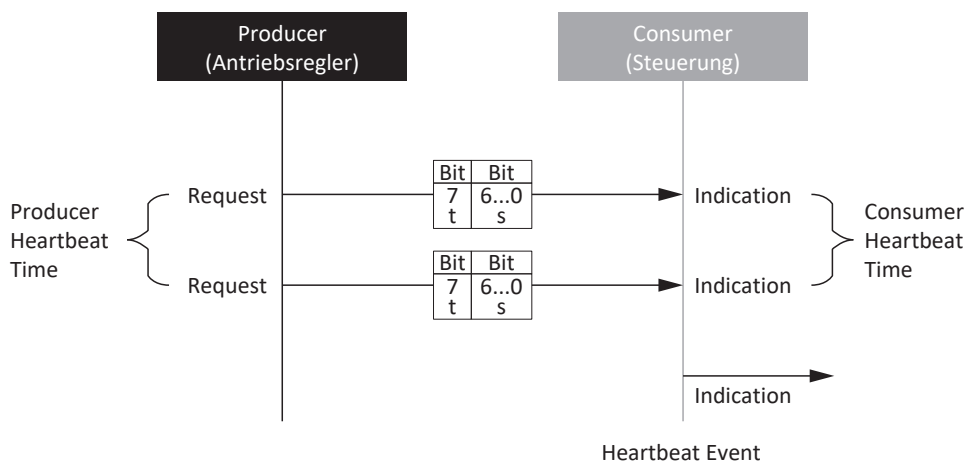


Abb. 9: Protokoll Heartbeat

s = NMT-Zustand

t = Toggle-Bit

Bei der Überwachung via Heartbeat-Funktion sendet ein Antriebsregler (Heartbeat-Producer) in regelmäßigen Abständen Heartbeat-Nachrichten, die seinen NMT-Zustand anzeigen. Das Zeitintervall, in dem der Antriebsregler die Heartbeat-Nachrichten sendet, wird über die Producer Heartbeat Time definiert (Parameter: A210).

Die Steuerung (Heartbeat-Consumer) erwartet die Heartbeat-Nachrichten in regelmäßigen Abständen (Consumer Heartbeat Time). Wenn die Steuerung innerhalb der definierten Consumer Heartbeat Time keine Heartbeat-Nachricht vom Antriebsregler empfängt, wird dies als Kommunikationsausfall gewertet (Heartbeat Event).



## 10.3.6 EMCY: Emergency Objects

Emergency-Objekte sind Peer-to-Peer-Objekte, die der Überwachung der Gerätezustände der Teilnehmer im Netzwerk dienen und bei geräteinternen Fehlern oder Störungen ausgelöst werden.

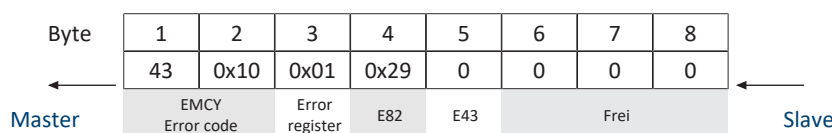
Wenn der EMCY-Dienst aktiv ist und ein Antriebsregler in den Gerätezustand Störung wechselt, sendet er eine EMCY-Nachricht an die Steuerung. Die EMCY-Nachricht enthält einen Fehler-Code (Error Code), der die Störung eindeutig identifiziert. Sobald die Störung behoben wurde und der Antriebsregler den entsprechenden Gerätezustand verlässt, sendet er eine weitere EMCY-Nachricht mit Fehler-Code 0 hex (KEIN FEHLER).

Durch diesen Mechanismus wird die Steuerung automatisch über das Auftreten und Verlassen des Störungszustands eines Antriebsreglers sowie über die zugehörige Störungsursache informiert.

Die COB-ID des EMCY-Objekts wird aus dem Wert 128 (80 hex) und aus der Busadresse (Node-ID) des Teilnehmers gebildet. Die COB-ID des SYNC-Objekts ist 128 (80 hex). Damit die Synchronisation der Prozessdaten nicht gestört wird, wird der Versand von EMCY-Objekten deaktiviert, wenn die Node-ID des Teilnehmers 0 ist (Parameter: A83).

### EMCY-Nachricht: Wechsel in den Störungszustand

Nachfolgende Grafik zeigt beispielhaft den Aufbau einer EMCY-Nachricht beim Wechsel in den Gerätezustand Störung.

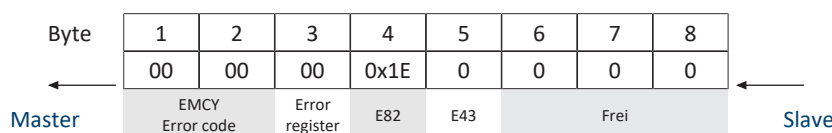


Byte 1 – 3 enthalten den Fehler-Code (Error Code) und das Fehlerregister (Error Register), Byte 4 – 5 enthalten die Werte der Parameter E82 Ereignis-Art und E43 Ereignis-Ursache.

Eine Tabelle mit den möglichen Fehler-Codes einer EMCY-Nachricht finden Sie unter [EMCY-Nachricht: Fehler-Codes](#) [Gerätestörung](#) [► 68].

### EMCY-Nachricht: Verlassen des Störungszustands

Nachfolgende Grafik zeigt beispielhaft den Aufbau einer EMCY-Nachricht beim Verlassen des Gerätezustands Störung.



Byte 1 – 3 enthalten den Fehler-Code 0 hex (KEIN FEHLER), Byte 4 enthält den Wert 1E hex für Parameter E82 = 30: Inaktiv.

## 10.4 Zykluszeiten

Mögliche Zykluszeiten entnehmen Sie der nachfolgenden Tabelle.

Typ	Zykluszeiten	Relevante Parameter
Feldbus CANopen, zyklische Kommunikation	1 ms, 2 ms, 4 ms, 8 ms	Einstellbar in A150

Tab. 11: Zykluszeiten

## 10.5 Aktionen ansteuern und ausführen

Um Aktionen via Feldbus ansteuern und ausführen zu können, müssen Sie vorab die Aktionsansteuerung in der DriveControlSuite aktivieren und die Prozessdaten um das Steuer-Byte und das Statuswort für Aktionen erweitern.

### Aktionsansteuerung aktivieren

1. Markieren Sie im Projektbaum den betreffenden Antriebsregler und klicken Sie im Projektmenü > Bereich Assistent auf die gewünschte projektierte Achse.
2. Wählen Sie Assistent Applikation CiA 402 > Zusatzfunktionen.
3. Aktivieren Sie die Option Aktionsansteuerung.

### Empfangs-Prozessdaten anpassen

1. Markieren Sie im Projektbaum den betreffenden Antriebsregler und klicken Sie im Projektmenü > Bereich Assistent auf die gewünschte projektierte Achse.
2. Wählen Sie Assistent CANopen > Empfangs-Prozessdaten RxPDO.
3. A225[0] – A225[5], A226[0] – A226[5]:  
Erweitern Sie die Empfangs-Prozessdaten um A75, das Steuer-Byte für die Ansteuerung von Aktionen.

### Sende-Prozessdaten anpassen

1. Markieren Sie im Projektbaum den betreffenden Antriebsregler und klicken Sie im Projektmenü > Bereich Assistent auf die gewünschte projektierte Achse.
2. Wählen Sie Assistent CANopen > Sende-Prozessdaten TxPDO.
3. A233[0] – A233[5], A234[0] – A234[5]:  
Erweitern Sie die Sende-Prozessdaten um A69, das Statuswort für die Ansteuerung von Aktionen.

## Aktion ausführen

Führen Sie im Anschluss die gewünschte Aktion aus. Berücksichtigen Sie hierfür eventuelle Voraussetzungen hinsichtlich des Gerätezustands sowie erforderliche weitere Maßnahmen nach Start der Aktion. Alle Voraussetzungen sowie nähere Informationen zu den einzelnen Aktionen entnehmen Sie den entsprechenden Parameterbeschreibungen in der DriveControlSuite.

Aktion auswählen	Gerätezustand herstellen	Aktion starten	Folgeschritt ausführen	Aktion abschließen (nach Fortschritt = 100 %)
0001 bin = Werte speichern (A00)	—	Execute ausführen (A75, Bit 0 =1)	—	Execute zurücknehmen (A75, Bit 0 = 0)
0011 bin = Schleppzeiger zurücksetzen (A37)				
0111 bin = Referenz löschen (I38)				
1000 bin = Endschalterspeicher löschen (I52)				
0010 bin = Neu starten (A09)	E48 ≠ 4: Freigegeben + E48 ≠ 7: Schnellhalt	Execute ausführen (A75, Bit 0 =1)	—	Execute zurücknehmen (A75, Bit 0 = 0)
1101 bin = Wicklung testen (B43)	E48 = 2: Einschaltbereit	Execute ausführen (A75, Bit 0 =1)	—	Execute zurücknehmen (A75, Bit 0 = 0)
1010 bin = Phasen testen (B40)	E48 = 2: Einschaltbereit	Execute ausführen (A75, Bit 0 =1)	Antriebsregler freigeben (E48 = 4: Freigegeben)	Execute zurücknehmen (A75, Bit 0 = 0) + Freigabe zurücknehmen
1011 bin = Motor einmessen (B41)				
1100 bin = Stromregler optimieren (B42)				
1110 bin = Stromregler optimieren (Stillstand) (B49)				
0100 bin = Bremsen testen (B300)				
0101 bin = Bremsen einschleifen (B301)				
0110 bin = Bremsen 2 einschleifen (B302)				
1001 bin = Bremsen testen (S18)				

Tab. 12: Aktion auswählen und ausführen

# 10.6      Feldbusskalierung

Über Parameter A213 definieren Sie in der Inbetriebnahme-Software DriveControlSuite die Skalierung sowohl für die zyklische Übertragung der Prozessdaten-Objekte als auch für die azyklische Übertragung der Servicedaten-Objekte im Netzwerk. Die Werte werden entweder umgerechnet und als Ganzzahl dargestellt oder entsprechend ihrer Datentypen unskaliert als Rohwert übertragen.

Unabhängig von den in Parameter A213 gewählten Einstellungen arbeiten sowohl die Konfiguration als auch die Firmware ausschließlich mit Rohwerten. Nachfolgende Grafik zeigt die Feldbusskalierung im Überblick.

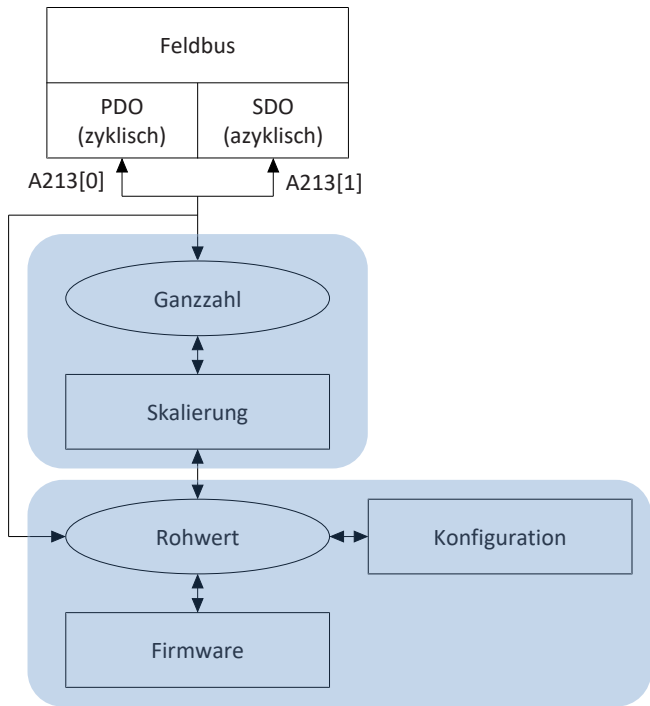


Abb. 10: Feldbusskalierung im Überblick

Bei der Übertragung als Ganzzahl kann die Anzahl der Dezimalstellen für alle Parameter definiert werden, die Positionen, Geschwindigkeiten, Beschleunigungen, Verzögerungen und Ruck betreffen. Für alle anderen Parameter ist die Anzahl der Dezimalstellen fest vorgegeben. Die Werte für die Skalierung werden in der DriveControlSuite bei den Eigenschaften eines Parameters ausgegeben. Nachfolgende Tabelle listet die Parameter, über die Sie die Anzahl der Dezimalstellen für die skalierte Übertragung festlegen können.

Skalierung	Achsmodell	Master-Achsmodell
Position	I06	G46
Geschwindigkeit (DB)	I66	G66
Geschwindigkeit (CiA)	A310	—
Beschleunigung, Verzögerung, Ruck (DB)	I67	G67
Beschleunigung, Verzögerung, Ruck (CiA)	A311	—

Tab. 13: Feldbusskalierung bei Ganzzahl: Parameter zur Definition der Dezimalstellen

# 11 Anhang

## 11.1 Unterstützte Kommunikationsobjekte

Nachfolgende Kapitel liefern Ihnen einen Überblick über die unterstützten Kommunikationsobjekte des standardisierten Profils CiA 301 CANopen application layer and communication profile für CANopen sowie deren Abbildung auf die Parameter von STÖBER.

Informationen zu den unterstützten Kommunikationsobjekten des Profils CiA 402 entnehmen Sie dem zugehörigen Applikationshandbuch.

### 11.1.1 CiA 301 CANopen: 1000 hex – 1FFFF hex

Nachfolgende Tabelle beinhaltet die unterstützten Kommunikationsobjekte des standardisierten Profils CiA 301 CANopen application layer and communication profile für CANopen sowie deren Abbildung auf die entsprechenden Parameter von STÖBER.

Index	Subindex	TxPDO	RxPDO	Name	Kommentar
1000 hex	0 hex	—	—	Device type	Konstanter Wert 20192 hex Bit 0 – 15: Device profile number, 192 hex = 402 Bit 16 – 23: Type, 2 hex = Servo drive Bit 24 – 31: Reserved
1001 hex	0 hex	—	—	Error register	
1002 hex	0 hex	—	—	Manufacturer status register	E48
1003 hex				Pre-defined error field	
1003 hex	0 hex	—	—	Number of errors	
1003 hex	1 hex – A hex	—	—	Standard error field	
1005 hex	0 hex	—	✓	COB-ID SYNC message	A200; Wertebereich 1 – 2047
1006 hex	0 hex	—	✓	Communication cycle period	A201; Wertebereich 0 – 32 000 000
1008 hex	0 hex	—	✓	Manufacturer device name	E50
1009 hex	0 hex	—	✓	Manufacturer hardware version	E52[1]
100A hex	0 hex	—	✓	Manufacturer software version	E53[3]
100C hex	0 hex	—	✓	Guard time	A203; Life time in ms = 100C hex * 100D hex
100D hex	0 hex	—	✓	Life time factor	A204; Life time in ms = 100C hex * 100D hex
1010 hex				Store parameters supported	Record mit 1 Element
1010 hex	0 hex	—	—	Highest sub-index supported	Konstanter Wert 1 hex
1010 hex	1 hex	—	✓	Save all parameters	A00[0]; ASCII save startet Werte speichern
1014 hex	0 hex	—	✓	COB-ID EMCY	A207

Index	Subindex	TxPDO	RxPDO	Name	Kommentar
1015 hex	0 hex	—	✓	Inhibit time EMCY	A208
1017 hex	0 hex	—	✓	Producer heartbeat time	A210
1018 hex				Identity object	Record mit 4 Elementen
1018 hex	0 hex	—	—	Highest sub-index supported	Konstanter Wert 4 hex
1018 hex	1 hex	—	—	Vendor ID	Hersteller-ID: B9 hex
1018 hex	2 hex	—	—	Product code	Nennleistung in Einheit 0,1 kW
1018 hex	3 hex	—	—	Revision number	SW-Build-Nummer
1018 hex	4 hex	—	—	Serial number	E52[2]
1020 hex				Verify Configuration	Record mit 2 Elementen
1020 hex	0 hex	—	—	Highest sub-index supported	Konstanter Wert 2 hex
1020 hex	1 hex	—	—	Configuration data	A211
1020 hex	2 hex	—	—	Configuration time	A212
1200 hex				1 <sup>st</sup> SDO server parameter	Record mit 2 Elementen
1200 hex	0 hex	—	—	Highest sub-index supported	Konstanter Wert 2 hex
1200 hex	1 hex	—	✓	COB-ID Client -> Server (rx)	
1200 hex	2 hex	—	✓	COB-ID Server -> Client (tx)	
1201 hex				2 <sup>nd</sup> SDO server parameter	Record mit 3 Elementen
1201 hex	0 hex	—	—	Highest sub-index supported	Anzahl Elemente: 3
1201 hex	1 hex	—	✓	COB-ID Client -> Server (rx)	A218[0]
1201 hex	2 hex	—	✓	COB-ID Server -> Client (tx)	A218[1]
1201 hex	3 hex	—	✓	Node-ID of the SDO client	A218[2]
1202 hex				3 <sup>rd</sup> SDO server parameter	Record mit 3 Elementen
1202 hex	0 hex	—	—	Highest sub-index supported	Konstanter Wert 3 hex
1202 hex	1 hex	—	✓	COB-ID Client -> Server (rx)	A219[0]
1202 hex	2 hex	—	✓	COB-ID Server -> Client (tx)	A219[1]
1202 hex	3 hex	—	✓	Node-ID of the SDO client	A219[2]
1203 hex				4 <sup>th</sup> SDO server parameter	Record mit 3 Elementen
1203 hex	0 hex	—	—	Highest sub-index supported	Konstanter Wert 3 hex
1203 hex	1 hex	—	✓	COB-ID Client -> Server (rx)	A220[0]
1203 hex	2 hex	—	✓	COB-ID Server -> Client (tx)	A220[1]
1203 hex	3 hex	—	✓	Node-ID of the SDO client	A220[2]
1400 hex				1 <sup>st</sup> RxPDO communication parameter	Record mit 2 Elementen
1400 hex	0 hex	—	—	Highest sub-index supported	Konstanter Wert 2 hex
1400 hex	1 hex	—	✓	COB-ID used by RxPDO	A221[0]
1400 hex	2 hex	—	✓	Transmission type	A221[1]; Wertebereich 1 - 240, 254

Index	Subindex	TxPDO	RxPDO	Name	Kommentar
1401 hex				2 <sup>nd</sup> RxPDO communication parameter	Record mit 2 Elementen
1401 hex	0 hex	—	—	Highest sub-index supported	Konstanter Wert 2 hex
1401 hex	1 hex	—	✓	COB-ID used by RxPDO	A222[0]
1401 hex	2 hex	—	✓	Transmission type	A222[1]; Wertebereich 1 - 240, 254
1402 hex				3 <sup>rd</sup> RxPDO communication parameter	Record mit 2 Elementen
1402 hex	0 hex	—	—	Highest sub-index supported	Konstanter Wert 2 hex
1402 hex	1 hex	—	✓	COB-ID used by RxPDO	A223[0]
1402 hex	2 hex	—	✓	Transmission type	A223[1]; Wertebereich 1 - 240, 254
1403 hex				4 <sup>th</sup> RxPDO communication parameter	Record mit 2 Elementen
1403 hex	0 hex	—	—	Highest sub-index supported	Konstanter Wert 2 hex
1403 hex	1 hex	—	✓	COB-ID used by RxPDO	A224[0]
1403 hex	2 hex	—	✓	Transmission type	A224[1]; Wertebereich 1 - 240, 254
1600 hex				1st RxPDO mapping parameter	Record mit 6 Elementen
1600 hex	0 hex	—	✓	Number of mapped application objects in RxPDO	Konstanter Wert 6 hex
1600 hex	1 hex – 6 hex	—	✓	Application objects	A225[0] – A225[5]
1601 hex				2nd RxPDO mapping parameter	Record mit 6 Elementen
1601 hex	0 hex	—	✓	Number of mapped application objects in RxPDO	Konstanter Wert 6 hex
1601 hex	1 hex – 6 hex	—	✓	Application objects	A226[0] – A226[5]
1602 hex				3rd RxPDO mapping parameter	Record mit 6 Elementen
1602 hex	0 hex	—	✓	Number of mapped application objects in RxPDO	Konstanter Wert 6 hex
1602 hex	1 hex – 6 hex	—	✓	Application objects	A227[0] – A227[5]
1603 hex				4th RxPDO mapping parameter	Record mit 6 Elementen
1603 hex	0 hex	—	—	Number of mapped application objects in RxPDO	Konstanter Wert 6 hex
1603 hex	1 hex – 6 hex	—	—	Application objects	A228[0] – A228[5]
1800 hex				1 <sup>st</sup> TxPDO communication parameter	Record mit 4 Elementen

Index	Subindex	TxPDO	RxPDO	Name	Kommentar
1800 hex	0 hex	—	—	Highest sub-index supported	Konstanter Wert 5 hex
1800 hex	1 hex	—	✓	COB-ID used by TxPDO	A229[0]
1800 hex	2 hex	—	✓	Transmission type	A229[1]; Wertebereich 1 – 240, 254
1800 hex	3 hex	—	✓	Inhibit time	A229[2]
1800 hex	5 hex	—	✓	Event timer	A229[3]
1801 hex				2 <sup>nd</sup> TxPDO communication parameter	Record mit 4 Elementen
1801 hex	0 hex	—	—	Highest sub-index supported	Konstanter Wert 5 hex
1801 hex	1 hex	—	✓	COB-ID used by TxPDO	A230[0]
1801 hex	2 hex	—	✓	Transmission type	A230[1]; Wertebereich 1 – 240, 254
1801 hex	3 hex	—	✓	Inhibit time	A230[2]
1801 hex	5 hex	—	✓	Event timer	A230[3]
1802 hex				3 <sup>rd</sup> TxPDO communication parameter	Record mit 4 Elementen
1802 hex	0 hex	—	—	Highest sub-index supported	Konstanter Wert 5 hex
1802 hex	1 hex	—	✓	COB-ID used by TxPDO	A231[0]
1802 hex	2 hex	—	✓	Transmission type	A231[1]; Wertebereich 1 – 240, 254
1802 hex	3 hex	—	✓	Inhibit time	A231[2]
1802 hex	5 hex	—	✓	Event timer	A231[3]
1803 hex				4 <sup>th</sup> TxPDO communication parameter	Record mit 4 Elementen
1803 hex	0 hex	—	—	Highest sub-index supported	Konstanter Wert 5 hex
1803 hex	1 hex	—	✓	COB-ID used by TxPDO	A232[0]
1803 hex	2 hex	—	✓	Transmission type	A232[1]; Wertebereich 1 – 240, 254
1803 hex	3 hex	—	✓	Inhibit time	A232[2]
1803 hex	5 hex	—	✓	Event timer	A232[3]
1A00 hex				1st TxPDO mapping parameter	Record mit 6 Elementen
1A00 hex	0 hex	—	✓	Number of mapped application objects in TxPDO	Konstanter Wert 6 hex
1A00 hex	1 hex – 6 hex	—	✓	Application objects	A233[0] - A233[5]
1A01 hex				2nd TxPDO mapping parameter	Record mit 6 Elementen
1A01 hex	0 hex	—	✓	Number of mapped application objects in TxPDO	Konstanter Wert 6 hex
1A01 hex	1 hex – 6 hex	—	✓	Application objects	A234[0] - A234[5]



Index	Subindex	TxPDO	RxPDO	Name	Kommentar
1A02 hex				3rd TxPDO mapping parameter	Record mit 6 Elementen
1A02 hex	0 hex	—	✓	Number of mapped application objects in TxPDO	Konstanter Wert 6 hex
1A02 hex	1 hex – 6 hex	—	✓	Application objects	A235[0] - A235[5]
1A03 hex				4th TxPDO mapping parameter	Array mit 6 Elementen
1A03 hex	0 hex	—	—	Number of mapped application objects in TxPDO	Konstanter Wert 6 hex
1A03 hex	1 hex – 6 hex	—	—	Application objects	A236[0] - A236[5]

Tab. 14: Kommunikationsobjekte CiA 301: 1000 hex – 1FFF hex

11.1.2      Herstellerspezifische Parameter: 2000 hex – 53FF hex

Index, Subindex und Berechnungsbeispiel

Information	
-------------	--

Index und Subindex müssen in dem von der Steuerung geforderten Format angegeben werden.

Information	
-------------	--

Die nachfolgend beschriebene Berechnung ist nur gültig für die Umrechnung der herstellerspezifischen Parameter.

Der Index berechnet sich aus der Gruppe und Zeile des Parameters nach folgender Formel:

$\text{Index} = 8192 + (\text{Nummer der Gruppe} \times 512) + \text{Nummer der Zeile}$

Der Subindex bei einfachen Parametern ist immer 0, bei Array- oder Record-Parametern entspricht der Subindex der Elementnummer des Parameters.

	Einfache Parameter	Array- oder Record-Parameter
Index	$8192 + (\text{Nummer der Gruppe} \times 512) + \text{Nummer der Zeile}$	
Subindex	0	Elementnummer

Tab. 15: Index und Subindex bei herstellerspezifischen Parametern

Berechnungsbeispiel

Berechnung für Parameter E200[0]:

$\text{Nummer der Gruppe} = 4$

$\text{Nummer der Zeile} = 200$

$\text{Index} = 8192 + (4 \times 512) + 200 = 10440 = 28C8 \text{ hex}$

$\text{Subindex} = 0 = 0 \text{ hex}$

### Kommunikationsobjekte

Nachfolgende Tabelle beinhaltet die unterstützten Kommunikationsobjekte sowie deren Abbildung auf die entsprechenden Parameter von STÖBER.

Index	Gruppe	Nummer	Parameter
2000 hex – 21FF hex	A: Antriebsregler	0	A00 – A511
2200 hex – 23FF hex	B: Motor	1	B00 – B511
2400 hex – 25FF hex	C: Maschine	2	C00 – C511
2600 hex – 27FF hex	D: Sollwert	3	D00 – D511
2800 hex – 29FF hex	E: Anzeigen	4	E00 – E511
2A00 hex – 2BFF hex	F: Klemmen	5	F00 – F511
2C00 hex – 2DFF hex	G: Technologie	6	G00 – G511
2E00 hex – 2FFF hex	H: Encoder	7	H00 – H511
3000 hex – 31FF hex	I: Motion	8	I00 – I511
3200 hex – 33FF hex	J: Fahrsätze	9	J00 – J511
3400 hex – 35FF hex	K: Steuertafel	10	K00 – K511
3600 hex – 37FF hex	M: Profile	12	M00 – M511
3E00 hex – 3FFF hex	P: Kundenspezifische Parameter	15	P00 – P511
4000 hex – 41FF hex	Q: Kundenspezifische Parameter, instanzabhängig	16	Q00 – Q511
4200 hex – 43FF hex	R: Fertigungsdaten	17	R00 – R511
4400 hex – 45FF hex	S: Sicherheit	18	S00 – S511
4600 hex – 47FF hex	T: Scope	19	T00 – T511
4800 hex – 49FF hex	U: Schutzfunktionen	20	U00 – U511
5200 hex – 53FF hex	Z: Störungszähler	25	Z00 – Z511

Tab. 16: Herstellerspezifische Kommunikationsobjekte: 2000 hex – 53FF hex

## 11.2 EMCY-Nachricht: Fehler-Codes Gerätestörung

Error code	Error register	Ereignis (E82)
0 hex: No error	0 hex: No error	30: Inaktiv
1000 hex: Generic error	1 hex: Generic error	48: Lüftüberwachung Bremse, 49: Bremse, 50: Sicherheitsmodul, 71: Firmware oder 80: Ungültige Aktion
2110 hex: Short circuit earth	2 hex: Current	31: Kurz-/Erdschluss
2230 hex: Intern short circuit earth	2 hex: Current	32: Kurz-/Erdschluss intern
2310 hex: Continous overcurrent	2 hex: Current	33: Überstrom
3110 hex: Mains overvoltage	4 hex: Voltage	36: Überspannung
3120 hex: Mains undervoltage	4 hex: Voltage	46: Unterspannung
3130 hex: Phase failure	1 hex: Generic error	83: Ausfall einer/aller Netzphasen
3180 hex: Mains failure	1 hex: Generic error	84: Netzeinbruch bei aktivem Leistungsteil
4210 hex: Temperature	8 hex: Temperature	38: Temperatur Antriebsreglersensor
4280 hex: Temperature device I <sup>2</sup> t	8 hex: Temperature	39: Übertemperatur Antriebsregler i2t oder 59: Übertemperatur Antriebsregler i2t
4310 hex: Temperature drive	8 hex: Temperature	41: Übertemperatur Motorsensor
4380 hex: Temperature drive I <sup>2</sup> t	8 hex: Temperature	45: Übertemperatur Motor i2t
5200 hex: Device hardware	1 hex: Generic error	34: Hardware-Defekt oder 55: Optionsmodul
5440 hex: Contacts	1 hex: Generic error	43: AI1 Drahtbruch
6010 hex: Internal software	1 hex: Generic error	35: Watchdog oder 57: Laufzeitauslastung
6320 hex: Loss of parameters	1 hex: Generic error	40: Ungültige Daten oder 70: Parameterkonsistenz
7110 hex: Brake chopper	1 hex: Generic error	72: Timeout Bremsentest, 73: Timeout Bremsentest Achse 2, 74: Timeout Bremsentest Achse 3 oder 75: Timeout Bremsentest Achse 4
	8 hex: Temperature	42: Übertemperatur Bremswiderstand i2t
7120 hex: Motor	1 hex: Generic error	69: Motoranschluss oder 81: Motorzuordnung
7303 hex: Resolver 1 fault	1 hex: Generic error	37: Motorencoder
7304 hex: Resolver 2 fault	1 hex: Generic error	58: Encodersimulation, 76: Positionsensor, 77: Masterencoder oder 79: Plausibilität Motor- / Positionsensor
7321 hex: Hall sensor failure	1 hex: Generic error	82: Hall-Sensor
7500 hex: Communication	10 hex: Communication	52: Kommunikation
7580 hex: Communication control panel	1 hex: Generic error	88: Steuertafel
8311 hex: Excess torque	1 hex: Generic error	47: Überschreitung max. M/F

Error code	Error register	Ereignis (E82)
8400 hex: Velocity speed control	1 hex: Generic error	56: Overspeed
8500 hex: Position control	1 hex: Generic error	53: Endschalter
8510 hex: Excessive reference position jump	1 hex: Generic error	85: Exzessiver Sollwertsprung
8600 hex: Positioning controller	1 hex: Generic error	51: Virtueller Master Software-Endschalter
8611 hex: Following error	1 hex: Generic error	54: Schleppabstand
8612 hex: Reference limit	1 hex: Generic error	78: Zyklische Positionsbegrenzung
FF00 – FF07 hex: Manufacturer specific error	1 hex: Generic error	60: Applikationsereignis 0 – 67: Applikationsereignis 7
FF09 hex: Manufacturer specific error	1 hex: Generic error	44: Externe Störung 1
FF0A hex: Manufacturer specific error	1 hex: Generic error	68: Externe Störung 2

Tab. 17: EMCY: Fehler-Codes Gerätestörung

## 11.3 Weiterführende Informationen

Die nachfolgend gelisteten Dokumentationen liefern Ihnen weitere relevante Informationen zur 6. STÖBER Antriebsreglergeneration. Den aktuellen Stand der Dokumentationen finden Sie im STÖBER Download-Center unter <http://www.stoeber.de/de/downloads/>, wenn Sie die ID der Dokumentation in die Suche eingeben.

Titel	Dokumentation	Inhalte	ID
Antriebsregler SD6	Handbuch	Systemaufbau, technische Daten, Projektierung, Lagerung, Einbau, Anschluss, Inbetriebnahme, Betrieb, Service, Diagnose	442425
Applikation CiA 402 – SD6	Handbuch	Projektierung, Konfiguration, Parametrierung, Funktionstest, weiterführende Informationen	443076

Zusätzliche Informationen und Quellen, die als Grundlage für diese Dokumentation dienen oder aus denen zitiert wird:

CiA DS 301 V4.02 – CANopen application layer and communication profile

CANopen-Kommunikationsprofil; beschreibt die maßgeblichen Dienste und Protokolle unter CANopen.

CiA DSP 302 V3.0 – CANopen application layer and communication profile

CANopen-Framework für programmierbare Geräte

CiA DS 402 V2.0 – CANopen device profile drives an motion control

CANopen-Geräteprofile; beschreiben das Verhalten zahlreicher Geräteklassen.

CiA DRP 303-1, ISO 11898-2

Empfehlungen für Kabel und Steckverbinder.

EN 50325-4 2002 : *Industrial communications subsystem based on ISO 11898 (CAN) for controller-device interfaces - Part 4: CANopen*. Specification. December 2002.

IEC 61800-7-201:2015 : *Adjustable speed electrical power drive systems - Part 7-201: Generic interface and use of profiles for power drive systems - Profile type 1 specification*. Specification. Version 2.0, November 2015.

IEC 61800-7-301:2015 : *Adjustable speed electrical power drive systems - Part 7-301: Generic interface and use of profiles for power drive systems - Mapping of profile type 1 to network technologies*. Specification. Version 2.0, November 2015.

### Informationen rund um CANopen

Generelle Informationen zu CAN und CANopen erhalten Sie auf der CiA-Website CAN in Automation

<https://www.can-cia.org/>.

## 11.4 Abkürzungen

Abkürzung	Bedeutung
CAN	Controller Area Network
CAN-H	CAN-High
CAN-L	CAN-Low
CAL	CAN Application Layer
CiA	CAN in Automation
COB-ID	Communication Object Identifier
DBT	Distributor
DC	Direct Current (Gleichstrom)
EMCY	Emergency (Notfall)
GND	Ground (Masse)
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IGB	Integrierter Bus
IP	Internet Protocol (dt.: Internetprotokoll)
LSB	Least Significant Bit (kleinstwertiges Bit)
LSW	Least Significant Word (kleinstwertiges Wort)
MSB	Most Significant Bit (höchstwertiges Bit)
MSW	Most Significant Word (höchstwertiges Wort)
NMT	Network Management (Netzwerk-Management)
PDO	Process Data Objects (Prozessdaten-Objekte)
RTR	Remote Transmission Request
RxPDO	Receive-PDO (Empfangs-Prozessdaten)
RxSDO	Receive-SDO (Empfangs-Servicedaten)
SDO	Service Data Objects (Servicedaten-Objekte)
SYNC	Synchronization (Synchronisation)
TxPDO	Transmit-PDO (Sende-Prozessdaten)
TxSDO	Transmit-SDO (Sende-Servicedaten)

## 12 Kontakt

### 12.1 Beratung, Service, Anschrift

Wir helfen Ihnen gerne weiter!

Auf unserer Webseite stellen wir Ihnen zahlreiche Informationen und Dienstleistungen rund um unsere Produkte bereit:

<http://www.stoeber.de/de/service>

Für darüber hinausgehende oder individuelle Informationen, kontaktieren Sie unseren Beratungs- und Support-Service:

<http://www.stoeber.de/de/support>

Sie benötigen unseren System-Support:

Fon +49 7231 582-3060

[systemsupport@stoeber.de](mailto:systemsupport@stoeber.de)

Sie benötigen ein Ersatzgerät:

Fon +49 7231 582-1128

[replace@stoeber.de](mailto:replace@stoeber.de)

So erreichen Sie unsere 24 h Service-Hotline:

Fon +49 7231 582-3000

Unsere Anschrift lautet:

STÖBER Antriebstechnik GmbH + Co. KG

Kieselbronner Straße 12

75177 Pforzheim, Germany

### 12.2 Ihre Meinung ist uns wichtig

Diese Dokumentation erstellen wir nach bestem Wissen mit dem Ziel, Sie beim Auf- und Ausbau Ihres Know-hows rund um unser Produkt nutzbringend und effizient zu unterstützen.

Ihre Anregungen, Meinungen, Wünsche und konstruktive Kritik helfen uns, die Qualität unserer Dokumentation sicherzustellen und weiterzuentwickeln.

Wenn Sie uns aus genannten Gründen kontaktieren möchten, freuen wir uns über eine E-Mail an:

[documentation@stoeber.de](mailto:documentation@stoeber.de)

Vielen Dank für Ihr Interesse.

Ihr STÖBER Redaktionsteam



## 12.3 Weltweite Kundennähe

Wir beraten und unterstützen Sie mit Kompetenz und Leistungsbereitschaft in über 40 Ländern weltweit:

**STOBER AUSTRIA**

www.stoeber.at  
+43 7613 7600-0  
sales@stoeber.at

**STOBER FRANCE**

www.stoeber.fr  
+33 478 98 91 80  
sales@stoeber.fr

**STOBER HUNGARY**

www.stoeber.de  
+36 53 5011140  
info@emtc.hu

**STOBER JAPAN**

www.stober.co.jp  
+81-3-5875-7583  
sales@stober.co.jp

**STOBER TAIWAN**

www.stober.tw  
+886 4 2358 6089  
sales@stober.tw

**STOBER UK**

www.stober.co.uk  
+44 1543 458 858  
sales@stober.co.uk

**STOBER CHINA**

www.stoeber.cn  
+86 512 5320 8850  
sales@stoeber.cn

**STOBER Germany**

www.stoeber.de  
+49 4 7231 582-0  
sales@stoeber.de

**STOBER ITALY**

www.stober.it  
+39 02 93909570  
sales@stober.it

**STOBER SWITZERLAND**

www.stoeber.ch  
+41 56 496 96 50  
sales@stoeber.ch

**STOBER TURKEY**

www.stober.com  
+90 216 510 2290  
sales-turkey@stober.com

**STOBER USA**

www.stober.com  
+1 606 759 5090  
sales@stober.com

# Glossar

## Broadcast-Domain

---

Logischer Verbund von Netzwerkgeräten in einem lokalen Netzwerk, der alle Teilnehmer über Broadcast erreicht.

## COB-ID

---

Jede Nachricht, die über einen CAN-Bus versendet wird, besitzt eine COB-ID, die die Priorität dieser Nachricht bestimmt (niedrige COB-ID = hohe Prio). Die COB-ID setzt sich aus dem Function code und der Node-ID zusammen ( $\text{COB-ID} = \text{FC} + \text{NID}$ ).

## EMCY

---

Kommunikationsobjekte in einem CANopen- oder EtherCAT-Netzwerk, die bei fehlerhaften Zustandsübergängen oder bei geräteinternen Fehlern zugehörige Fehler-Codes und -ursachen übertragen.

## Function Code

---

Kennung zur Unterscheidung der Art einer CAN-Nachricht sowie deren Priorität. Function Code und Node-ID bilden die COB-ID.

## IPv4-Limited-Broadcast

---

Art eines Broadcast in einem Netzwerk mit IPv4 (Internet Protocol Version 4). Als Ziel wird die IP-Adresse 255.255.255.255 angegeben. Der Inhalt des Broadcast wird von einem Router nicht weitergeleitet und ist somit auf das eigene lokale Netzwerk limitiert.

## Node-ID

---

Busadresse eines Geräts (Master, Slave) in einem CANopen-Netzwerk. Node-ID und Function Code bilden die COB-ID.

## Process Data Objects (PDO)

---

Kommunikationsobjekte in einem CANopen- oder EtherCAT-Netzwerk, die Daten wie Soll- und Istwerte, Steuerbefehle oder Statusinformationen ereignis- oder zielorientiert, zyklisch oder auf Anforderung in Echtzeit übertragen. PDO werden über den Prozessdaten-Kanal generell mit hoher Priorität ausgetauscht. Abhängig von der Sicht der jeweiligen Teilnehmer werden Empfangs-PDO (RxPDO) von Sende-PDO (TxPDO) unterschieden.

## SDO

---

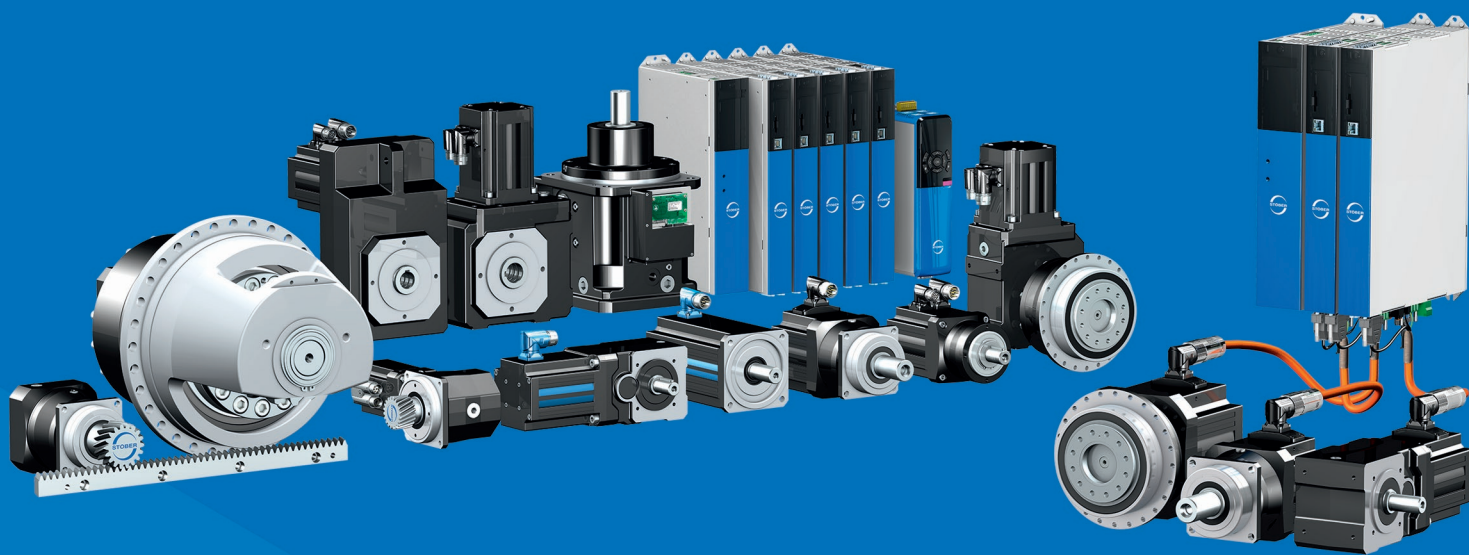
Kommunikationsobjekte in einem CANopen- oder EtherCAT-Netzwerk, die den Zugriff auf das Objektverzeichnis erlauben und eine Gerätekonfiguration ermöglichen. SDO werden azyklisch im laufenden zyklischen CANopen- oder EtherCAT-Betrieb über den Mailbox-Kanal übertragen.

# Abbildungsverzeichnis

Abb. 1	CANopen: Netzwerkaufbau am Beispiel der Baureihe SD6.....	11
Abb. 2	CANopen-Netzwerk: Abschlusswiderstände .....	13
Abb. 3	DS6: Programmoberfläche .....	16
Abb. 4	DriveControlSuite: Navigation über Textlinks und Symbole .....	18
Abb. 5	Leuchtdioden für den CAN-Zustand .....	35
Abb. 6	CAN-Nachricht: Aufbau .....	41
Abb. 7	Network Management State Machine: Zustände und Zustandswechsel.....	53
Abb. 8	Protokoll Guarding (Node Guarding/Life Guarding).....	55
Abb. 9	Protokoll Heartbeat.....	56
Abb. 10	Feldbuskalierung im Überblick .....	60

## Tabellenverzeichnis

Tab. 1	CAN-Bus: Übertragungsgeschwindigkeit und max. Kabellänge .....	14
Tab. 2	Anschlussbeschreibung X200 .....	14
Tab. 3	Parametergruppen .....	19
Tab. 4	Parameter: Datentypen, Parameterarten, mögliche Werte .....	20
Tab. 5	Parametertypen .....	21
Tab. 6	Bedeutung der grünen LED (Run).....	35
Tab. 7	Bedeutung der roten LED (Error) .....	35
Tab. 8	Ereignis 52 – Ursachen und Maßnahmen .....	37
Tab. 9	Objektverzeichnis – Struktur .....	40
Tab. 10	Transmission Type.....	47
Tab. 11	Zykluszeiten.....	57
Tab. 12	Aktion auswählen und ausführen .....	59
Tab. 13	Feldbuskalierung bei Ganzzahl: Parameter zur Definition der Dezimalstellen .....	60
Tab. 14	Kommunikationsobjekte CiA 301: 1000 hex – 1FFFF hex.....	61
Tab. 15	Index und Subindex bei herstellerspezifischen Parametern .....	66
Tab. 16	Herstellerspezifische Kommunikationsobjekte: 2000 hex – 53FF hex .....	67
Tab. 17	EMCY: Fehler-Codes Gerätestörung.....	68



4 4 2 6 3 6 - 0 2

10/2023

STÖBER Antriebstechnik GmbH + Co. KG  
Kieselbronner Str. 12  
75177 Pforzheim  
Germany  
Tel. +49 7231 582-0  
mail@stoeber.de  
www.stober.com

24 h Service Hotline  
+49 7231 582-3000

[www.stober.com](http://www.stober.com)