Baumer

Sensor Solutions • Motion Control • Vision Technologies

Handbuch

Stellantrieb *FlexiDrive* MSIA mit CANopen

Software Revision (Firmware Version) 2.05.xx

Baumer Electric AG
Hummelstrasse 17
8501 Frauenfeld · Switzerland
Telefon +41 (0)52 728 11 22
Telefax +41 (0)52 728 11 44
E-Mail: sales.ch@baumerelectric.com
www.baumerelectric.com

09.01.2008

Irrtum sowie Änderungen in
Technik und Design vorbehalten

HB MSIA CANopen D V2.05.xx rev06.doc

Inhalt

Seite

1	**Einleitung**	**4**
1.1	Produktzuordnung	4
1.2	Funktionsübersicht	5
1.3	Sicherheits- und Betriebshinweise	6
2	**Kommunikation über CANopen**	**7**
2.1	Allgemeine Informationen	7
2.2	Einzelheiten über CANopen	8
2.2.1	Kommunikationsprofil	8
2.2.2	Meldungsaufbau	8
2.2.3	Servicedaten-Kommunikation (SDO)	9
2.2.4	Prozessdaten-Kommunikation (PDO)	11
2.2.5	Emergency-Dienst	13
2.2.6	Netzwerkmanagement und NMT-Zustandsdiagramm	13
2.2.7	Node Guarding und Life Guarding	15
2.2.8	Heartbeat Protokoll	16
2.2.9	Layer Setting Services (LSS)	17
2.3	Zustandsdiagramm für Positioning / Homing	20
2.4	Betrieb des Antriebs	24
2.4.1	Antriebsstatus und aktuelle Position auslesen	24
2.4.2	Sollposition anfahren über SDO (mit Fahrauftrag)	24
2.4.3	Fahrauftrag stoppen	24
2.4.4	Ziel anfahren über R_PDO (mit Fahrauftrag)	25
2.5	Befehlssynchronisation zwischen Antrieb und Steuerung	26
3	**Objektverzeichnis**	**29**
4	**Inbetriebnahme**	**46**
4.1	Node-ID und Baudrate einstellen	46
4.2	Objekte schreiben und lesen	48
4.3	Antriebsparameter konfigurieren	50
4.3.1	Position, Geschwindigkeit und Getriebe	50
4.3.2	Drehrichtung (Polarity)	52
4.3.3	Strom und Geschwindigkeit	52
4.3.4	Positioning Timeout	52
4.3.5	Spielausgleich	52
4.3.6	Einstellungen speichern	53
4.4	Antriebsposition referenzieren (Homing)	54
4.4.1	Position oder Referenzposition direkt setzen	55
4.4.2	Referenzieren auf Referenzschalter	56
4.4.3	Referenzieren auf Anschlag	57
4.4.4	Referenzieren mit Einmessmethode	58
4.4.5	Zurücksetzen der Statusbits HomingOK und CalibrationOK	58
4.5	Positionen der Software-Endschalter setzen	59
4.6	Fahraufträge ausführen	59
5	**Fehlerdiagnose**	**60**
5.1	Feldbus-Kommunikation	60
5.2	Fehlerdiagnose am Antrieb	60
5.3	EMV-gerechte Verdrahtung	62
6	**Technische Daten**	**65**

6.1 Elektrische und mechanische Daten 65
6.2 Anschlussbelegung 66
6.3 Blockschaltbild und Schalteingänge 66
6.4 Rückspeiseschutz beim Abbremsen des Motors 67
6.5 Motorstromreduktion bei erhöhter Umgebungstemperatur 68
6.6 Abmessungen und Einbaumasse 70
6.7 Befehlssequenzen als Beispiel 71

1 Einleitung

1.1 Produktzuordnung

Tabelle 1: Produktzuordnung.

Produktfamilie	Produkt	Produktcode	EDS Datei	Schnittstelle
FlexiDrive	MSIA 42C2 xxxx - xxxxxx xxx	100h	MSIA_xxC.eds	CANopen
FlexiDrive	MSIA 53C2 xxxx - xxxxxx xxx	100h	MSIA_xxC.eds	CANopen
FlexiDrive	MSIA 68C2 xxxx - xxxxxx xxx	100h	MSIA_xxC.eds	CANopen
FlexiDrive	MSIA 83C2 xxxx - xxxxxx xxx	100h	MSIA_xxC.eds	CANopen

1.2 Funktionsübersicht

Intelligenter Positioniermotor als dezentraler Teilnehmer am CAN-Bus

Getriebebaukasten mit Planeten-, Stirnrad oder Schneckengetriebe

Absolut-Multiturn Positionserfassung (14 Bit ST, 18 Bit MT)

Motor- / Elektronikstromversorgung
- über gemeinsames Kabel
- über getrennte Kabel (siehe Anschlussbelegung)

4 Digitaleingänge für Jogging, Hardware-Endschalter oder Referenzierung
- ansteuerbar über Standardsensor
- verschiedene Funktionen parametrierbar (Objekt 2111h-0Dh)

1 Digitalausgang als zur Ansteuerung externer Geräte (optional bis 4 Digitalausgänge, Objekt 2111h-14h)

Betriebsarten (Objekt 6060h):
- Positionieren
- Referenzieren
- drehzahlgeregelter Betrieb

Sonstige Funktionen
- Software-Endschalter zur Begrenzung des Fahrwegs (Objekt 607Dh)
- Hardware-Endschalter (Objekt 2111h-0Dh)
- Sollposition setzen (Objekt 607Ah)
- Halteregler
- Stillstandsüberwachung (Objekt 6041h)
- Spielausgleich gegen Getriebe- oder Spindelspiel (Objekt 2111h-01h)
- Drehmoment- / Strombegrenzung (Objekt 6073h)
- Linear- oder Rundachse (Objekt 608Ah)
- Drehrichtung umkehrbar (Objekt 607Eh)
- Stand-Alone-Betrieb / Tippbetrieb ohne CAN-Bus und Parametrierung (Objekt 2111h-13h)
- Komfortable Referenzierungsfunktionen (Objekt 6098h)
    - Istwert setzen (Objekt 2111h-11h)
    - Fahren auf Anschlag
    - Fliegendes Referenzieren
- Status- und Fehlerdiagnose über CAN-Bus (Objekt 1003h und 603Fh)
- Temperaturüberwachung mit Warnmeldung und Abschaltung im Fehlerfall (Objekt 2114h)
- Fahrdatensätze programmierbar

Kommunikation über CANopen
- SDO-Kommunikation (Parameter lesen / schreiben)
- PDO-Kommunikation (Ansteuerung, Senden von Status und Position)

Einfache Inbetriebnahme
- Anpassung weniger Parameter
- Feldbusanschluss durchschleifbar
- Vorkonfektionierte Kabel für Bus und Spannungsversorgung als Zubehör erhältlich
- Parametrier- und Inbetriebnahmetool CANmaster (siehe CD-ROM Art. Nr. 147362)

Anschlussdeckel mit integrierten Einstellschaltern für Knotennummer und Baudrate

Nichtflüchtiger Speicher (EEPROM) für Default- und Anwenderdaten (Objekt 2300h)

1.3 Sicherheits- und Betriebshinweise

Zusätzliche Informationen
- Das Handbuch ist eine Ergänzung zu bereits vorhandenen Dokumentationen (Kataloge, Produktinformationen und Montageanleitungen).
- Die Anleitung muss unbedingt vor Inbetriebnahme gelesen werden.

Bestimmungsgemässer Gebrauch
- Der Antrieb ist ein Präzisionsgerät. Er dient ausschließlich zur automatischen Verstellung von mechanischen Anschlägen und Spindeleinrichtungen.

Inbetriebnahme
- Einbau und Montage des Antriebs darf ausschließlich durch eine Fachkraft erfolgen.
- Betriebsanleitung des Maschinenherstellers beachten.

Sicherheitshinweise
- Vor Inbetriebnahme der Anlage alle elektrischen Verbindungen überprüfen.
- Wenn Montage, elektrischer Anschluss oder sonstige Arbeiten am Antrieb und an der Anlage nicht fachgerecht ausgeführt werden, kann es zu Fehlfunktion oder Ausfall des Antriebs führen.
- Eine Gefährdung von Personen, eine Beschädigung der Anlage und eine Beschädigung von Betriebseinrichtungen durch den Ausfall oder Fehlfunktion des Antriebs muss durch geeignete Sicherheitsmassnahmen ausgeschlossen werden.
- Antrieb darf nicht außerhalb der Grenzwerte betrieben werden (siehe weitere Dokumentationen).

Bei Nichtbeachtung kann es zu Fehlfunktionen, Sach- und Personenschäden kommen!

Transport und Lagerung
- Transport und Lagerung ausschließlich in Originalverpackung.
- Antrieb nicht fallen lassen oder größeren Erschütterungen aussetzen.

Montage
- Schläge oder Schocks auf Gehäuse und Welle vermeiden.
- Gehäuse nicht verspannen.
- Antrieb nicht öffnen oder mechanisch verändern.

Welle, Kugellager oder elektronische Teile können beschädigt werden. Die sichere Funktion ist dann nicht mehr gewährleistet.

Elektrische Inbetriebnahme
- Antrieb nicht elektrisch verändern.
- Keine Verdrahtungsarbeiten unter Spannung vornehmen.
- Der elektrische Anschluss darf unter Spannung nicht aufgesteckt oder abgenommen werden.
- Die gesamte Anlage EMV gerecht installieren. Einbauumgebung und Verkabelung beeinflussen die EMV des Antriebs. Antrieb und Zuleitungen räumlich getrennt oder in großem Abstand zu Leitungen mit hohem Störpegel (Frequenzumrichter, Schütze usw.) verlegen.
- Bei Verbrauchern mit hohen Störpegeln separate Spannungsversorgung für den Antrieb bereitstellen.
- Antriebsgehäuse und die Anschlusskabel vollständig schirmen.
- Antrieb an Schutzerde (PE) anschließen. Geschirmte Kabel verwenden. Schirmgeflecht muss mit dem Stecker verbunden sein. Anzustreben ist ein beidseitiger Anschluss an Schutzerde (PE), Gehäuse über den mechanischen Anbau, Kabelschirm über die nachfolgenden angeschlossenen Geräte. Bei Problemen mit Erdschleifen mindestens eine einseitige Erdung.

Bei Nichtbeachtung kann es zu Fehlfunktionen, Sach- und Personenschäden kommen!

2 Kommunikation über CANopen

2.1 Allgemeine Informationen

Der CAN-Bus (CAN: Controller Area Network) wurde ursprünglich von Bosch und Intel für die schnelle und kostengünstige Datenübertragung im Kraftfahrzeug entwickelt. Heute ist der CAN-Bus einer der am weitesten verbreiteten Feldbusse in der industriellen Automatisierung. Über ihn kommunizieren Geräte, Aktoren und Sensoren verschiedener Hersteller miteinander. Normen für den CAN-Bus werden durch die Nutzer-Vereinigung „CAN in Automation“ (CiA) festgelegt.

CAN-Bus-Eigenschaften

- Datenrate von 1 MBit/s bei einer Netzausdehnung bis 25 m
- Beidseitig abgeschlossenes Netzwerk
- Busmedium ist ein Twisted-Pair-Kabel
- Echtzeitfähigkeit: Definierte maximale Wartezeit für Nachrichten hoher Priorität.
- Bis zu 127 Teilnehmer an einem Bus (physikalisch treiberbedingt maximal 32 Teilnehmer)
- Sicherstellung netzweiter Datenkonsistenz. Gestörte Nachrichten werden für alle Netzknoten als fehlerhaft bekannt gemacht.
- Nachrichtenorientierte Kommunikation
- Jede Nachricht wird mit einer Nachrichtenkennung (Identifier) gekennzeichnet. Alle Netzknoten prüfen anhand des Identifiers, ob die Nachricht für sie relevant ist.
- Broadcasting: Alle Netzknoten erhalten gleichzeitig jede Nachricht. Daher ist eine Synchronisation möglich.
- Multi-Master-Fähigkeit: Jeder Teilnehmer im Feldbus kann selbständig Daten senden und empfangen, ohne dabei auf eine Priorität der Master angewiesen zu sein. Jeder kann seine Nachricht beginnen, wenn der Bus nicht belegt ist. Bei einem gleichzeitigen Senden von Nachrichten setzt sich der Teilnehmer mit der höchsten Priorität durch.
- Priorisierung von Nachrichten: Der Identifier (COB-ID) setzt die Priorität der Nachricht fest. Dadurch können wichtige Nachrichten schnell über den Bus übertragen werden.
- Restfehlerwahrscheinlichkeit: Sicherungsverfahren im Netzwerk reduzieren die Wahrscheinlichkeit einer unentdeckten, fehlerhaften Datenübertragung auf unter 10^{-11}. Praktisch kann von einer 100% sicheren Übertragung ausgegangen werden.
- Funktionsüberwachung: Lokalisation fehlerhafter oder ausgefallener Stationen. Das CAN-Protokoll beinhaltet eine Funktionsüberwachung von Netzknoten. Netzknoten, die fehlerhaft sind, werden in ihrer Funktion eingeschränkt oder ganz vom Netzwerk abgekoppelt.
- Datenübertragung mit kurzer Fehlererholzeit: Durch mehrere Fehlererkennungsmechanismen werden verfälschte Nachrichten mit großer Wahrscheinlichkeit erkannt. Wird ein Fehler erkannt, so wird die Nachrichtensendung automatisch wiederholt.

Im CAN-Bus sind mehrere Netzwerkteilnehmer über ein Buskabel miteinander verbunden. Jeder Netzwerkteilnehmer kann Nachrichten senden und empfangen. Die Daten zwischen den Netzwerk-Teilnehmern werden seriell übertragen.

Beispiele für CAN-Bus-Geräte als Netzwerkteilnehmer sind:

- Automatisierungsgeräte wie SPS oder PC
- Ein- / Ausgangsmodule
- Antriebssteuerungen
- Analysegeräte, z. B. ein CAN-Monitor
- Bedien- und Eingabegeräte
- Sensoren und Aktoren

2.2 Einzelheiten über CANopen

Unter technischer Leitung des Steinbeis Transferzentrum für Automatisierung wurde auf der Basis der Schicht 7 Spezifikation CAL (CAN Application Layer) das CANopen-Profil entwickelt. Im Vergleich zu CAL sind in CANopen nur die für diesen Einsatz geeignete Funktionen enthalten. CANopen stellt somit eine für die Anwendung optimierte Teilmenge von CAL dar und ermöglicht dadurch einen vereinfachten Systemaufbau und den Einsatz vereinfachter Geräte. CANopen ist optimiert für den schnellen Datenaustausch in Echtzeitsystemen. Die CAN in Automation (CiA) ist zuständig für die geltenden Normen der entsprechenden Profile.

CANopen ermöglicht:

- einfachen Zugriff auf alle Geräte- und Kommunikationsparameter
- Synchronisation von mehreren Geräten
- automatische Konfiguration des Netzwerkes
- zyklischen und ereignisgesteuerten Prozessdatenverkehr

Alle CANopen Geräte- und Kommunikationsparameter sind in einem Objektverzeichnis gegliedert. Ein CANopen Objekt umfasst Objekt-Nummer mit Subindices, Objekt-Format sowie Struktur und Inhalt der Parameter. Gemäss CiA ist dieses Objektverzeichnis in drei verschiedene Teile unterteilt:

- Kommunikationsprofil
- Geräteprofil
- Herstellerspezifischen Teil

Für Einzelheiten zu den Objekten dieses Antriebs, siehe Kapitel Objektverzeichnis.

2.2.1 Kommunikationsprofil

Die CANopen Kommunikation zwischen Netzwerkteilnehmern und Master erfolgt über Objekte. Diese Objekte werden mit einem 16 Bit-Index adressiert. Im CANopen Kommunikationsprofil DS 301 sind die verschiedenen Kommunikationsobjekte (COB) standardisiert:

- Prozessdaten-Objekte PDO (Process Data Object) zur Echtzeitübertragung von Prozessdaten
- Servicedaten-Objekte SDO (Service Data Object) für den Schreib- und Lesezugriff auf das Objektverzeichnis
- Objekte zur Synchronisation und Fehleranzeige von CAN-Teilnehmern:
  SYNC-Objekt (Synchronisation Object) zur Synchronisation von Netzwerkteilnehmern
  EMCY-Objekt (Emergency Object) zur Fehleranzeige eines Gerätes oder seiner Peripherie
- Network Management (Network Management NMT, Lifeguarding) zur Initialisierung und Netzwerksteuerung
- Layer Setting Services (LSS) zur Konfiguration über Seriennummer oder Revisionsnummer

2.2.2 Meldungsaufbau

Der erste Teil einer Meldung ist die so genannte COB-ID (Identifier).

Aufbau der 11-Bit COB-ID:

Funktionscode				Node-ID						
4 Bit Function code				7 Bit Node-ID						

Der Funktionscode gibt Aufschluss über die Art der Meldung und die Priorität. Je niedriger der Funktionscode, desto höher die Priorität der Meldung.

Broadcast-Meldungen:

Funktionscode	COB-ID
NMT	0
SYNC	80h

Peer-to-Peer-Meldungen:

Funktionscode	COB-ID
EMERGENCY	80h + Node-ID
PDO1 (tx)[1]	180h + Node-ID
PDO2 (tx)[1]	280h + Node-ID
SDO (tx)[1]	580h + Node-ID
SDO (rx)[1]	600h + Node-ID
Nodeguard, Heartbeat	700h + Node-ID
LSS (tx) [1]	7E4h
LSS (rx) [1]	7E5h

[1](tx) und (rx) aus der Sicht des Antriebs.
Für Einstellung der Node-ID (Knotennummer des Busteilnehmers) siehe Kapitel Inbetriebnahme.

Ein CAN-Telegramm besteht aus der COB-ID und bis zu 8 Datenbyte. Die genauen Telegramme werden im folgenden ausführlich erläutert.

COB-ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Xxx	x	xx	xx	xx	xx	xx	xx	xx	xx

2.2.3 Servicedaten-Kommunikation (SDO)

Die Servicedaten-Objekte (SDO) entsprechen dem Protokoll DSP-402. Über Index und Subindex kann auf ein Objekt zugegriffen werden. Die Daten können angefordert oder ins Objekt geschrieben werden.

Ein SDO-Telegramm setzt sich zusammen aus einem Kommando-Byte, zwei Objekt-Bytes, einem Subindex-Byte sowie bis zu vier Datenbytes und ist folgendermaßen aufgebaut:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3

Die COB-ID eines SDO setzt sich folgendermaßen zusammen:

- Kommunikation Master zu Antrieb: 600h + Node-ID
- Kommunikation Antrieb zu Master: 580h + Node-ID

Tabelle 2: Beschreibung von SDO-Kommandos. Der DLC (Data Length Code) bezeichnet die Länge des Telegramms. Das Kommando-Byte legt fest, ob Daten gelesen oder gesetzt werden und um wie viele Datenbyte es sich handelt.

SDO Kommando	Beschreibung	Datenlänge	Beschreibung
22h	Download Request	Max. 4 Byte	Parameter an Antrieb senden
23h	Download Request	4 Byte	
2Bh	Download Request	2 Byte	
2Fh	Download Request	1 Byte	
…			
60h	Download Response	-	Bestätigung der Übernahme an Master
40h	Upload Request	-	Parameter vom Antrieb anfordern
…			
42h	Upload Response	Max. 4 Byte	Parameter an Master mit max. 4 Byte
43h	Upload Response	4 Byte	
4Bh	Upload Response	2 Byte	
4Fh	Upload Response	1 Byte	
…			
80h	Abort Message	-	Antrieb meldet Abort Code an Master

Eine SDO Abort Message zeigt einen Fehler in der CAN-Kommunikation an. Das zugehörige SDO Kommando-Byte ist 80h. Objekt und Subindex sind die des gewünschten Objektes. Byte 4...7 ergeben zusammen den SDO Abort Code (Byte 7 = MSB).

COB-ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
580h+Node-ID	8	80h	Objekt L	Objekt H	Subindex	ErrByte 0	ErrByte 1	ErrByte 2	ErrByte 3

Für Beschreibung der SDO Abort Codes, siehe Kapitel Fehlerdiagnose.

Beispiele für eine SDO-Kommunikation

Anfrage eines Wertes vom Master beim Antrieb. Eine häufige verwendete Anfrage ist beispielsweise die Statusanfrage (siehe Objekt 6041h):

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
600h+Node-ID	8	40h	41h	60h	0	x	x	x	x

Antwort des Antriebs auf die Anfrage eines Wertes. Der Status ist 4 Byte lang (siehe Objekt 6041h):

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	43h	41h	60h	0	a	b	c	d

Schreiben eines Werts vom Master zum Antrieb. Mit dem Kontrollwort wird der Antrieb gesteuert (siehe Objekt 6040h):

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
600h+Node-ID	8	22h	40h	60h	0	a	b	x	x

Antwort des Antriebs auf das Schreiben des Werts:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	60h	40h	60h	0	0	0	0	0

2.2.4 Prozessdaten-Kommunikation (PDO)

Prozessdatenobjekte dienen dem Echtzeit-Datenaustausch für Prozessdaten wie beispielsweise die aktuelle Position des Antriebs oder den Betriebsstatus. PDO können synchron oder zyklisch (asynchron) gesendet werden. Der Antrieb unterstützt PDO1 (Status) und PDO2 (Position).

Synchrone Kommunikation

Um die Prozessdaten synchron zu senden, muss im Objekt 1800-2h bzw.1801-2h ein Wert zwischen 1 und F0h eingeschrieben werden. Mit Wert = 3 wird das PDO auf jedes dritte SYNC-Telegramm gesendet. Im Objekt 2800h bzw. 2801h ist die maximale Anzahl definiert. Ist dort zum Beispiel eine 5 eingeschrieben, wird das PDO nach wie vor auf jedes dritte SYNC-Telegramm geschrieben, insgesamt aber nur 5 mal. Dem entsprechend folgt auf das 15. SYNC-Telegramm das letzte PDO. Durch einen NMT-Reset wird dieser Zähler wieder zurückgesetzt. Im synchronen Betrieb werden die PDO vom Master über das SYNC-Telegramm angefordert:

Byte 0	Byte 1
COB-ID = 80	0

Zyklische (asynchrone) Kommunikation

Sollen PDO zyklisch gesendet werden, muss ins Objekt 1800-2h bzw. 1801-2h der Wert FEh geschrieben werden. Im Objekt 1800-5h bzw. 1801-5h steht die Zykluszeit in Millisekunden. Die kleinstmögliche Zykluszeit beträgt 1ms. Wird der Wert 0ms gespeichert, werden keine PDO gesendet. Die Funktion ist ausgeschaltet. Im weiteren kann im Objekt 2800h bzw. 2801h bestimmt werden, ob das Senden des PDO nur auf Änderung erfolgen soll: Beträgt der Wert 0, läuft das zyklische Senden wie oben beschrieben. Beträgt der Wert 1, wird zyklisch geprüft ob eine Änderung des Wertes vorliegt. Wenn keine Änderung vorliegt, wird nicht gesendet. Beträgt der Wert 4, wird bei Änderung das PDO viermal gesendet im vordefinierten Zyklus.

Tabelle 3: Beispiele für Einstellungen der PDO Kommunikation.

1800h		2800h	Beschreibung
Sub2	Sub5		
FEh	10 ms	0h	Zyklisches Senden alle 10ms
FEh	10 ms	2h	Falls Änderung vorliegt, wird PDO zwei mal gesendet im Abstand von 10 ms
FEh	0 ms	0h	PDO senden ausgeschaltet
FEh	0 ms	2h	PDO senden ausgeschaltet
3h	xxx	0h	Bei jedem dritten SYNC-Telegramm senden
3h	xxx	2Bh	Auf jedes dritte SYNC-Telegramm, aber insgesamt nur 43 mal (=2Bh).

PDO1 (Status)

Telegrammaufbau:

COB-ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
180h+Node-ID	8	xx	xx	xx	-	xx	xx	xx	xx

COB-ID : 180h + Node-ID

Länge : 2 (bzw. 3 oder 8) Datenbyte (einstellbar im Objekt 2110h Version control)

Tabelle 4: Statuswort in Byte 0...1 des PDO1-Telegramms (Objekt 6041h).

Bit	Beschreibung
0	Ready to switch on
1	SwitchOn enabled
2	Operation enabled
3	Error active
4	Voltage enabled
5	QuickStop active
6	SwitchOn disabled

7	Warning active
8	CalibrationOK
9	Not used (permanently 1)
10	Target position reached
11	Internal software limit switch active
12	Drive moving
13	EEPROM access
14	HomingOK
15	External hardware limit switch active

Tabelle 5: Statuswort-Zusatz in Byte 2 des PDO1-Telegramms für Anzeige des internen Antriebszustands. Statuswort-Zusatz ist standardmässig aktiviert, die Länge des PDO1-Telegramms ist dann 3 Byte. Deaktivieren von Statuswort-Zusatz erfolgt in Versionskontrolle Objekt 2110h (Bit 1).

Bit	Bit	Beschreibung
SELF_TEST	00	Selbsttest nach Initialisierung
READY	01	Nach erfolgreichem Selbsttest -> Ready
POSITIONING_READY	02	SwitchOn, EnableVoltage, QuickStop, EnableOperation und Positioning -> PositioningReady
HOMING_READY	03	SwitchOn, EnableVoltage, QuickStop, EnableOperation und Homing -> HomingReady
POSITIONING_MOVING	04	Einschalten des Antriebs (Bit4 des Kontrollworts) -> PositioningMove
POSITIONING_FIRSTTARGET	05	Bei indirektem Anfahren der Zielposition, um Spindelspiel aufzuheben
POSITIONING_ENDTARGET	06	Sollposition setzen
HOMING_MOVING1	07	
HOMING_MOVING2	08	
HOMING_SET	09	
HOMING_STEP1	10 (0Ah)	Manuell fahren
HOMING_STEP2	11 (0Bh)	Manuell fahren
HOMING_STEP3	12 (0Ch)	Manuell fahren
HOMING_STEP4	13 (0Dh)	Manuell fahren
HOMING_STEP5	14 (0Eh)	Manuell fahren
QUICK_STOP_ACTIVE_1	15 (0Fh)	Motor stoppen; Bit [0:3] des Kontrollworts bewirkt QuickStop, → Sollposition verwerfen
QUICK_STOP_ACTIVE_2	16 (10h)	Position lesen
QUICK_STOP_ACTIVE_3	17 (11h)	Sollposition = aktuelle Position; Sollposition verwerfen
HALT_ACTIVE	18 (12h)	Bit [8] des Kontrollworts bewirkt Zwischenhalt, Sollposition wird beibehalten
ERROR_DIAGNOSTIC	19 (13h)	Aus jedem Zustand wird im Fehlerfall auf ErrorDiagnostic gewechselt
COMU	20 (14h)	Eigener Zustand für interne Kommunikation
TIPP_BETRIEB	21 (15h)	Antriebssteuerung über Hardware-Eingänge
GET_DSP_STATUS	22 (16h)	Auslesen des DSP Status
HOMING_REFERENCE 1	23 (17h)	Starten der Referenzfahrt
HOMING_REFERENCE 2	24 (18h)	Warten, bis Referenzfahrt beendet
VELOCITY_READY	27 (1Bh)	SwitchOn, EnableVoltage, QuickStop, EnableOperation und Positioning -> VelocityReady

Tabelle 6: Statuswort-Zusatz in Byte 4...7 des PDO1-Telegramms für Anzeige der digitalen Hardware-Schalteingänge (Objekt 60FDh). Aktivieren erfolgt in Versionskontrolle Objekt 2110h (Bit 12). Die Gesamtlänge des PDO1 Telegramms beträgt dann 8 Byte. Byte 3 enthält nie Informationen.

Bit	Beschreibung
0..15	Nicht definiert
16	Monitorbit für Digital input 1
17	Monitorbit für Digital input 2
18	Monitorbit für Digital input 3
19	Monitorbit für Digital input 4
20...31	Nicht definiert

PDO2 (Position)

Telegrammaufbau:

COB-ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3
280h+Node-ID	3	Xx	Xx	Xx	Xx

COB-ID : 280h + Node-ID
Länge : 4 Datenbyte
Byte 0.. 3 : Aktuelle Position in benutzerdefinierter Einheit (z.B. Mikrometer, siehe Objekt 6064h)

2.2.5 Emergency-Dienst

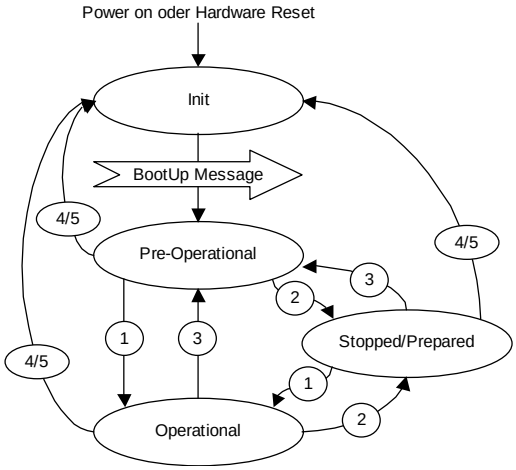
Interne Fehler im Antrieb oder Busfehler lösen eine Emergency-Meldung aus (EMCY-Meldung):

COB-ID	DLC	Byte0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
80h+Node-ID	8	Error Code		Error Register 1001h	Error Code Zusatz Information		Not used		Not used

Für Beschreibung der Error Register und Error Code Definitionen, siehe Kapitel Fehlerdiagnose.

2.2.6 Netzwerkmanagement und NMT-Zustandsdiagramm

Nachdem der Antrieb initialisiert wurde, befindet er sich im Pre-Operational Mode (siehe Figur 1). In diesem Zustand können SDO gelesen und geschrieben werden. Um die PDO-Kommunikation zu starten, muss das NMT-Kommando Start Remote Node gesendet werden. Dann befindet sich der Antrieb im Operational Mode. Nun werden die gewünschten PDO gesendet. Zusätzlich können SDO gelesen und geschrieben werden.
Nachdem der Antrieb mit dem NMT-Kommando Stop Remote Node gestoppt wird, befindet sich der Antrieb im Stopped/Prepared Mode. Es können keine SDO- oder PDO-Parameter mehr gelesen oder gesetzt werden. In diesem Zustand ist nur noch NMT-Kommunikation, LSS, Heartbeat oder Node Guarding möglich. Durch das NMT-Kommando Reset Remote Node wird der Antrieb erneut initialisiert und befindet sich danach wieder im Pre-Operational Mode.



Figur 1: NMT-Zustandsdiagramm.

Mit den Diensten des Netzwerkmanagement (NMT) können die Busteilnehmer initialisiert, gestartet und gestoppt werden. Zusätzlich existieren NMT-Dienste zur Verbindungsüberwachung.

Beschreibung der NMT-Kommandos

Alle NMT-Kommandos werden als unbestätigte Objekte übertragen und sind folgendermaßen aufgebaut:

Byte 0	Byte 1	Byte 2
COB-ID = 0	Kommando-Byte	Node-ID

Die COB-ID für NMT-Kommandos ist immer Null. NMT-Kommandos haben damit maximale Priorität. Die Node-ID des gewünschten Teilnehmers wird in Byte 2 des NMT-Kommandos übertragen. Mit Node-ID = 0 werden alle Teilnehmer angesprochen (Broadcast Meldung).

Tabelle 7: NMT-Kommando-Byte.

Kommando-Byte	Beschreibung	Sprung im NMT Zustandsdiagramm (siehe Figur 1)
01h	Start Remote Node	1
02h	Stop Remote Node	2
80h	Enter Pre-Operational Mode	3
81h, 82h	Reset Remote Node	4, 5

NMT-Zustandswechsel

Init

Nach dem Initialisieren meldet sich der Antrieb mit einer BootUp Meldung auf dem CAN-Bus. Danach wechselt der Antrieb automatisch in den Pre-Operational Mode.

Die COB-ID der BootUp Meldung setzt sich zusammen aus 700h und der Node-ID:

COB-ID	Byte 0
700h+Node-ID	00h

Start Remote Node (1)

Mit dem Startbefehl wird der Antrieb in den Operational Mode gebracht.

COB-ID	Kommando Byte	Node-ID
0	1h	0..127

Stop Remote Node (2)

Mit dem Stopbefehl wird der Antrieb in den Stopped/Prepared Mode gebracht.

COB-ID	Kommando Byte	Node-ID
0	2h	0..127

Enter Pre-Operational Mode (3)

Wechsle in den Pre-Operational Mode.

COB-ID	Kommando Byte	Node-ID
0	80h	0..127

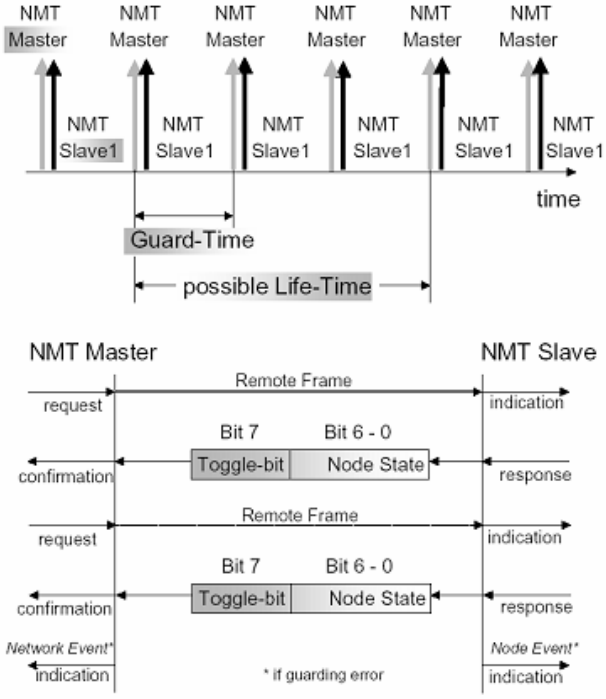
Reset Remote Node (4) oder Reset Kommunikation (5)

Mit dem Resetbefehl wird der Antrieb neu initialisiert.

COB-ID	Kommando Byte	Node-ID	Beschreibung
0	81h	0..127	Reset kompletter Antrieb
0	82h	0..127	Reset CAN Kommunikation

2.2.7 Node Guarding und Life Guarding

Das Nodeguarding Protokoll dient zur Überwachung von NMT-Slaves durch den NMT-Master. Node Guarding wird eingeschaltet durch Setzen von Bit5 im Objekt Versionskontrolle (Objekt 2110h, Version control). Der NMT-Slave erscheint dann in einer Datenbank des NMT-Masters mit den NMT-Zuständen der NMT-Slaves. Mit dem Nodeguarding Protokoll wird überprüft, ob alle NMT-Slaves korrekt arbeiten oder ob sich beispielsweise ein NMT-Slave vom Bus zurückgezogen hat. Gleichzeitig kann jeder NMT-Slave überwachen, ob der NMT-Master noch aktiv ist.



Figur 2: Node Guarding und Life Guarding.

Der NMT-Master startet den Überwachungsdienst durch zyklisches Senden von Remote Frames an den NMT-Slave. Das Zeitintervall dazwischen ist die so genannten Guard Time (siehe Figur 2 und Objekt 100Ch). Sie kann unterschiedlich sein für jeden NMT-Slave.

Die Antwort des NMT-Slaves beinhaltet seinen jeweiligen NMT-Zustand. Auf diese Weise kann der NMT-Master überprüfen, ob sich der NMT-Slave im richtigen NMT-Zustand befindet und gegebenenfalls auf Fehler reagieren.
Die so genannte Knoten-Lebensdauer (Node life time) eines NMT-Slaves ist gegeben durch die Guard Time multipliziert mit dem Lifetime factor (Objekt 100Dh). Sie kann ebenfalls unterschiedlich sein für jeden NMT-Slave. Erhält ein NMT-Slave nicht innerhalb der Knoten-Lebensdauer einen Remote Frame des NMT-Masters, so wird ein so wird ein Life Guarding Event ausgelöst. Das Verhalten im Fehlerfall wird definiert im Objekt Error behavior - Communication error (Objekt 1029h, Subindex 01h).
Gemäss CAN-Nutzerorganisation sollte zur Reduktion des Buslast anstatt Node Guarding das aktuellere Überwachungsprotokoll Heartbeat verwendet werden.

Beispiel eines Node Guarding Protokolls:

COB-ID	Data / Remote	Byte 0
700h+Node-ID	r	00h (0d)
700h+Node-ID	d	FFh (255d)
700h+Node-ID	r	00h (0d)
700h+Node-ID	d	7Fh (127d)

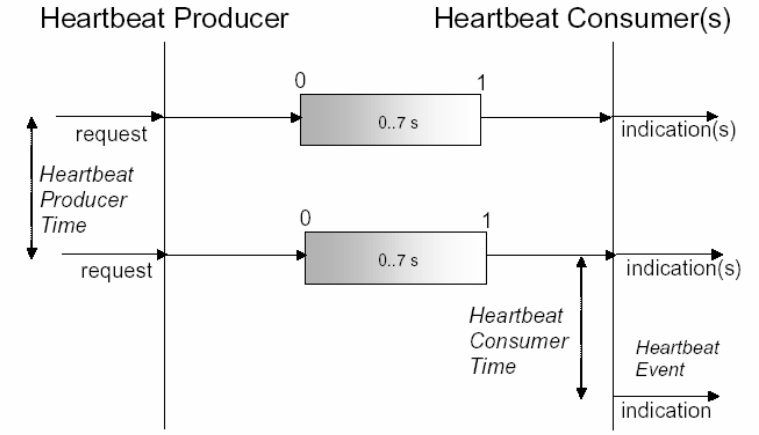
Mögliche Zustände der NMT-Teilnehmer:

0: BootUp-Event
4: Stopped/Prepared
5: Operational
127: Pre-Operational

Die unteren 7 Bits ergeben in obigem Beispiel 7Fh, d.h. der Antrieb befindet sich im Pre-Operational Mode.

2.2.8 Heartbeat Protokoll

Für alle neuen Applikationen ist es dringend empfehlenswert, das modernere Überwachungsprotokoll Heartbeat zu verwenden. Heartbeat wird eingeschaltet im Objekt Versionskontrolle (Objekt 2110h, Version control) durch Setzen von Bit5 auf Null. Ein Heartbeat Producer produziert zyklisch eine so genannte Heartbeat-Meldung (siehe Figur 3). Ein oder mehrere Heartbeat Consumer können diese Heartbeat Meldung empfangen. Falls das zyklische Senden dieser Heartbeat Meldung ausbleibt, wird ein so genannter Heartbeat Event ausgelöst. Das Verhalten im Fehlerfall wird definiert im Objekt Error behavior - Communication error (Objekt 1029h-01h). Beispielsweise kann der Slave (Antrieb) als ein Consumer definiert werden, der beim Ausbleiben der Producer Meldung der Masters (Steuerung) stoppt. Gleichzeitig überwacht der Master den Slave entweder ebenfalls über Heartbeat oder alternativ über seine SDO Antworten.



Figur 3: Heartbeat Protokoll.

Beispiel eines Heartbeat Protokolls:

COB-ID	Data/Remote	Byte 0
700h+Node-ID	d	7Fh (127d)

Die Heartbeat Meldungen bestehen aus der COB-ID und einem Byte, mit dem der NMT-Zustand übertragen wird:

0: BootUp Meldung
4: Stopped/Prepared Mode
5: Operational Mode
127: Pre-Operational, d.h. der Antrieb befindet sich im Pre-Operational Mode (7Fh = 127).

2.2.9 Layer Setting Services (LSS)

Um mehrere Busteilnehmer mit gleicher Node-ID auf dem selben Bussystem ansprechen zu können, wird LSS verwendet (siehe Vorgehen unter Layer Setting Services and Protocol, CiA Draft Standard Proposal 305).
Jeder Busteilnehmer mit LSS besitzt eine eindeutige Seriennummer, über die er angesprochen und initialisiert werden kann. Anschließend können sowohl Node-ID als auch Baudrate neu gesetzt werden. LSS kann nur im Stopped/Prepared Mode ausgeführt werden.

Meldungsaufbau

COB-ID

Master → Antrieb : 2021 = 7E5h
Master ← Antrieb : 2020 = 7E4h
Nach der COB-ID wird ein LSS Command Specifier gesandt. Danach folgen bis zu sieben Datenbyte:

COB-ID	Command Specifier	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6

Switch Mode Global

7E5h →	04h	Mode	reserviert

Mode: : 0 → Operationsmode
1 → Konfigurationsmode

Switch Mode Selektiv

Mit folgendem Ablauf kann ein ganz bestimmter Busteilnehmer im System angesprochen werden.

7E5h →	40h	Vendor-ID	reserviert
7E5h →	41h	Product code	reserviert
7E5h →	42h	Revision number	reserviert
7E5h →	43h	Serial number	reserviert

7E4h ←	44h	Mode	reserviert

Vendor-ID : 5Fh
Product code : Interner Produktcode für den jeweiligen Busteilnehmer
Revision number : Aktuelle Revisionsnummer des Busteilnehmers
Serial number : Eindeutige, fortlaufende Seriennummer
Mode : Antwort des Busteilnehmers ist der neue Mode (0=Operating mode; 1=Konfigurationsmode)

Node-ID setzen

7E5h →	11h	Node-ID	reserviert

7E4h ←	11h	Error code	Specific error	reserviert

Node-ID : Neue Node-ID des Busteilnehmers
Error code : 0=OK; 1=Node-ID außerhalb des Bereiches; 2..254=reserviert; 255→Specific error
Specific error : Falls ErrorCode=255 → Applikationsspezifischer Error code.

Baudrate setzen

7E5h →	13h	Table Sel	Table Ind	reserviert

7E4h ←	13h	Error code	Specific error	reserviert

Table Sel : Selektiert die Baudraten-Tabelle (default = 0: Standard CiA Baudraten-Tabelle)
Table Ind : Baudraten-Index in Baudraten-Tabelle (siehe Tabelle 8).
Error code : 0=OK; 1= Baudrate außerhalb des Bereiches; 2..254=reserviert; 255→Specific error
Specific error : Falls Error code=255 → Applikationsspezifischer Error code.

Tabelle 8: Standard CiA Baudraten.

Baudrate [kBit/s]	10	20	50	100	125	250	500	800	1000
Index (LSS) nach CiA-Tabelle	8h	7h	6h	5h	4h	3h	2h	1h	0h
Index (Definition in Objekt 2100h)	0h	1h	2h	3h	4h	5h	6h	7h	8h

Die Baudrate kann auch im Objekt 2100h geändert werden. Hierbei wird ebenfalls ein Index ins Objekt geschrieben, nicht die effektive Baudrate. ACHTUNG: Die Indices für Objekt 2100h und LSS unterscheiden sich.

Speichern des Konfigurationsprotokoll

Dieses Protokoll speichert die Konfigurationsparameter im EEPROM.

7E5h →	17h	reserviert

7E4h ←	17h	Error code	Specific error	reserviert

Error Code : 0=OK;1=Speichern nicht unterstützt;2=Zugriffsfehler;3..254=reserviert;255→Specific error
Specific error : Falls ErrorCode=255 → Applikationsspezifischer Error code.

Aktiviere BitTiming Parameter

Die neuen BitTiming Parameter werden mit dem Command Specifier 21 aktiviert.

7E5h →	15h	16 Bit Switch Delay	reserviert

Switch Delay : Verzögerung des Antrieb Resets in ms.
Nach der Verzögerungszeit meldet sich der Busteilnehmer mit der neuen Baudrate an.

Vendor-ID anfordern

Vendor-ID eines selektierten Busteilnehmers anfordern

7E5h →	5Ah	reserviert	

7E4h ←	5Ah	32 Bit Vendor-ID	reserviert

Vendor-ID : = 5Fh

Produktcode anfordern

Produktcode eines selektierten Busteilnehmers anfordern

7E5h →	5Bh	reserviert	

7E4h ←	5Bh	Product code	reserviert

Product code : Herstellerabhängiger Produktcode

Revisionsnummer anfordern

Revisionsnummer eines selektierten Busteilnehmers anfordern

7E5h →	5Ch	reserviert	

7E4h ←	5Ch	32 Bit Revisionsnummer	reserviert

Revisionsnummer : aktuelle Revision

Seriennummer anfordern

Seriennummer eines selektierten Busteilnehmers anfordern

7E5h →	5Dh	reserviert	

7E4h ←	5Dh	32 Bit Serial number	reserviert

Seriennummer : eindeutige fortlaufende Seriennummer des Busteilnehmers

Bereichsanfrage

Busteilnehmer können auch in einem gewissen Bereich gesucht werden. Hierzu werden folgende Objekte nacheinander versandt:

7E5h →	46h	Vendor-ID	reserviert

7E5h →	47h	Product code	reserviert

7E5h →	48h	Revision number LOW	reserviert
7E5h →	49h	Revision number HIGH	reserviert

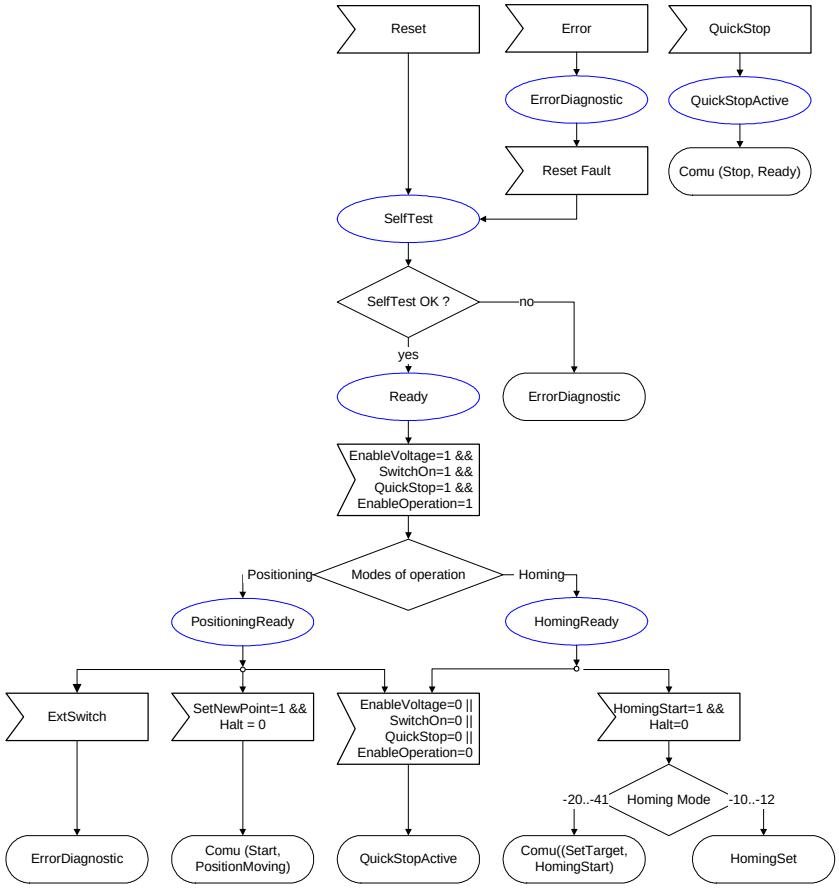
7E5h →	4Ah	Serial number LOW	reserviert
7E5h →	4Bh	Serial number HIGH	reserviert

Jeder Busteilnehmer mit den entsprechenden Parametern meldet sich mit folgender Meldung:

7E4h ←	4Fh	reserviert

2.3 Zustandsdiagramm für Positioning / Homing

Beim Starten des Antriebs wird zuerst das NMT Zustandsdiagramm aktiviert. In den NMT-Zuständen Pre-Operational Mode oder Operational Mode gilt folgendes Zustandsdiagramm für den Modus Positioning und den Modus Homing (siehe Objekt 6060h).



Figur 4: Zustandsdiagramm für Modus Positioning und Modus Homing.

Selftest
Nach einem Reset des Antriebs wird ein Selbsttest durchgeführt. Hier wird zuerst ein Reset an den Regler gesendet. Danach werden alle Parameter mit dem Regler ausgetauscht und auf Richtigkeit überprüft. Zudem werden Abhängigkeiten der einzelnen Werte geprüft. Nach erfolgreichem Test wird automatisch in den Ready Mode gewechselt.

Ready
Im Ready Mode können beliebige Parameter gesetzt und gelesen werden. Dieser Zustand ist der Grundzustand des Antriebes. Dieser Zustand wird erst verlassen, wenn vom Master dazu aufgefordert wird.

PositioningReady
Wechsel vom Ready Mode erfolgt durch:
1. Objekt 6060h (Modes of operation) auf 1 = Positioning setzen
2. Objekt 6040h (Kontrollwort) auf 000Fh setzen
   SwitchOn, EnableVoltage, QuickStop und EnableOperation = 1

Im PositioningReady Mode wartet der Antrieb nur noch auf das Start-Signal, welches mit dem Kontrollwort gesetzt wird. QuickStop sowie Ansprechen der Schalteingänge werden in diesem Zustand speziell verarbeitet.

HomingReady (Manueller Modus)
Wechsel vom Ready Mode erfolgt durch:
1. Objekt 6060h (Modes of operation) auf 6=Modus Homing setzen
2. Objekt 6040h (Kontrollwort) auf 000Fh setzen
   SwitchOn, EnableVoltage, QuickStop und EnableOperation = 1

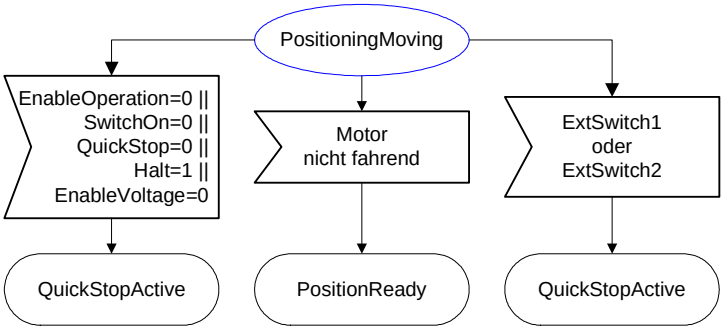
In diesem Zustand wird mit dem Objekt 6098h (Homing method) die gewünschte Referenzierungsfunktion vorgewählt. Mit dem Kontrollwort, Bit4 (HomingOperationStart) wird in den HomingMove Mode gewechselt.

QuickStop active
QuickStop ist kein eigentlicher Zustand. Falls in den QuickStop gewechselt wird, wird ein QuickStop an den Regler gesendet und in den Ready Mode gewechselt. Falls ein Fehler anliegt, wird erst dann in den Zustand ErrorDiagnostic gewechselt.

ErrorDiagnostic
Im ErrorDiagnostic wird eine EMCY-Meldung abgesetzt (Error code siehe Tabelle 18) und auf eine Entscheidung vom Master gewartet. Parameter können auch in diesem Zustand beliebig gelesen und geschrieben werden. Nach einem Error reset (Kontrollwort) wird in den SelfTest Mode gewechselt, wo auch der Regler einen Reset des Fehlers erhält.

PositioningMoving

Das Ziel ist im Regler immer aktuell. Deshalb wird nur noch der Startbefehl gesendet und in den PositioningMoving Mode gewechselt.



Figur 5: PositioningMoving Mode.

Events:

1. Software-Stoppbefehl über CAN -> QuickStop an Regler senden
2. Motor nicht mehr fahrend -> Sollposition erreicht
3. Externer Hardware-Endschalter spricht an -> QuickStop an Regler senden

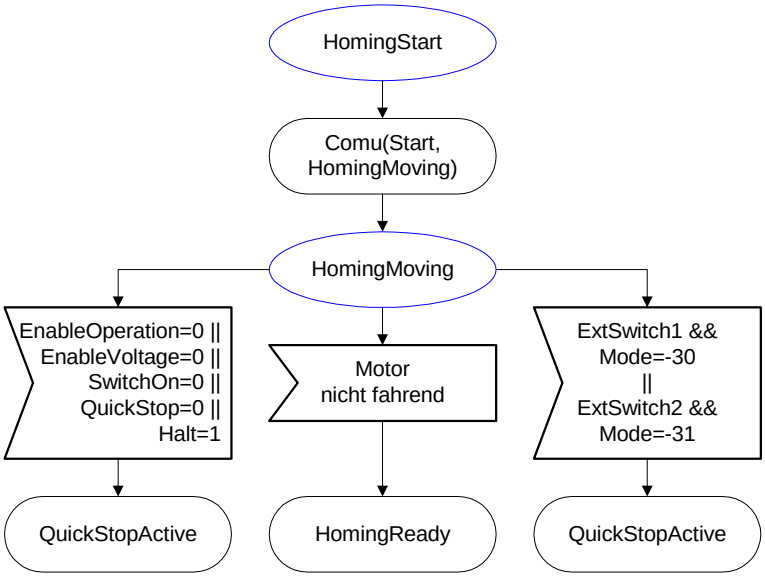
Homing

Homing wird in zwei verschiedene Vorgänge unterteilt. Im HomingMoving wird der Motor in Bewegung gesetzt. Im HomingSet hingegen werden nur Werte abgespeichert.

HomingMoving

Neues Ziel ist gesetzt. Danach wird der Startbefehl an den Regler gesendet und in den HomingMoving Mode übergegangen.
Events:

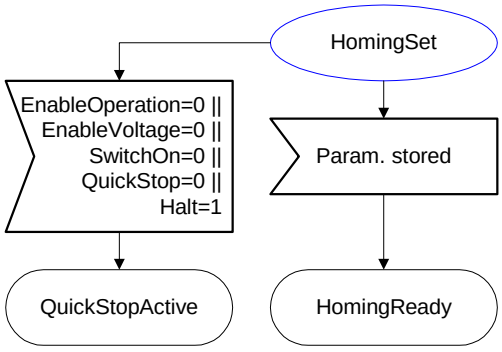
1. Software-Stoppbefehl über CAN -> QuickStop an Regler senden
2. Motor nicht mehr fahrend -> Ziel erreicht
3. Externe Schalter sprechen an -> QuickStop an Regler senden



Figur 6: HomingMoving.

HomingSet

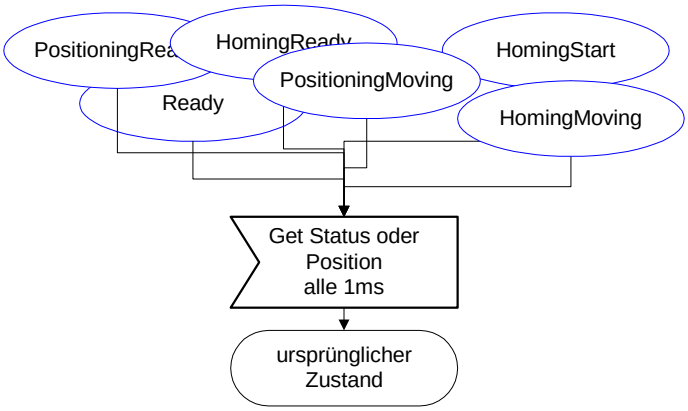
Im HomingSet Mode werden nur Parameter wie z.B. Position der Software-Endschalter gesetzt.



Figur 7: HomingSet.

Status oder Position einlesen

In den meisten Zuständen werden abwechselnd alle 1ms Status oder aktuelle Position eingelesen.



Figur 8: Status oder aktuelle Position einlesen.

Kontrollwort

Zustandswechsel werden über das Kontrollwort gesteuert.
Ein Master sendet folgendes Kontrollwort um den Antrieb zu aktivieren: **xxxx xxx0 0xxx 1111**
Ein Master sendet folgendes Kontrollwort um den Antrieb zu starten: **xxxx xxx0 0xx1 1111**

Tabelle 9: Kontrollwort (Objekt 6040h). Bit 13...15 sind nicht definiert.

Bit	Beschreibung
0	SwitchOn
1	EnableVoltage
2	QuickStop
3	EnableOperation
4	StartAction (Fahre zum Ziel)
5	Change set immediately
6	Relative/absolute movement
7	Error reset
8	Halt
11	Jogging+
12	Jogging-

Beispiel für einen Zustandswechsel:

Aktion	Kontrollwort Bit					Beschreibung
	7	3	2	1	0	
SwitchOn	0	**1**	**1**	**1**	**1**	Ready → PositioningReady oder HomingReady (je nach Objekt 6060h)
DisableOperation	0	**0**	1	1	1	PositioningReady oder HomingReady → Ready
DisableVoltage	0	1	1	**0**	1	PositioningReady oder HomingReady → Ready
SwitchOff	0	1	1	1	**0**	PositioningReady oder HomingReady → Ready
QuickStop	0	**1**	**0**	1	1	Von PositioningReady oder HomingReady → QuickStop active Falls Motor drehend, wird dies speziell behandelt. Falls QuickStop wieder 1 gesetzt wird, kann direkt zu OperationEnabled gewechselt werden.

2.4 Betrieb des Antriebs

2.4.1 Antriebsstatus und aktuelle Position auslesen

Je nach NMT-Zustand kann der Status des Antriebs als SDO (Objekt 6041h) ausgelesen oder als PDO1 zyklisch bzw. synchron angefordert werden.

COB-ID	DLC	Byte 0	Byte 1	Byte 2
180h+Node-ID	3	xx	xx	xx

Die effektive Position kann als SDO ausgelesen werden oder wird zyklisch oder synchron als PDO2 gesendet.

Anfrage der Position beim Antrieb → Objekt 6064h:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
600h+Node-ID	8	40h	64h	60h	0	0	0	0	0

Antwort des Antriebs auf die Anfrage der Position (abcd):

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	43h	64h	60h	0	a	b	c	d

Üblicher und praktischer wäre die Aktualisierung der Position mittels Prozessdatenobjekte (PDO). PDO2 ist standardmäßig auf das Objekt 6064h (Position) eingestellt. Hierbei kann zwischen synchroner und asynchroner PDO-Übertragung gewählt werden (siehe Objekt 1400h und folgende).

PDO2 Telegrammaufbau:

COB-ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3
280h+Node-ID	4	Ww	Xx	Yy	Zz

Byte1 bis Byte4 beinhalten die Position.

2.4.2 Sollposition anfahren über SDO (mit Fahrauftrag)

Sollposition wird gesetzt (Objekt 607Ah):

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
600h+Node-ID	8	23h	7Ah	60h	0	x	x	x	x

Antwort:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	60h	7Ah	60h	0	0	0	0	0

Es wird ein Start ins Kontrollwort (Objekt 6040h) geschrieben:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
600h+Node-ID	8	23h	40h	60h	0	1Fh	0	0	0

Antwort:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	60h	40h	60h	0	0	0	0	0

Der Antrieb fährt los.

2.4.3 Fahrauftrag stoppen

Es wird ein Stop ins Kontrollwort geschrieben:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
600h+Node-ID	8	23h	40h	60h	0	1Fh	01h	0	0

Antwort:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	60h	40h	60h	0	0	0	0	0

Der Antrieb wird gestoppt.

2.4.4 Ziel anfahren über R_PDO (mit Fahrauftrag)

NMT Start:

COB-ID	DLC	Byte 0	Byte 1
0	2	01h	Node-ID

Antwort T_PDO1 (Statuswort Objekt 6041h):

COB-ID	DLC	Byte 0	Byte 1	Byte 2
180h+Node-ID	3	21h	0Ah	01h

Antwort T_PDO2 (Ist-Position Objekt 6064h):

COB-ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3
280h+Node-ID	4	Ww	Xx	Yy	Zz

Antrieb in den PositioningReady Mode setzen mit dem R_PDO1 (Kontrollwort Objekt 6040h):

COB-ID	DLC	Byte 0	Byte 1
200h+Node-ID	2	0Fh	00h

Antwort T_PDO1 (Statuswort Objekt 6041h):

COB-ID	DLC	Byte 0	Byte 1	Byte 2
180h+Node-ID	3	37h	0Ah	02h

Kontrollwort + Ziel wird gesetzt mit dem R_PDO2 (Objekt 6040h, 607Ah):

COB-ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
300h+Node-ID	6	1Fh	00h	x	x	x	x

Wenn R_PDO2 auf synchron eingestellt ist, muss noch das SYNC-Telegramm gesendet werden:

COB-ID	DLC
80h	0

Der Antrieb fährt los.

2.5 Befehlssynchronisation zwischen Antrieb und Steuerung

Für die CAN-Bus Kommunikation gelten allgemein die Anforderungen:

- Die Busbelastung soll möglichst klein sein (zyklische Kommunikation sollte daher eher vermieden werden, mit Ausnahme des Nodeguarding).
- Ein Startbefehl soll möglichst schnell ausgeführt werden, um kurze Zykluszeiten zu erreichen.

Bei CANopen erfolgt der Fahrbefehl vom CAN-Bus Master an den Antrieb über ein Kontrollwort (PDO-RX oder SDO Objekt 6040h). Für die Statusanzeige sieht CANopen ein PDO-TX vor (Statuswort Objekt 6041h), in dem die Statusinformationen des Antriebs (Drive moving, Target position reached, etc.) zusammengefasst sind. Mit diesem Statuswort müssen bei einem Fahrbefehl Master und Antrieb synchronisiert werden. Um die Anforderungen nach minimaler Buslast und Zykluszeit zu erfüllen, sendet der Antrieb das Statuswort nur bei Änderungen.

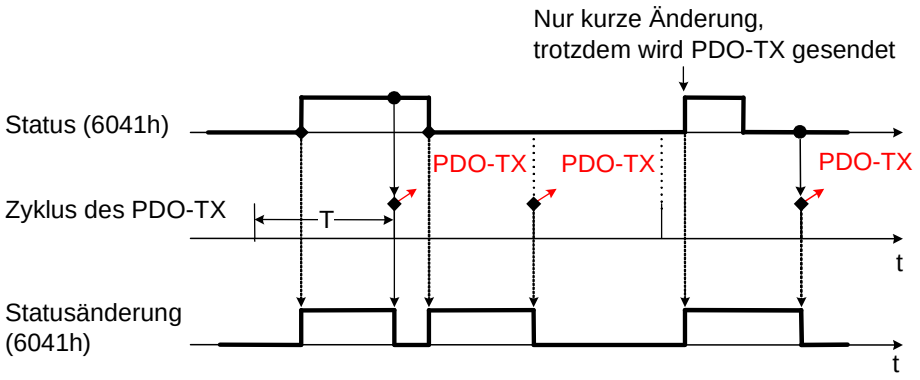
Zur Befehlssynchronisation ist im Antrieb folgendes implementiert:

- Nach jeder Statusänderung wird ein PDO-TX gesendet.
- Jeder Fahrbefehl erzeugt eine Statusänderung.

Diese beiden Punkte werden im folgenden erläutert. Zusätzlich muss bei einem Fahrbefehl auf einen korrekten und vollständigen Ablauf der Kommunikation zwischen Master und Antrieb geachtet werden.

Nach jeder Statusänderung wird ein PDO-TX gesendet

Der Antrieb hat einen einstellbaren Zyklus mit Zykluszeit T (Transmit PDO2 Event Timer, Objekt 1801h-05h) für das Senden eines PDO-TX. Gesendet wird dabei normalerweise immer der Status (Objekt 6041h) zum Sendezeitpunkt. Der Antrieb muss jedoch auch dann ein PDO-TX senden, wenn nur eine kurze Statusänderung innerhalb eines Zyklus stattfindet und diese Statusänderung noch vor dem nächsten Zyklus des PDO-TX wieder verschwindet (siehe Figur 9). Zur Befehlssynchronisation ist dies im Antrieb implementiert. Nur wenn keine Änderung stattgefunden hat, wird kein PDO-TX gesendet.

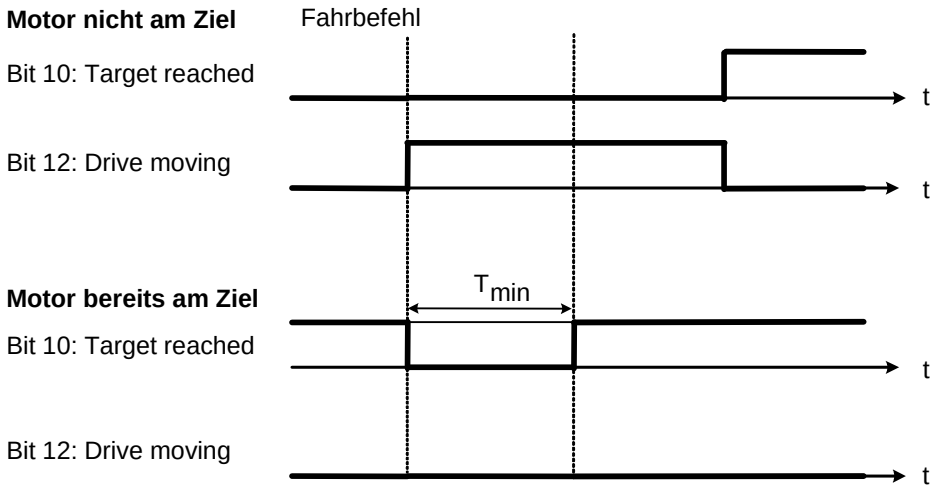


Figur 9: Nach jeder Statusänderung wird ein PDO-TX gesendet.

Jeder Fahrbefehl erzeugt eine Statusänderung

Sendet der Master einen Fahrbefehl an einen Antrieb mit einer Sollposition weit weg von der aktuellen Position, so ergeben sich klare Statusänderungen und das Statusbit 10 „Target position reached“ wird erst gesetzt beim Erreichen der Zielposition. Wenn sich jedoch der Antrieb bereits an der Sollposition befindet, dann würde normalerweise keine Statusänderung erzeugt werden. Dies könnte zu fehlerhafter Synchronisierung zwischen Master und Antrieb führen. Zur Befehlssynchronisation wird daher im Antrieb in einem solchen Fall das Statusbit 10 „Target position reached“ für eine Zeit Tmin zwangläufig auf Null gesetzt, nachdem ein Fahrbefehl empfangen wurde. Dies gilt auch beim Referenzieren und Einmessen, wo die entsprechenden Statusbits zunächst für Tmin auf Null und beim Beenden wieder auf Eins gesetzt

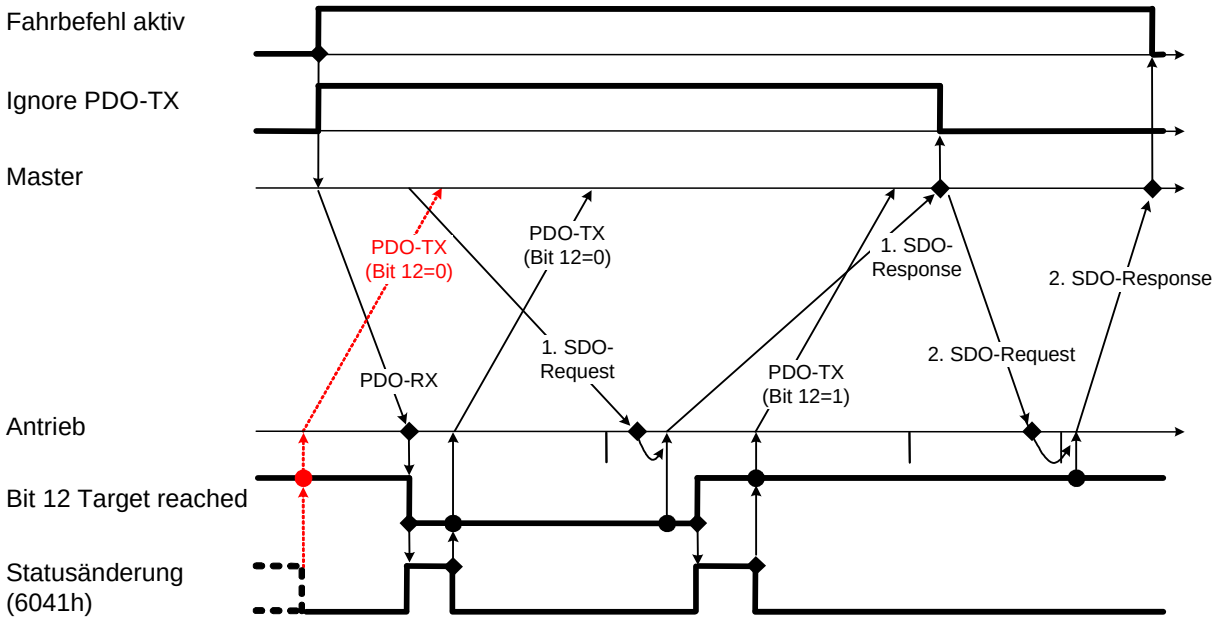
werden. Somit ist gewährleistet, dass bei allen Befehlen eine entsprechende Statusänderung erzeugt und ein entsprechende PDO-TX gesendet werden.



Figur 10: Jeder Fahrbefehl erzeugt eine Statusänderung (auch beim Referenzieren und Einmessen).

Korrekter Kommunikationsablauf mit bis zu zwei SDO Abfragen des Statusworts 6041h

Das undefinierte Situation könnte möglicherweise entsteht, wenn zufällig gleichzeitig mit dem Senden eines Fahrauftrags PDO-RX durch den Master eine Statusänderung am Antrieb erfolgt und dieser somit ein PDO-TX sendet. In diesem Fall empfängt der Master nach dem Absetzen eines Fahrbefehles einen Status, der für ihn jedoch aus der Vergangenheit kommt und den er auf keinen Fall beachten darf. Um diese unerwünschten PDO-TX Meldungen nach einem Fahrauftrag zu unterdrücken, müssen gleichzeitig mit dem Senden des Fahrauftrags beim Master die PDO-TX Meldungen gesperrt werden (Ignore PDO-TX = 1). Um die Sperrung wieder zu aufzuheben, fragt der Master den Status über den 1. SDO-Request ab. Sobald er die 1. SDO Response empfängt, hebt er die Sperrung auf (Ignore PDO-TX = 0). Dann werden empfangene PDO-TX Meldungen wieder interpretiert. Wenn die 1. SDO-Response den Status „Target position reached = 1“ enthält, ist der Fahrbefehl korrekt abgeschlossen. Wenn ein nach Aufhebung der Sperrung empfangenes PDO-TX den Status „Target position reached = 1“ enthält, ist der Fahrbefehl ebenfalls korrekt abgeschlossen.
Es kann jedoch vorkommen, dass eine PDO-TX Meldung aufgrund ihrer hohen Priorität die 1. SDO-Response überholt und daher unterdrückt wird (siehe Figur 11). Der Fahrbefehl kann in diesem Fall nur durch eine zweite SDO-Request mit anschließender SDO-Response korrekt abgeschlossen werden. Auch beim Referenzieren und Einmessen muss dieser Kommunikationsablauf unbedingt in dieser Form realisiert werden.



Figur 11: Korrekter Kommunikationsablauf mit bis zu zwei SDO Abfragen des Statusworts 6041h.

3 Objektverzeichnis

Alle Objekte für den Betrieb des Antriebs sind beschrieben in Tabelle 10. Nach CiA (CAN in Automation) werden die Objekte in drei Gruppen unterteilt:

- **Standard-Objekte**
  1000h, 1001h, 1018h
- **Herstellerspezifische Objekte**
  2000h - 5FFFh
- **Gerätespezifische Objekte**
  Alle anderen Objekte von 1000h - 1FFFh, 6000h - FFFFh

Sendet der CAN Master ein unzulässiges Kommando an den Antrieb, so wird als Antwort eine Abort Message mit Abort Code gesendet (siehe Tabelle 19). Die Änderung von Objekten mit Schreibzugriff kann nichtflüchtig gespeichert werden in einem EEPROM im Antrieb. Nach Einschalten des Antriebs werden alle Parameter aus diesem EEPROM geladen.

Tabelle 10: Zusammenfassung aller unterstützten Objekte. Speichern der aktuellen Objekt-Konfiguration, siehe Objekt 1010h. Laden der Default-Objekte: siehe Objekt 1011h. Für weitere Details, siehe Kapitel 4.2. Hauptunterschiede zu FlexiDrive MSQA sind hellblau hinterlegt.

Objekt	Objekt-Nummer in hexadezimaler Darstellung (Beispiel 1000h)
Name	**---**
Format	Variablenformat des Objekts: U/I = Unsigned/Integer, Zahl=Anzahl Bit, ARR = Array, REC = Record
Zugriff	ro = read only, wo = write only, rw = read / write
Default	Default-Wert des Parameters nach der ersten Initialisierung oder nach Laden der Default-Parameter (Objekt 1011h)
Save	Speichern bzw. Laden des Parameters (EEPROM) über Objekte 1010h und 1011h, Subindex 1, 2, 3 oder 4
Beschreibung	Beschreibung der Objektoptionen

Objekt	Name	Format	Zugriff	Default	Save	Beschreibung
1000h	Device type	U32	ro	00020192h 01h 02h 192h	-	Gerätetyp gemäss CiA Motortyp = 01h (Bürstenloser DC Motor) Antriebstyp = 02h (ServoDrive) Profilnummer = 192h = 402
1001h	Error register	U8	ro	0h	-	Fehlerregister mit Fehlerklasse (siehe auch Tabelle 17). Bit 2 = 1 Spannungs-Fehler Bit 3 = 1 Temperatur-Fehler Bit 4 = 1 CAN-Bus Kommunikations-Fehler Bit 5 = 1 Gerätespezifischer Fehler
1003h	Predefined error field	ARR			-	Schieberegister für die letzten 8 Fehler oder Warnungen
00h	Grösster Subindex	U8	rw	0h	-	-
01h	Last entry	U32	ro		-	Zuletzt aufgetretener Fehler (siehe Tabelle 18 oder Objekt 603Fh). Error code / Beschreibung 0000h Kein Fehler 3110h Fehler Überspannung 3120h Fehler Unterspannung 4210h Fehler / Warnung Übertemperatur (zur Unterscheidung, siehe Zusatz-Information in EMCY-Meldung bzw. Tabelle 18) 5441h Warnung unterer Hardware-Endschalter aktiv 5442h Warnung oberer Hardware-Endschalter aktiv 5530h Warnung Speicher (EEPROM) 6010h Warnung Software (Watchdog) 7121h Warnung Motor blockiert 7320h Fehler Encoder 8110h Warnung CAN-Bus Kommunikation 8130h Fehler Lifeguard oder Heartbeat

Objekt	Name	Format	Zugriff	Default	Save	Beschreibung
						FF00h Warnung Data valid multiturn (DVMT) FF02h Fehler Positioning timeout FF03h Warnung Not moving timeout FF10h Warnung unterer Software-Endschalter aktiv FF11h Warnung oberer Software-Endschalter aktiv
...	...	...	...	...	-	...
08h	First entry	U32	ro		-	Beschreibung siehe Subindex 01h
1005h	SYNC COB-ID	U32	rw	80h	1, 2	COB-ID des SYNC Objekts
1008h	Device name	U32	ro	MSIA	-	Gerätename in ASCII (Beispiel: MSIA = „AISM“ = 41h 49h 53h 4Dh)
1009h	Hardware version	U32	ro	werksseitig	-	Hardware Version des Geräts in ASCII (Beispiel: 2.01 = „1 0 . 2“ = 31h 30h 2Eh 32h)
100Ah	Software version	U32	ro	werksseitig	-	Unterversion für Gesamtversion der Firmware des Geräts in ASCII. Sie bezieht sich auf die Korrektur eines Softwarefehlers. Für Gesamtversion, siehe Objekt 1018h-3h. Beispiel: 0017 = 17.
100Ch	Guard time	U16	rw	0h	1, 2	Guard time für Node Guarding (siehe Figur 2).
100Dh	Life time factor	U8	rw	0h	1, 2	Life time factor für Node Guarding (siehe Figur 2). Die Knoten-Lebensdauer ergibt sich durch Guard Time multipliziert mit Life time factor. Im Fehlerfall, beispielsweise nach Ablauf der Knoten-Lebensdauer wird ein Life Guarding Event ausgelöst. Das Fehlerverhalten wird definiert im Objekt Error behavior (Objekt 1029h). ACHTUNG: Einschalten des Node Guarding Protokolls erfolgt in der Versionskontrolle (Objekt 2110h, Bit5).
1010h	Store parameters	ARR				Speichern der aktuellen Parameter des Antriebs. Spalte Save in dieser Tabelle gibt für jedes Objekt den benötigten Subindex zum Speichern der Parameter an.
00h	Grösster Subindex	U8	ro	6h		-
01h	Store all parameters	U32	rw			Speichern aller Parameter im EEPROM durch Schreiben von save = „evas“ = 65h 76h 61h 73h
02h	Store communication parameters	U32	rw			Speichern der Kommunikationsparameter im EEPROM durch Schreiben von save = „evas“ = 65h 76h 61h 73h
03h	Store application parameters	U32	rw			Speichern der Applikationsparameter im EEPROM durch Schreiben von save = „evas“ = 65h 76h 61h 73h
04h	Store manufacturer specific parameters	U32	rw			Speichern der herstellerspezifischen Parameter im EEPROM durch Schreiben von save = „evas“ = 65h 76h 61h 73h
06h	Store regulator parameters	U32	rw			Speichern der Reglerparameter im EEPROM durch Schreiben von save = „evas“ = 65h 76h 61h 73h
1011h	Restore parameters	ARR				Laden der Default-Parameter des Antriebs. Spalte Save in dieser Tabelle gibt für jedes Objekt den benötigten Subindex für Laden der Default-Parameter an.
00h	Grösster Subindex	U8	ro	4h		-
01h	Restore all parameters	U32	rw			Laden aller Parameter aus dem EEPROM durch Schreiben von load = „daol“ = 64h 61h 6Fh 6Ch
02h	Restore communication parameters	U32	rw			Laden der Kommunikationsparameter aus dem EEPROM durch Schreiben von load = „daol“ = 64h 61h 6Fh 6Ch
03h	Restore application parameters	U32	rw			Laden der Applikationsparameter aus dem EEPROM durch Schreiben von load = „daol“ = 64h 61h 6Fh 6Ch

Objekt	Name	Format	Zugriff	Default	Save	Beschreibung
04h	Restore manufacturer specific parameters	U32	rw			Laden der herstellerspezifischen Parameter aus dem EEPROM durch Schreiben von load = „daol“ = 64h 61h 6Fh 6Ch
1014h	Emergency COB-ID	U32	rw	81h	1, 2	COB-ID des Emergency Objektes EMCY COB-ID = 80h + Node-ID
1016h	Consumer heartbeat	ARR				
00h	Grösster Subindex	U8	ro	1h		-
01h	Consumer heartbeat time	U32	rw	00010000h 0001h 0000h	1, 2	Consumer Heartbeat Zeitintervall (siehe Figur 3). Bit16..23: Node- ID Bit0..15: Consumer Heartbeat Zeitintervall in ms ACHTUNG: Einschalten des Heartbeat Protokolls in der Versionskontrolle (Objekt 2110h, Bit5).
1017h	Producer heartbeat time	U16	rw	0h	1, 2	Producer Heartbeat Zeitintervall in ms
1018h	Identity object	REC				Hersteller- und Geräteidentifikation für LSS
00h	Grösster Subindex	U8	ro	4h		-
01h	Vendor-ID	U32	ro	5Fh		Nummer zur Identifikation des Geräteherstellers nach CiA. (Baumer Electric AG = 5Fh)
02h	Product code	U32	ro	00000100h		Nummer zur Identifikation des Geräts (MSIA = 100h)
03h	Revision number	U32	ro	werkseitig		Firmware-Version des Geräts (Gesamtversion). Beispiel: Gesamtversion 1.00 = 0001h’0000h. Hauptindex und Subindex beziehen sich auf Gerätefunktionen (1.00.xx). Der letzte Index (xx) bezieht sich auf die Korrektur eines Softwarefehlers und ist gespeichert in Objekt 100Ah Software version. Diese Firmware-Version wird verwendet zur LSS Identifikation des Geräts und muss identisch sein mit der Version des Handbuchs sowie mit dem Aufdruck RevNr auf dem Typenschild auf der Aussenseite des Antriebs.
04h	Serial number	U32	ro	werkseitig		Eindeutige fortlaufende Seriennummer des Geräts
1029h	Error behaviour	ARR				Fehlerbehandlung von NMT-Kommunikationsfehlern.
00h	Grösster Subindex	U8	ro	1h		-
01h	Communication error	U8	rw	1h	1, 2	Verhalten nach Kommunikationsfehler: 0h = Wechsel in den Pre-Operational Mode 1h = kein Zustandswechsel 2h = Wechsel in den Stopped/Prepared Mode 3h = Reset des Antriebs
1400h	Receive PDO1 parameter	REC				
00h	Grösster Subindex	U8	ro	2h		-
01h	COB-ID	U32	rw	200h + ID	1, 2	PDO-ID = 200h + Node-ID
02h	R_PDO1 type	U8	rw	FEh	1, 2	1h..F0h = Synchroner Betrieb FEh = Asynchroner Betrieb
1401h	Receive PDO2 parameter	REC				
00h	Grösster Subindex	U8	ro	2h		-
01h	COB-ID	U32	rw	300h + ID	1, 2	PDO-ID = 300h + Node-ID
02h	R_PDO2 type	U8	rw	FEh	1, 2	1h..F0h = Synchroner Betrieb FEh = Asynchroner Betrieb

Objekt	Name	Format	Zugriff	Default	Save	Beschreibung
1600h	Receive PDO1 mapping	ARR				
00h	Grösster Subindex	U8	ro	1h		-
01h	R_PDO1 content	U32	ro	60400010h		ReadOnly, obwohl von CiA als ReadWrite
1601h	Receive PDO2 mapping	ARR				
00h	Grösster Subindex	U8	ro	1h		-
01h	R_PDO2 content	U32	ro	60400010h		ReadOnly, obwohl von CiA als ReadWrite
02h	R_PDO2 content	U32	ro	607A0020h		ReadOnly, obwohl von CiA als ReadWrite
1800h	Transmit PDO1 parameter	REC				
00h	Grösster Subindex	U8	ro	5h		-
01h	COB-ID	U32	rw	180h + ID	1, 2	PDO-ID = 180h + Node-ID
02h	T_PDO1 type	U8	rw	FEh	1, 2	1h..F0h = Synchroner Betrieb FEh = Asynchroner Betrieb
05h	Event timer	U16	rw	203h	1, 2	Zykluszeit in ms
1801h	Transmit PDO2 Parameter	REC				
00h	Grösster Subindex	U8	ro	5h		-
01h	COB-ID	U32	rw	280h + ID	1, 2	PDO-ID = 280h + Node-ID
02h	T_PDO2 type	U8	rw	FEh	1, 2	1h..F0h = Synchroner Betrieb FEh = Asynchroner Betrieb
05h	Event timer	U16	rw	100h	1, 2	Zykluszeit in ms
1A00h	Transmit PDO1 mapping	ARR				
00h	Grösster Subindex	U8	ro	2h		-
01h	T_PDO1 content	U32	ro	60410010h		ReadOnly, obwohl von CiA als ReadWrite
02h	T_PDO1 content	U32	ro	60FD0020h		ReadOnly, obwohl von CiA als ReadWrite
1A01h	Transmit PDO2 mapping	ARR				
00h	Grösster Subindex	U8	ro	1h		-
01h	T_PDO2 content	U32	ro	60640020h		ReadOnly, obwohl von CiA als ReadWrite
2100h	Baudrate	U8	rw	2h	1, 2	Baudrate des Antriebs für Buskommunikation (siehe Tabelle 14). 0h = 10 kBit/s 1h = 20 kBit/s 2h = 50 kBit/s 3h = 100 kBit/s 4h = 125 kBit/s 5h = 250 kBit/s 6h = 500 kBit/s 7h = 800 kBit/s 8h = 1000 kBit/s ACHTUNG: Nach Änderung der Baudrate muss Antrieb neu initialisiert werden, beispielsweise durch Aus-Ein-Schalten.
2101h	Node-ID	U8	rw	1h	1, 2	Knotennummer bzw. Teilnehmeradresse des Antriebs auf dem CAN-Bus. Mögliche Werte sind 1...127. ACHTUNG: Nach Änderung der Node-ID muss der Antrieb neu initialisiert werden, beispielsweise durch Aus-Ein-Schalten.

Objekt	Name	Format	Zugriff	Default	Save	Beschreibung
2102h	Node-ID offset	U8	rw	0h	1, 2	Offset für Node-ID. Mögliche Werte sind 0..126. Node-ID Offset wird sofort nach Eingabe zur Node-ID addiert und aktiviert ohne Initialisierung. Node-ID Offset kann nicht dauerhaft abgespeichert werden und ist nach Initialisierung beispielsweise durch Ein-Aus-Schalten wieder auf 0h. Zur Bestätigung der Änderung des Node-ID offset sendet der Antrieb eine Download Response mit neuer Node-ID (siehe SDO Kommandos in Tabelle 2). ACHTUNG: Der Node-ID Offset kann nicht abgespeichert werden.
2110h	Version control	U32	rw	Ah	1, 3	Versionskontrolle: Definition von PDO1 Inhalt und Verhalten im Fehlerfall Bit1: PDO1 Standardinhalt 0 = PDO1 enthält Statuswort (Tabelle 4) mit 2 Byte Länge 1 = PDO1 enthält Statuswort (Tabelle 4) und Statuswort-Zusatz (Tabelle 5) mit 3 Byte Gesamtlänge (Default). Bit3: Verhalten bei BusOFF 0 = bei BusOFF zieht sich Antrieb vom Bus zurück 1 = bei BusOFF automatisch Initialisierung (Default) Bit5: 0 = Heartbeat (Default) 1 = Nodeguarding Bit9: Servicebetrieb (Jogging ohne Bus möglich) 0 = ausgeschaltet (Default) 1 = eingeschaltet Bit10: Emergency-Meldung beim Referenzieren 0 = aktiviert (Default) 1 = deaktiviert Bit11: Emergency-Meldung der Software-Endschalter 0 = aktiviert (Default) 1 = deaktiviert Bit12: PDO1 Zusatzinhalt: Sende zusätzlich zum PDO1 Standardinhalt den Zustand der Hardware-Schalteingänge (Tabelle 6, Object 60FDh) mit 8 Byte Gesamtlänge 0 = deaktiviert (Default) 1 = aktiviert
2111h	Positioning parameter	REC				Positionierungsparameter
00h	Grösster Subindex	U8	ro	1Fh		-
01h	Backlash compensation	U8	rw	0h	1, 3	Spielausgleich, um Spindel- oder Getriebespiel auszugleichen. Hierbei kann definiert werden, aus welcher Richtung die Sollposition angefahren werden soll. Wenn nötig, wird dazu die Sollposition um Backlash delta überfahren und anschliessend direkt angefahren. 0h = Sollposition wird direkt angefahren 1h = Sollposition wird von unten angefahren 2h = Sollposition wird von oben angefahren
02h	Backlash delta	U32	rw	3E8h	1, 3	Parameter in benutzerdefinierter Positionseinheit. Gibt an, wie weit eine Sollposition überfahren wird bei eingeschaltetem Spielausgleich.
03h	Positioning timeout	U32	rw	0h	1, 3	Timeout für Fahrauftrag in Millisekunden: Benötigt ein Fahrauftrag mehr Zeit als Positioning timeout, so wird ein QuickStop ausgelöst und in den Zustand ErrorDiagnostic gewechselt. Zusätzlich wird eine EMCY-Meldung mit Fehlercode FF02h ausgegeben. 0h = Timeout deaktiviert >0h = Timeout in Millisekunden (max. $2^{31} - 1$)

Objekt	Name	Format	Zugriff	Default	Save	Beschreibung
0Bh	Time of jogging start speed	U32	rw	3E8h	1, 3	Zeitdauer für Anliegen der Anfangsgeschwindigkeit beim Tippbetrieb in Millisekunden
0Ch	Jogging start speed	U32	rw	0Ah	1, 3	Anfangsgeschwindigkeit beim Tippbetrieb in benutzerdefinierter Geschwindigkeitseinheit
0Dh	Digital input 1	U8	rw	0h	1, 3	Digitaler Hardware-Schalteingang 1 zur Steuerung des Antriebs (vgl. Kontrollwort Objekt 6040h bzw. Tabelle 9). Eingang ist HIGH aktiv, Invertierung durch Addition mit 80h. Parameter / Beschreibung: 00h = Schalteingang deaktiviert 01h = SwitchOn 02h = EnableVoltage 03h = QuickStop 04h = EnableOperation 05h = StartAction 06h = Change set immediately 07h = Relative/absolute 08h = Error reset 09h = Halt 0Ah = Jogging+ (im Positioning Modus nur innerhalb Software-Endschalter, im Homing Modus darüber hinaus) 0Bh = Jogging- (im Positioning Modus nur innerhalb Software-Endschalter, im Homing Modus darüber hinaus) 10h = Wechsle von Ready zu Positioning/Homing/Velocity Ready (Parameter 01h bis 04h zusammengefasst, s.o.) 12h = Referenzieren des Antriebs mit Presetwert (Objekt 2111h, Subindex 11h). Nach Referenzieren wird Parameter wieder auf 00h zurückgesetzt. Referenzschalter anschliessen an Digital input 1. 13h = Jogging+ (mit Warnung auch über Software-Endschalter hinaus unabhängig vom Mode) 14h = Jogging- (mit Warnung auch über Software-Endschalter hinaus unabhängig vom Mode) 15h = Oberer externer Hardware-Endschalter. Endschalter anschliessen an Digital input 1. 16h = Unterer externer Hardware-Endschalter. Endschalter anschliessen an Digital input 1 17h = Bitorientiertes Freischalten des Eingangs zur Ansteuerung von Traversing Block Zeilen. Mit Digital input 1...4 ist eine 4-Bit-Adressierung möglich. Adressierung von Traversing Block Zeile 1 mit 0000, d.h. alle Eingänge inaktiv. Anhalten des Antriebs, wenn Signal an Digital input 1...4 anliegt d.h. alle Eingänge= aktiv. Adressierung wird nur aktiv nach Veränderung. Definition für Zeitverzögerung der Eingänge in Objekt 2111h-12h. ACHTUNG: Bei jedem Einstellwert >0h steuert der Eingang direkt den Antrieb. Der Master auf dem CAN-Bus verliert in diesem Fall teilweise die Kontrolle.
0Eh	Digital input 2	U8	rw	0h	1, 3	Digitaler Hardware-Schalteingang 2. Beschreibung siehe Digital input 1.
0Fh	Digital input 3	U8	rw	0h	1, 3	Digitaler Hardware-Schalteingang 3. Beschreibung siehe Digital input 1.
10h	Digital input 4	U8	rw	0h	1, 3	Digitaler Hardware-Schalteingang 4. Beschreibung siehe Digital input 1.

Objekt	Name	Format	Zugriff	Default	Save	Beschreibung
11h	Reference position	I32	rw	0h	1, 3	Presetwert für Referenzierung des Antriebs. Für Einzelheiten zur Referenzierung, siehe Objekt 6098h Homing-Methoden und Kapitel 4.3. Bei den Homing-Methoden E2h, E1h, D8h, D7h, wird die Referenzposition direkt dem Objekt 6064h Position actual value zugewiesen
12h	Input delay	U32	rw	3E8h	1, 3	Zeitverzögerung in Millisekunden für Ansprechen der digitalen Hardware-Schalteingänge (Objekt 2111h, Subindex 0Dh bis 10h). Nur gültig für Steuerung von Traversing Blocks (Objekteinstellung 17h). Nach dieser Zeitverzögerung wird die nächste Traversing Block Zeile gestartet (3E8h ms = 1'000 ms)
13h	Simulation controlword	U16	rw	0h	1, 3	Kontrollwort, das automatisch gesetzt wird, wenn keine Buskommunikation vorliegt (z.B. für Stand-Alone-Betrieb des Antriebs ohne Feldbus).
14h	Output 1	U32	rw	0h	1, 3	Digitaler Hardware Ausgang 1 zur Ausgabe von Status und Fehler bzw. Warnung (siehe Tabelle 4 und Tabelle 18). Ausgang HIGH aktiv. Invertierung durch Addition von 80h. Bit 0 = Switch Output 1 (siehe Objekt 2111h-18h) Bit 1 = Operation enabled Bit 2 = Drive moving Bit 3 = Target position reached Bit 4 = Software limit switch active Bit 5 = External hardware limit switch active Bit 6 = Warning active Bit 7 = Error active Bit 8 = Fehler Positioning timeout Bit 9 = Fehler Übertemperatur Bit 10 = Fehler Überspannung Bit 11 = Fehler Unterspannung Bit 12 = Fehler Encoder Bit 13 = Warnung Motor blockiert Bit 14 = Ansteuerung externe Haltebremse Bit 15 = Simulation output (siehe Objekt 2111h-1Ch). Freischalten des Ausgangs zur Ansteuerung von externen Geräten. Ausgang LOW aktiv Bit 17...31 entspricht Bit 0...15 in LOW aktiv
15h	Output 2	U32	rw	0h	1, 3	Digitaler Hardware Ausgang 2 (optional). Siehe Beschreibung Output 1
16h	Output 3	U32	rw	0h	1, 3	Digitaler Hardware Ausgang 3 (optional). Siehe Beschreibung Output 1
17h	Output 4	U32	rw	0h	1, 3	Digitaler Hardware Ausgang 4 (optional). Siehe Beschreibung Output 1
18h	Switch output 1 position	I32	rw	0h	1, 3	Position in benutzerdefinierter Positionseinheit für Nockenfunktion von Output 1. Sobald Actual position grösser ist als Switch output 1 position, wird Output 1 aktiviert. Freischalten von Output 1 mit Objekt 2111h-14h, Bit 15.
19h	Switch output 2 position	I32	rw	0h	1, 3	Position in benutzerdefinierter Positionseinheit für Nockenfunktion von Output 2 (optional). Siehe Beschreibung Switch output 1 position
1Ah	Switch output 3 position	I32	rw	0h	1, 3	Position in benutzerdefinierter Positionseinheit für Nockenfunktion von Output 3 (optional). Siehe Beschreibung Switch output 1 position
1Bh	Switch output 4 position	I32	rw	0h	1, 3	Position in benutzerdefinierter Positionseinheit für Nockenfunktion von Output 4 (optional). Siehe Beschreibung Switch output 1 position
1Ch	Simulation output	U8	rw	0h	1, 3	Verwenden von Output 1...4 zur Ansteuerung externer Geräte. Bit 0...3 von Simulation output wird ausgegeben

Objekt	Name	Format	Zugriff	Default	Save	Beschreibung
						auf die freigeschalteten Ausgänge Output 1...4 (siehe Objekt 2111-14h).
1Dh	Service speed	U32	rw	8h	1, 3	Geschwindigkeit in benutzerdefinierter Einheit beim Servicebetrieb ohne Busankopplung
1Eh	Free referencing distance	U32	rw	2710h	1, 3	Weg der beim Referenzieren frei gefahren wird in benutzerdefinierter Positionseinheit (Freifahrweg).
1Fh	Not moving timeout	U32	rw	0h	1, 3	Timeout für Fahrauftrag in Millisekunden: Steht der Antrieb während eines Fahrauftrags länger als Not moving timeout, so wird eine EMCY-Meldung mit Fehlercode FF03h ausgegeben. 0h = Timeout deaktiviert >0h = Timeout in Millisekunden (max. $2^{31} - 1$)
2112h	Statistics	REC	rw			Statistik für Betrieb des Antriebs
00h	Grösster Subindex	U8	ro	5h		-
02h	Number of watchdog events	U16	ro	-		Anzahl der Watchdog-Events des Prozessors
05h	Electronics temperature	U16	ro	-		Aktuelle Temperatur der Leistungselektronik in ℃
2114h	Warning temperature	U16	rw	64h	1, 3	Temperatur-Grenzwert in ℃, ab dem die Übertemperatur-Warnung (Objekt 1003h und 603Fh) gesetzt und eine EMCY-Meldung generiert wird. Der Antrieb kann dennoch weiter betrieben werden (64h = 100℃). Temperatur-Grenzwert für Übertemperatur-Fehler wird werksseitig definiert und kann nicht verändert werden. Bei Temperaturfehler wird der Antrieb sofort gestoppt und in den ErrorDiagnostic Mode versetzt. Das Zurücksetzen (Error reset) erfolgt mit dem Kontrollwort (Objekt 6040h).
2300h	Customer EEPROM	ARR				Speicherbereich für Anwenderdaten
00h	Grösster Subindex	U8	ro	7h		-
01h	Data0	U16	rw	0h	1, 4	-
...	...	...	...	...	...	...
07h	Data6	U16	rw	0h	1, 4	-
2800h	PDO1 add-on	U8	rw	1h	1, 2	0h = PDO1 zyklisch senden 1h = PDO1 nur bei Änderung senden nh = PDO1 nur bei Änderung senden (n mal, n : 2...255)
2801h	PDO2 add-on	U8	rw	1h	1, 2	0h = PDO2 zyklisch senden 1h = PDO2 nur bei Änderung senden nh = PDO2 nur bei Änderung senden (n mal, n : 2...255)
603Fh	Error code	U16	ro	0h		Zuletzt aufgetretener Fehler (Tabelle 18, Objekt 1003h).
6040h	Controlword	U16	rw	0h		Kontrollwort (siehe Tabelle 9) Bit 0 = SwitchOn Bit 1 = EnableVoltage Bit 2 = QuickStop Bit 3 = EnableOperation Bit 4 = StartAction Bit 5 = Change set immediately Bit 6 = Relative/absolute movement Bit 7 = Error reset Bit 8 = Halt Bit 9…10 nicht definiert Bit 11 = Jogging+ Bit 12 = Jogging- Bit 13 = Überfahren des Nullpunkts erlaubt (Rundachse) Bit 14...15 nicht definiert

Objekt	Name	Format	Zugriff	Default	Save	Beschreibung
6041h	Statusword	U16	ro	0h		Statuswort (siehe Tabelle 5) Bit 0 = Ready to switch on Bit 1 = SwitchOn enabled Bit 2 = Operation enabled Bit 3 = Error active Bit 4 = Voltage enabled Bit 5 = QuickStop Bit 6 = SwitchOn disabled Bit 7 = Warning active Bit 8 = CalibrationOK Bit 9 = Not used (permanently 1) Bit 10 = Target position/speed reached Bit 11 = Internal software limit switch active Bit 12 = Drive moving Bit 14 = HomingOK Bit 15 = External hardware limit switch active ACHTUNG: HomingOK und CalibrationOK werden auf Null gesetzt, sobald ein Getriebe- oder Positionsparameter verändert wird (siehe Objekt 6098h).
605Dh	Halt option code	I16	rw	2h	1, 3	Verhalten des Antriebs nach Halt-Befehl (Controlword Bit 8) 0h = Austrudeln des Antriebs, Regler wird deaktiviert 1h = Bremsen mit Profile Deceleration (Objekt 6084h, Regler bleibt anschliessend aktiv) 2h = Bremsen mit QuickStop Rampe (max. Verzögerung, danach Regelung auf Position, an der Halt-Befehl erfolgte)
6060h	Modes of operation	I8	wo	1h		Betriebsmodus des Antriebs (siehe Zustandsdiagramm in Figur 4) 1h = Modus Profile Positioning 3h = Modus Profile Velocity 6h = Modus Homing
6061h	Modes of operation display	I8	ro	1h		Betriebsmodus des Antriebs (siehe Zustandsdiagramm in Figur 4) 1h = Modus Profile Positioning 3h = Modus Profile Velocity 6h = Modus Homing
6063h	Position actual steps	I32	ro	-		Aktuelle Position in Encoder-Schritten
6064h	Position actual value	I32	ro	-		Aktuelle Position in benutzerdefinierter Positionseinheit (siehe Objekt 608Ah). Wert wird berechnet aus aktueller Position in Encoder-Schritten multipliziert mit Position factor. Position actual value = Position actual steps * Position factor ACHTUNG: Wenn Positionseinheit Encoder-Schritte, dann Position factor = 1. Gear ratio oder Feed constant werden in diesem Fall nicht berücksichtigt. Der Antrieb ist mit einem Absolut-Multiturn-Encoder ausgerüstet. Nach dem Wiedereinschalten des Antriebs oder nach einem Ausfall der Betriebsspannung steht die aktuelle Position daher jederzeit sofort zur Verfügung – ohne Referenzierung.
606Ch	Velocity actual value	I32	ro	-		Aktuelle Geschwindigkeit in benutzerdefinierter Geschwindigkeitseinheit (siehe Objekt 608Ch). Wert wird berechnet aus aktueller Motordrehzahl in U/s multipliziert mit dem Velocity factor. Velocity actual value = Motordrehzahl * Velocity factor Für Geschwindigkeitseinheit Encoder-Schritte / Sekunde, Velocity factor = 4. Gear ratio oder Feed constant dann werden nicht berücksichtigt.

Objekt	Name	Format	Zugriff	Default	Save	Beschreibung
6073h	Maximum current	U16	rw	werkseitig	1, 3	Vom Anwender einstellbare Begrenzung für den Motorstrom zur Drehmomentbegrenzung und als Überlastschutz. 00h = Strombegrenzung mit zulässigem Maximalwert xxh = Maximaler Motorstrom in mA Wenn Ansprechen der Strombegrenzung über mehr 500 ms und keine Bewegung des Antriebs, dann EMCY-Meldung 7121h Warnung Motor blockiert. ACHTUNG: Werkseitige Einstellung je nach Motorentyp (siehe Kapitel Technische Daten bzw. Datenblatt). Die werksseitige Einstellung ist der Höchstwert für den Einstellbereich von Maximum current.
607Ah	Target position	I32	rw	0h	1, 3	Sollposition in benutzerdefinierter Positionseinheit (siehe Objekt 608Ah). Bit 6 im Kontrollwort gibt vor, ob Zielposition absolut oder relativ zu aktueller Position ist. Zwischenposition für Spielausgleich wird bei eingeschaltetem Spielausgleich automatisch bestimmt. Wenn aktuelle Position ausserhalb der Software-Endschalter liegt, sendet der Antrieb eine Abort Message (0609003h = Wert ausserhalb des zulässigen Wertebereichs).
607Dh	Software position limit	ARR				Positionen der Software-Endschalter. Diese Positionen können direkt definiert oder über Referenzierung gesetzt werden (siehe Objekt 6098h).
00h	Grösster Subindex	U8	ro	2h		-
01h	Minimum software position limit	I32	rw	0h	1, 3	Position des unteren Software-Endschalters in benutzerdefinierter Positionseinheit (siehe Objekt 608Ah)
02h	Maximum software position limit	I32	rw	0h	1, 3	Position des oberen Software-Endschalters in benutzerdefinierter Positionseinheit (siehe Objekt 608Ah)
607Eh	Polarity	U8	rw	0h	1, 3	Drehrichtung der Motorachse bei steigenden Positionswerten (Blick auf Achse von Getriebeseite her) Position Bit 7 = 0 Drehrichtung im Gegenuhrzeigersinn (CCW) Bit 7 = 1 Drehrichtung im Uhrzeigersinn (CW) Geschwindigkeit (Velocity) Bit 6 = 0 Drehrichtung im Gegenuhrzeigersinn (CCW) Bit 6 = 1 Drehrichtung im Uhrzeigersinn (CW) Bit 0...5 nicht definiert ACHTUNG: Nach Änderung der Drehrichtung müssen Positionen der Software-Endschalter gespiegelt werden. Bei Stirnradgetrieben führt jede Getriebestufe zu einer Umkehrung der Drehrichtung an der Ausgangswelle. Bei Änderung wird automatisch HomingOK und CalibrationOK auf Null gesetzt (siehe Objekt 6098h).
607Fh	Max profile velocity	U32	rw	werksseitig	1, 3	Maximal zulässige Geschwindigkeit in benutzerdefinierter Geschwindigkeitseinheit (siehe Objekt 608Ch). Max profile velocity resultiert aus umgerechneter maximaler Motordrehzahl. Maximaler Wert wird werksseitig eingestellt und kann nicht überschritten werden.
6081h	Profile velocity	U32	rw	1Eh	1, 3	Sollgeschwindigkeit in benutzerdefinierter Geschwindigkeitseinheit (siehe Objekt 608Ch). Für aktuelle Ist-Geschwindigkeit, siehe Objekt 606Ch). ACHTUNG: Sollgeschwindigkeit muss kleiner als maximal zulässige Geschwindigkeit (607Fh) sein. Ansonsten wird eine Abbruchmeldung ausgegeben (siehe SDO Abort Code in Tabelle 19)

Objekt	Name	Format	Zugriff	Default	Save	Beschreibung
6083h	Profile acceleration	U32	rw	64h	1, 3	Sollbeschleunigung in benutzerdefinierter Beschleunigungseinheit (siehe Objekt 608Eh).
6084h	Profile deceleration	U32	rw	64h	1, 3	Sollverzögerung in benutzerdefinierter Beschleunigungseinheit (siehe Objekt 608Eh).
6086h	Motion profile type	I16	rw	0h	1, 3	Beschleunigungsprofil lineare Rampe (Trapezoid Profil), d.h. konstante Beschleunigung und Verzögerung.
6089h	Position notation	I8	rw	FDh	1, 3	Notation der benutzerdefinierten Positionseinheit. FAh = mikro FDh = milli 00h = 1 (none) 03h = kilo ACHTUNG: Nur gültig für Linearbewegung mit Längeneinheit (Position dimension = 01h). Bei Änderung wird automatisch HomingOK und CalibrationOK auf Null gesetzt (siehe Objekt 6098h).
608Ah	Position dimension	U8	rw	01h	1, 3	Dimension der benutzerdefinierten Positionseinheit Längeneinheit für Linearbewegung 01h = Meter Winkeleinheit für Drehbewegung (Getriebeausgang) 00h = Umdrehungen ACh = Encoder-Schritte 10h = Radiant 41h = Grad 42h = Winkelminute 43h = Winkelsekunde Die benutzerdefinierte Positionseinheit ergibt sich durch Multiplikation von Position notation mit Position dimension Millimeter (nur gültig für Linearbewegung). Beispiel: Positionseinheit in Millimeter (FDh und 01h). ACHTUNG: Bei Änderung wird automatisch HomingOK und CalibrationOK auf Null gesetzt (siehe Objekt 6098h).
608Bh	Velocity notation	I8	rw	00h	1, 3	Notation der benutzerdefinierten Geschwindigkeitseinheit. FAh = mikro FDh = milli 00h = 1 (none) 03h = kilo ACHTUNG: Nur gültig für Linearbewegung mit Längeneinheit (Velocity dimension = A6h..A8h).

Objekt	Name	Format	Zugriff	Default	Save	Beschreibung
608Ch	Velocity dimension	U8	rw	A3h	1, 3	Dimension der benutzerdefinierten Geschwindigkeitseinheit Längeneinheit für Linearbewegung: A6h = Meter pro Sekunde A7h = Meter pro Minute A8h = Meter pro Stunde Winkeleinheit für Drehbewegung: 00h = Encoder-Schritte pro Sekunde A3h = Umdrehungen pro Sekunde A4h = Umdrehungen pro Minute A5h = Umdrehungen pro Stunde Die benutzerdefinierte Geschwindigkeitseinheit ergibt sich durch Multiplikation von Velocity notation mit Velocity dimension (nur gültig für Linearbewegung). Beispiel: Geschwindigkeitseinheit in Millimeter / Sekunde (FDh und A6h).
608Dh	Acceleration notation	I8	rw	00h	1, 3	Notation der benutzerdefinierten Beschleunigungseinheit FAh = mikro FDh = milli 00h = 1 (none) 03h = kilo ACHTUNG: Nur gültig für Linearbewegung mit Längeneinheit (Acceleration dimension = A6h...A8h).
608Eh	Acceleration dimension	U8	rw	A6h	1, 3	Dimension der benutzerdefinierten Beschleunigungseinheit Längeneinheit für Linearbewegung: A6h = Meter pro Sekunde^2 A7h = Meter pro Minute^2 A8h = Meter pro Stunde^2 Winkeleinheit für Drehbewegung: 00h = Encoder-Schritte / Sekunde^2 A3h = Umdrehungen pro Sekunde^2 A4h = Umdrehungen pro Minute^2 A5h = Umdrehungen pro Stunde^2 Die benutzerdefinierte Beschleunigungseinheit ergibt sich durch Multiplikation von Acceleration notation mit Acceleration dimension (nur gültig für Linearbewegung). Beispiel: Beschleunigungseinheit in Millimeter / Sekunde^2 (FDh und A6h).
608Fh	Position encoder resolution	ARR				Singleturn-Auflösung des Absolut-Multiturn-Encoders in Schritten pro Motorumdrehung (Zähler). Defaultwert ist 16'384, d.h. 14 Bit. Die Multiturn-Auflösung beträgt 18 Bit bzw. 262'144 Umdrehungen. ACHTUNG: Bei Änderung wird automatisch HomingOK und CalibrationOK auf Null gesetzt (siehe Objekt 6098h).
00h	Grösster Subindex	U8	ro	2h		-
01h	Encoder increments	U32	rw	4000h	1, 3	Anzahl Schritte des Encoders pro Motorumdrehung (Zähler)
02h	Encoder revolutions	U32	rw	1h	1, 3	Anzahl Encoder- bzw. Motorumdrehungen (Nenner)

Objekt	Name	Format	Zugriff	Default	Save	Beschreibung
6091h	Gear ratio	ARR				Getriebeuntersetzung i in Motorumdrehungen pro Umdrehung der Welle am Getriebeausgang. Gear ratio = Motor revolutions / Gear shaft revolutions Darstellung als Bruch, da nur ganzzahlige Werte erlaubt. Beispiel: i = 61,25 : 1 = 6125 / 100. ACHTUNG: Bei Änderung wird automatisch HomingOK und CalibrationOK auf Null gesetzt (siehe Objekt 6098h).
00h	Grösster Subindex	U8	ro	2h		-
01h	Motor revolutions	U32	rw	1h	1, 3	Anzahl Motorumdrehungen (Zähler)
02h	Gear shaft revolutions	U32	rw	1h	1, 3	Anzahl Umdrehung der Welle am Getriebeausgang (Nenner)
6092h	Feed constant	ARR				Berechnungsfaktor zur Umrechnung von Umdrehungen des Motors oder der Getriebeausgangswelle in Bewegung auf Benutzerseite. Feed constant = Feed / Spindle shaft revolutions. Bei Linearbewegungen mit Spindel entspricht Feed constant der Spindelsteigung. Darstellung als Bruch, da nur ganzzahlige Werte erlaubt. Beispiel: Spindelsteigung 2,5 mm / Umdrehung = 1000 mm / 400 Umdrehungen = 1 Meter / 400 Umdrehungen. ACHTUNG: Die Einheit für Weg (Feed) ist immer Meter. Es wird kein Bezug hergestellt zur eingestellten Position notation (6089h) oder Position dimension (608Ah). Bei Änderung wird automatisch HomingOK und CalibrationOK auf Null gesetzt (siehe Objekt 6098h).
00h	Grösster Subindex	U8	ro	2h		-
01h	Feed	U32	rw	1h	1, 3	Weg in Meter (Zähler) ACHTUNG: Die Einheit für Weg (Feed) ist immer Meter
02h	Spindle shaft revolutions	U32	rw	1h	1, 3	Anzahl Umdrehungen (Nenner)
6093h	Position factor	ARR				Berechnungsfaktor zur Umrechnung von Position actual steps (in Encoder-Schritten) in Position actual value (in benutzerdefinierter Positionseinheit, siehe Objekt 608Ah).
00h	Grösster Subindex	U8	ro	2h		-
01h	Numerator	U32	ro	10h		-
02h	Feed constant	U32	ro	1h		-
6094h	Velocity factor	ARR				Berechnungsfaktor zur Umrechnung von Motordrehzahl (in Umdrehungen/s) in Velocity actual value (in benutzerdefinierter Geschwindigkeitseinheit, siehe Objekt 608Ch).
00h	Grösster Subindex	U8	ro	2h		-
01h	Numerator	U32	ro	8h		-
02h	Divisor	U32	ro	1h		-
6098h	Homing method	I8	rw	0h		Methode zur Referenzierung der Antriebs-Position. Nach dem Abschluss der Prozedur wird die Referenzposition gespeichert (Objekt 11h) Für Einzelheiten siehe Kapitel 4.3. Parameter / Beschreibung 00h Keine Referenzierung aktiv F6h Setze aktuelle Position (Objekt 6064h) als unteren Software-Endschalter (Objekt 607Dh-1h) F5h Setze aktuelle Position (Objekt 6064h) als oberen Software-Endschalter (Objekt 607Dh-2h) F4h Referenzieren mit Presetwert: Aktuelle Position (Objekt 6064h) mit überschrieben mit Sollposition

Objekt	Name	Format	Zugriff	Default	Save	Beschreibung
						(Objekt 607Ah). Anschliessend wird im Statuswort (Objekt 6041h) automatisch Bit14 (HomingOK) gesetzt. F3h Einmessmethode (Figur 16): Funktion dient zur Kalibrierung der Position beispielsweise schwer zugänglicher Referenzschalter: Aktuelle Position (Objekt 6064h) wird automatisch überschrieben mit Sollposition (Objekt 607Ah). Zusätzlich wird automatisch Referenzposition (Objekt 2111h-11h) um Einmess-Offset korrigiert. Anschliessend wird im Statuswort (Objekt 6041h) automatisch zusätzlich Bit8 (CalibrationOK) gesetzt. ACHTUNG: Nur zulässig, wenn vorher im Statuswort (Objekt 6041h) bereits Bit14 (HomingOK) gesetzt ist. ECh Fahr zu unterem Software-Endschalter EBh Fahr zu oberem Software-Endschalter EAh Fahr zu Position Null E2h Referenzieren auf unteren Referenzschalter (Figur 14): Fahr zu unterem Referenzschalter. Sobald unterer Referenzschalter anspricht, wird automatisch aktuelle Position (Objekt 6064h) überschrieben mit gespeicherter Referenzposition (Objekt 2111h-11h). Anschliessend wird im Statuswort (Objekt 6041h) automatisch Bit14 (HomingOK) gesetzt. ACHTUNG: Nur möglich wenn Input 1 oder Input 2 (Objekt 0Dh oder 0Eh) eingestellt ist auf Referenzpunkt setzen (12h). E1h Referenzieren auf oberen Referenzschalter (Figur 14): Beschreibung analog Beschreibung E2h Referenzieren auf unteren Referenzschalter. D8h Referenzieren auf unteren Anschlag (Figur 15): Fahr zu unterem Anschlag. Sobald Antrieb blockiert, wird automatisch aktuelle Position (Objekt 6064h) überschrieben mit gespeicherter Referenzposition (Objekt 2111h-11h). Anschliessend wird im Statuswort (Objekt 6041h) automatisch Bit14 (HomingOK) gesetzt. D7h Referenzieren auf oberen Anschlag (Figur 15) Beschreibung analog Beschreibung D8h Referenzieren auf unteren Anschlag. Nach der Referenzierung wird das Objekt Homing method wieder automatisch auf 0h zurückgesetzt. ACHTUNG: HomingOK und CalibrationOK werden automatisch auf Null gesetzt bei Schalten in den Modus Homing oder bei Veränderung eines der Parameter zur Positionsberechnung: Polarity 607Eh, Position notation 6089h, Position dimension 608Ah, Position encoder resolution 608Fh, Gear ratio 6091h, Feed constant 6092h (siehe Kapitel 4.3.5).
6099h	Homing speed	ARR				Geschwindigkeit für Referenzierung
00h	Grösster Subindex	U8	ro	2h		-
01h	Speed during search for switch	U32	rw	0Ah	1, 3	Geschwindigkeit in benutzerdefinierter Einheit beim Referenzieren auf Anschlag oder beim Fahren auf Endschalter.
02h	Speed during search for zero position	U32	rw	0Ah	1, 3	Geschwindigkeit in benutzerdefinierter Einheit beim Fahren zur Position Null.
609Ah	Homing acceleration	U32	rw	64h	1, 3	Beschleunigung in benutzerdefinierter Beschleunigungseinheit für alle Homing Methoden (Beschleunigung und Verzögerung)
60C0h	Interpolation sub mode select	I16	ro	0h	1, 3	Linearer Interpolationsalgorithmus für Positionierung mit Traversing Blocks

Objekt	Name	Format	Zugriff	Default	Save	Beschreibung
60C1h	Traversing block array	ARR				Zweidimensionales Traversing Block Array (Fahrdatensatz) mit 20 Traversing Block Zeilen und 7 Unterparametern pro Zeile. Für Einzelheiten, siehe Tabelle 11a und 11b. Zur Verwendung des Traversing Blocks, Antrieb betreiben im Positioning bzw. Velocity Modus (siehe Fig. 5). Start und Stop erfolgt über Kontrollworte bzw. Hardware-Eingänge.
00h	Grösster Subindex	U8	ro	7h		-
01h	Target position	I32	rw	0h		Sollposition in benutzerdefinierter Positionseinheit
02h	Time	I32	rw	0h		Timer für Ablaufsteuerung, Zeit in Millisekunden. (Timereigenschaften definiert in Objekt 60C1-03h Regulator mode) Programmierung von Schleifen: Schleifenanfang Time = -1h, Regulator mode = 20h Schleifenende Time = -n, Regulator mode = 20h. n= Anzahl der Schleifendurchgänge, d.h. Time = -1... -32'768 (FFFF'FFFFh bis FFFF'8000h). Endlosschleife bei Time = -32'768. Zwischen Schleifenanfang und Schleifenende beliebige Fahrdaten (Beispiel in Tabelle 11b).
03h	Regulator mode	I32	rw	0h		Einstellungen für Regler (Timer definiert in Objekt 60C1-02h) Gehe sofort zu nächsten Traversing Block Zeile nach Beenden der aktuellen Zeile. Timer startet nach Erreichen der Sollposition / Sollgeschwindigkeit 0h = Geschwindigkeitsregler 1h = Positionsregler absolut 2h = Positionsregler relativ Gehe sofort zu nächsten Traversing Block Zeile nach Beenden der aktuellen Zeile. Timer startet bei Start des Fahrauftrags 10h = Geschwindigkeitsregler 11h = Positionsregler absolut 12h = Positionsregler relativ Stop nach Beenden der aktuellen Traversing Block Zeile. Timer startet nach Erreichen der Sollposition / Sollgeschwindigkeit 20h = Geschwindigkeitsregler 21h = Positionsregler absolut 22h = Positionsregler relativ Stop nach Beenden der aktuellen Traversing Block Zeile. Timer startet mit Starten des Fahrauftrags 30h = Geschwindigkeitsregler 31h = Positionsregler absolut 32h = Positionsregler relativ
04h	Target velocity	I32	rw	0h		Sollgeschwindigkeit in benutzerdefinierter Geschwindigkeitseinheit
05h	Acceleration	I32	rw	0h		Sollbeschleunigung in benutzerdefinierter Beschleunigungseinheit
06h	Deceleration	I32	rw	0h		Sollverzögerung in benutzerdefinierter Beschleunigungseinheit
07h	Max current	I32	rw	0h		Maximaler Strom in mA
60C4h	Traversing block configuration	REC				Parameter für Zugriff auf Traversing block array (siehe Objekt 60C1h)
00h	Grösster Subindex	U8	ro	6h		-
01h	Maximum buffer size	U32	ro	14h		Maximal 20 Traversing Block Zeilen können gespeichert werden.
02h	Actual buffer size	U32	rw	0h		Aktuelle Anzahl verwendeter Traversing Block Zeilen

Objekt	Name	Format	Zugriff	Default	Save	Beschreibung
03h	Buffer organization	U8	ro	0h		FIFO Puffer-Organisation
04h	Buffer position	U16	rw	0h		Zeiger zu aktueller Traversing Block Zeile
05h	Size of data record	U8	wo	7h		Grösse des Datenrecords, entspricht einer Traversing Block Zeile
06h	Clear input buffer	U8	wo	1h		Funktion zum Löschen des Puffers 0 = Clear input buffer deaktiviert 1 = Clear input buffer aktiviert
60FDh	Digital input monitor	U32	ro	-		Monitor für externe Hardware-Schalteingänge (Objekt 2111h Subindex 0Dh...10h) Bit 0..15 nicht unterstützt Bit 16 Monitorbit für Digital input 1 Bit 17 Monitorbit für Digital input 2 Bit 18 Monitorbit für Digital input 3 Bit 19 Monitorbit für Digital input 4 Bit 20..31 nicht unterstützt
60FEh	Brake output	ARR				
00h	Grösster Subindex	U8	ro	1h		-
01h	Brake output	U32	rw	1h	1, 3	Parameter für Haltebremse, Ansteuerung der Haltebremse erfolgt automatisch Bit 0 = 0 Haltebremse deaktiviert Bit 0 = 1 Haltebremse aktiviert Bit 1..15 = nicht unterstützt Bit 16..31 = Zeitverzögerung in Millisekunden nach Einschalten des Positions- bzw. Geschwindigkeitsreglers bis zum Öffnen der Haltebremse (0 ms..65'535 ms)
6402h	Motor type	U16	rw	Ah		Motortyp (Ah = sinuskommutierter pulsmodulierter bürstenloser DC Motor)
6410h	Motor data	ARR				Motorparameter
00h	Grösster Subindex	U8	ro	2h		-
01h	Motor operating time	U32	ro		1, 4	Betriebszeit des Motors bzw. Reglers in Sekunden. Wert wird alle 6 Minuten gespeichert.
02h	Software version controller	U32	ro	werkseitig		Version der Firmware des Reglerprozessors (Beispiel: Version 1.2.3 = 123h)
6502h	Drive modes	U32	ro	23h		Unterstützte Betriebsarten des Antriebs zur Information (mehrere Bits können gesetzt sein). Bit 0 = Modus Profile Positioning Bit 2 = Modus Profile Velocity Bit 5 = Modus Homing
6510h	Drive data	ARR				Antriebsparameter
00h	Grösster Subindex	U8	ro	4		-
01h	Drive operating time	U32	ro	-	1, 4	Betriebszeit des Antriebs in Sekunden. Wert wird alle 6 Minuten gespeichert.
03h	Number of positioning tasks	U32	ro	-	1, 4	Anzahl der Positionieraufträge. Wert wird alle 6 Minuten gespeichert.
04h	Number of homing tasks	U32	ro	-	1, 4	Anzahl der durchgeführten Referenzierungen und Endschalter-Definitionen. Wert wird alle 6 Minuten gespeichert.
05h	Number of saving tasks	U32	ro	-	1, 4	Anzahl der Speichervorgänge ins EEPROM. Wert wird alle 6 Minuten gespeichert.

Objekt	Name	Format	Zugriff	Default	Save	Beschreibung
06h	Error temperature	U16	ro	80h		Temperatur-Grenzwert in ℃ für Ansprechen des Übertemperatur-Fehlers (80h = 128℃). Überschreitet die Electronics temperature (Objekt 2112h-05h) diesen Grenzwert, wird der Antrieb gestoppt und eine EMCY-Meldung generiert (Objekt 1003h und 603Fh). Anschliessend wechselt der Antrieb in den Zustand Error Diagnostic (Error code 4210h-0000h). Zurücksetzen des Temperatur-Fehlers (Error reset) erfolgt mit Kontrollwort (Objekt 6040h). Der Temperatur-Grenzwert für den Übertemperatur-Fehler ist werkseitig definiert und kann nicht verändert werden.
07h	Max motor current	U16	ro	werkseitig		Maximaler Motorstrom in mA
08h	Min position	S32	ro	-		Minimal zulässige Positionsangabe in benutzerdefinierter Positionseinheit (siehe Objekt 608Ah)
09h	Max position	S32	ro	-		Maximal zulässige Positionsangabe in benutzerdefinierter Positionseinheit (siehe Objekt 608Ah)
0Ah	Max velocity	U32	ro	-		Maximal zulässige Geschwindigkeitsangabe in benutzerdefinierter Geschwindigkeitseinheit (siehe Objekt 608Ch).
0Bh	Max acceleration deceleration	U32	ro	-		Maximal zulässige Beschleunigung in benutzerdefinierter Beschleunigungseinheit (siehe Objekt 608Eh).

Tabelle 11a: Traversing Block Array (Fahrdatensatz) mit 20 Traversing Block Zeilen (Objekt 60C1h). Zeilenauswahl in Objekt 60C4h-4h. Parameter in Objekt 60C1h, Subindex 1h...7h. Werte in Tabelle sind Defaultwerte.

Traversing Block		Target position	Time	Regulator mode	Target velocity	Accelera-tion	Decelera-tion	Max current
60C4h-4h	Objekt 60C1h, Subindex	1h	2h	3h	4h	5h	6h	7h
1h	Traversing Block Zeile 1	0	0	20h	0	0	0	0
2h	Traversing Block Zeile 2	0	0	20h	0	0	0	0
3h	Traversing Block Zeile 3	0	0	20h	0	0	0	0
...								
14h	Traversing Block Zeile 20	0	0	20h	0	0	0	0

Tabelle 11b: Traversing Block Array; Beispiel mit Endlosschleife.

Traversing Block		Target position	Time	Regulator mode	Target velocity	Accelera-tion	Decelera-tion	Max current
60C4h-4h	Objekt 60C1h, Subindex	1h	2h	3h	4h	5h	6h	7h
1h	Traversing Block Zeile 1	0	FFFFFFFFh	20h	0	0	0	0
2h	Traversing Block Zeile 2	2000h	500h	2h	500h	64h	64h	2710h
3h	Traversing Block Zeile 3	0	FFFF8000h	20h	0	0	0	0
...								
14h	Traversing Block Zeile 20	0	0	20h	0	0	0	0

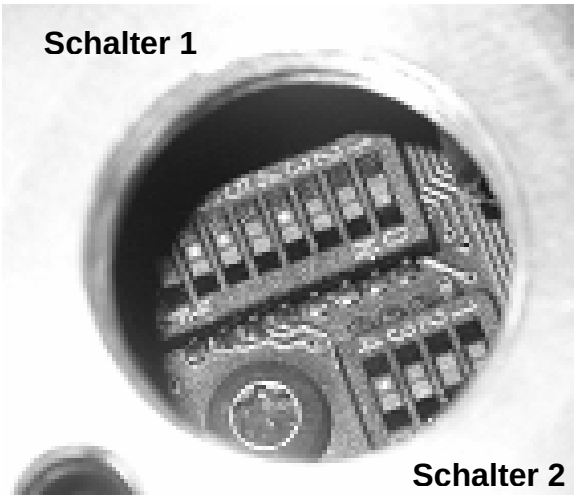
4 Inbetriebnahme

4.1 Node-ID und Baudrate einstellen

Wenn der Antrieb an den Bus angeschlossen wird, meldet er sich nach dem Einschalten mit einer BootUp-Meldung auf dem Bus an. Nun muss der Antrieb an seine Umgebung angepasst und konfiguriert werden.

Node-ID und Baudrate einstellen über Einstellschalter

Node-ID und Baudrate können über Schalter eingestellt werden. Hierbei ist die Einstellung der Einstellschalter dominant für alle anderen Einstellmöglichkeiten. Einstellschalter sind werksseitig deaktiviert und Node-ID sowie Baudrate sind definiert durch Parameter in Objekt 2100h und 2101h bzw. aus dem EEPROM.



Figur 12: Rückansicht des Antriebs mit Schalter 1 zur Einstellung der Node-ID und mit Schalter 2 zur Einstellung der Baudrate.

Tabelle 12: Einstellung von Node-ID mit Schalter 1 (Zuordnung, siehe Figur 12). Werkseinstellung ist 00h. Node-ID und Baudrate sind in diesem Fall definiert durch Objekt 2100h und 2101h. Gleichzeitig ist Schalter 2 (Tabelle 13) deaktiviert (OFF = -, kein Einfluss = x).

Stellung S1	Stellung S2	Stellung S3	Stellung S4	Stellung S5	Stellung S6	Stellung S7	Stellung S8	Node-ID
$2^0 = 1$	$2^1 = 2$	$2^2 = 4$	$2^3 = 8$	$2^4 = 16$	$2^5 = 32$	$2^6 = 64$		
-	-	-	-	-	-	-	x	Werkseinstellung
ON	-	-	-	-	-	-	x	1
-	ON	-	-	-	-	-	x	2
ON	ON	-	-	-	-	-	x	3
-	-	ON	-	-	-	-	x	4
ON	-	ON	-	-	-	-	x	5
-	ON	ON	-	-	-	-	x	6
ON	ON	ON	-	-	-	-	x	7
-	-	-	ON	-	-	-	x	8
ON	-	-	ON	-	-	-	x	9
-	ON	-	ON	-	-	-	x	10
ON	ON	-	ON	-	-	-	x	11
-	-	ON	ON	-	-	-	x	12
ON	-	ON	ON	-	-	-	x	13

...							x	...
ON	ON	ON	ON	ON	ON	ON	x	127h

Tabelle 13: Einstellung von Baudrate mit Schalter 2 (Zuordnung, siehe Figur 12). Werkseinstellung ist S1...S4=OFF (OFF = -, kein Einfluss = x).

Stellung S1	Stellung S2	Stellung S3	Stellung S4	Baudrate [kBit/s]
-	-	-	x	10
ON	-	-	x	20
-	ON	-	x	50
ON	ON	-	x	125
-	-	ON	x	250
ON	-	ON	x	500
-	ON	ON	x	800
ON	ON	ON	x	1000

Beim nächsten Initialisieren meldet sich der Antrieb mit neuer Node-ID und Baudrate auf dem Bus an. ACHTUNG: Auch die Baudrate des Masters muss nun geändert werden, um den Antrieb anzusprechen.

Node-ID und Baudrate einstellen im EEPROM

Die Einstellung der Node-ID im EEPROM erfolgt über das Objekt 2101h. Dies ist nur möglich, wenn alle Einstellschalter sich in Werkseinstellung befinden.

Beispiel Node-ID auf 23h einstellen:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
600h+Node-ID	8	2Fh	01h	21h	0h	23h	Xx	Xx	Xx

Bestätigung:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	60h	01h	21h	0h	0h	0h	0h	0h

Tabelle 14: Einstellung der Baudrate im EEPROM über Objekt 2100h. Hierbei wird ein Index ins Objekt geschrieben, nicht die effektive Baudrate (nur möglich, wenn Einstellschalter in Werkseinstellung 00h). ACHTUNG: Die Indices für Objekt 2100h und LSS unterscheiden sich.

Baudrate [kBit/s]	10	20	50	100	125	250	500	800	1000
Index (Definition in Objekt 2100h)	0h	1h	2h	3h	4h	5h	6h	7h	8h
Index (LSS) nach CiA-Tabelle	8h	7h	6h	5h	4h	3h	2h	1h	0h

Beispiel: Einstellen der Baudrate auf 250 kBit/s (5h):

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
600h+Node-ID	8	2Fh	00h	21h	0h	05h	Xx	Xx	Xx

Bestätigung:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	60h	00h	21h	0h	0h	0h	0h	0h

Nun müssen die neuen Einstellungen gespeichert werden. Über das Objekt 1010h wird das Speichern untenstehender Objekte in den nichtflüchtigen Speicher (EEPROM) ausgelöst. Um ein unabsichtliches Speichern zu verhindern muss dazu die Botschaft "save“ in den Subindex 1 geschrieben werden.

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
600h+Node-ID	8	23h	10h	10h	01h	73 's’	61 'a’	76 'v’	65 'e’

Bestätigung:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	60h	10h	10h	01h	0h	0h	0h	0h

Beim nächsten Initialisieren meldet sich der Antrieb mit neuer Node-ID und Baudrate auf dem Bus an. ACHTUNG: Auch die Baudrate des Masters muss nun geändert werden, um den Antrieb anzusprechen.

Node-ID und Baudrate einstellen mit LSS

Wenn mehrere Teilnehmer am selben Bus die gleiche Node-ID und Baudrate besitzen, können diese zwei Objekte mit LSS umgestellt werden. Mit LSS wird der Antrieb über Produktcode, Revisionsnummer, Vendor-ID und Seriennummer angesprochen und konfiguriert (siehe 2.2.9).

Beim nächsten Initialisieren meldet sich der Antrieb mit neuer Node-ID und Baudrate auf dem Bus an. ACHTUNG: Auch die Baudrate des Masters muss nun geändert werden, um den Antrieb anzusprechen.

4.2 Objekte schreiben und lesen

Um ein Objekt (SDO) zu schreiben oder zu lesen werden immer zwei Telegramme gesendet.

Objekt setzen

Zuerst sendet der Master den zu setzenden Wert. Anschliessend sendet der Antrieb die Bestätigung.

Wert (wxyz) wird gesendet:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
600h+Node-ID	8	23h	7Dh	60h	1h	w	x	y	z

Bestätigung:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	60h	7Dh	60h	1h	0	0	0	0

Objekt lesen

Zuerst sendet der Master eine Aufforderung des gewünschten Objekts. Dann sendet der Antrieb den geforderten Wert.

Anfrage vom Master:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
600h+Node-ID	8	40h	41h	60h	0	x	x	x	x

Antwort (abcd) des Antriebs auf die Anfrage:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	43h	41h	60h	0	a	b	c	d

Parameter speichern ins EEPROM

Mit dem Objekt 1010h können die in den Objekten enthaltenen Parameter nichtflüchtig in ein EEPROM gespeichert werden. Um unbeabsichtigtes Speichern zu verhindern, muss dabei der Befehl save in Objekt

1010h, Subindex 1, Byte 5 bis 8 geschrieben werden. Welches der Objekte mit welchem Speicherkommando gespeichert wird, ist spezifiziert in Tabelle 10, Spalte Save und Beschreibung zu Objekt1010h.

COB-ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
600h+Node-ID	8	23	10	10	01	73 (= s)	61 (= a)	76 (= v)	65 (= e)

Default Parameter laden aus dem EEPROM

Mit dem Objekt 1011h könne die Default-Werte der Objekt-Parameter aus dem EEPROM geladen werden.

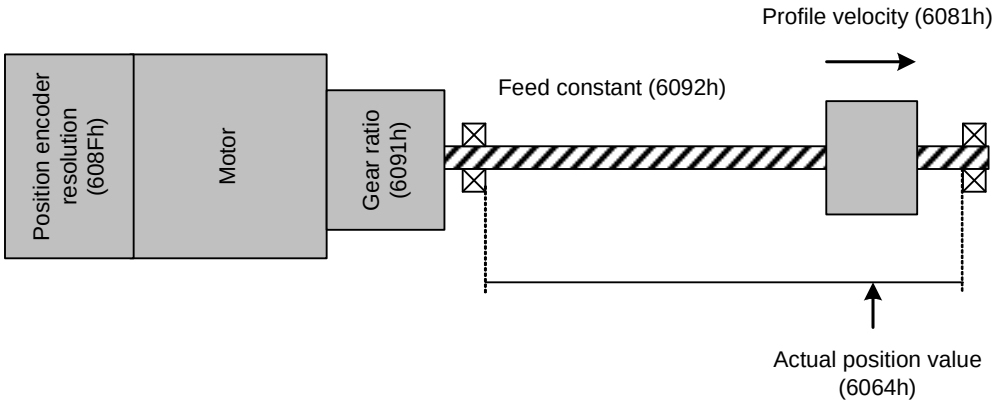
Um unbeabsichtigtes Laden zu verhindern, muss dabei der Befehl load in Objekt 1011h, Subindex 1, Byte 5 bis 8 geschrieben werden. Welches der Objekte mit welchem Ladekommando geladen wird, ist spezifiziert in Tabelle 10, Spalte Save und Beschreibung zu Objekt1011h.

ACHTUNG: Die aktuellen Werte im RAM werden bei Laden überschrieben.

COB-ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
600h+Node-ID	8	23	11	10	01	6C (= l)	6F (= o)	61 (= a)	64 (= d)

4.3 Antriebsparameter konfigurieren

4.3.1 Position, Geschwindigkeit und Getriebe



Figur 13: Konfigurierung des Antriebs.

Benutzerdefinierte Positionseinheit (Position notation und Position dimension)

Die benutzerdefinierte Positionseinheit setzt sich zusammen aus Position notation (Objekt 6089h) und Position dimension (Objekt 608Ah). Position notation entspricht einem Vorfaktor, d.h. für Linearbewegungen mikro (10^{-6}), milli (10^{-3}), 1 (10^{-0}) oder kilo (10^{3}). Für Drehbewegungen ist nur Position notation = 1 sinnvoll. Die Position dimension entspricht der eigentlichen Positionseinheit, d.h. für Linearbewegungen Meter und für Drehbewegungen Encoder-Schritte, Grad, Minuten, Sekunden und Radiant. Die benutzerdefinierte Positionseinheit ergibt sich als Multiplikation:

$$\text{Benutzerdefinierte Positionseinheit} = \text{Position notation} * \text{Position dimension}$$

$$\text{Beispiel : } [\text{Position}] = \text{milli} * \text{meter} = \text{Millimeter}$$

Nach Definition der benutzerdefinierten Positionseinheit beziehen sich alle Positionsangaben zur Ansteuerung des Antriebs (ausser Objekt 6063h) nur noch auf diese Einheit und werden intern automatisch umgerechnet unter Berücksichtigung von Getriebeuntersetzung (Gear ratio) und der Spindel bzw. Kundengetriebe (Feed constant).

Dieses Prinzip gilt analog für die benutzerdefinierte Geschwindigkeitseinheit und Beschleunigungseinheit.

Position encoder resolution (Objekt 608Fh)

Dieses Objekt ist in 2 Subindexe unterteilt. Subindex 1 ist die Anzahl der Encoder-Schritte. Subindex 2 beschreibt beinhaltet die Anzahl Umdrehungen. Der MSIA Encoder hat 16’384@Schritte pro Motor-Umdrehung, also ist Position encoder resolution = 16’384.

$$\text{Position encoder resolution} = \frac{\text{Encoder increments}}{\text{Motor revolutions}}$$

Gear ratio (Objekt 6091h)

In diesem Objekt wird in Subindex 1 und 2 das Getriebe-Untersetzungsverhältnis dargestellt. Wenn zum Beispiel das Untersetzungsverhältnis den Wert 61.25 hat, wird in Subindex 1 der Wert 6125 und in Subindex 2 der Wert 100 eingetragen. 6125/100 = 61.25

$$\text{Gear ratio} = \frac{\text{Motor revolutions}}{\text{Gear shaft revolutions}}$$

Feed constant (Objekt 6092h)

Dieses Objekt definiert in Subindex 1 und 2 das Kundengetriebe. Zum Beispiel 400 Umdrehungen der Spindel ergeben einen Meter Weg (Linearumsetzung).
ACHTUNG: Die Wegeinheit für Feed ist immer Meter. Aus Vereinfachungsgründen wird kein Bezug zur eingestellten Position notation (6089h) oder Position dimension (608Ah) hergestellt.

$$\text{Feed constant} = \frac{\text{Feed}}{\text{Spindle shaft revolutions}}$$

Position factor (Objekt 6093h)

$$\text{Position factor} = \text{Position notation} \cdot \text{Position dimension} \cdot \text{Encoder resolution} \frac{\text{Gear ratio}}{\text{Feed constant}}$$

16’384@

$$\text{Beispiel}: \text{Position factor} = \frac{1}{1} \cdot \frac{1}{1000} \cdot \frac{\frac{4}{1} \cdot \frac{6125}{100}}{\frac{1}{400}} = 98$$

Position actual value (Objekt 6064h)

$$\text{Position actual value} = \frac{\text{Position actual steps}}{\text{Position factor}}$$

16’384@

$$100\,\text{mm (anlagenseitig)} \Leftrightarrow 9800\,\text{Encoder - Schritte (motorseitig)}$$

ACHTUNG: Falls als Positionseinheit Encoder-Schritte eingestellt ist, dann gilt automatisch Position factor =1. Gear ratio oder Feed constant werden dann nicht in der Umrechnung mitberücksichtigt.

Benutzerdefinierte Geschwindigkeitseinheit (Velocity notation and Velocity dimension)

Die benutzerdefinierte Geschwindigkeitseinheit setzt sich zusammen aus Velocity notation (Objekt 608Bh) und Velocity dimension (Objekt 608Ch). Velocity notation entspricht einem Vorfaktor, d.h. für Linearbewegungen mikro (10^{-6}), milli (10^{-3}), 1 (10^{-0}) oder kilo (10^{3}). Für Drehbewegungen ist nur Position notation = 1 sinnvoll.
Die Velocity dimension entspricht der eigentlichen Geschwindigkeitseinheit, d.h. für Linearbewegungen beispielsweise Meter pro Sekunde und für Drehbewegungen Encoder-Schritte pro Sekunde oder Grad pro Sekunde. Die benutzerdefinierte Positionseinheit ergibt sich als Multiplikation:

$$\text{Benutzerdefinierte Geschwindigkeitseinheit} = \text{Velocity notation} * \text{Velocity dimension}$$

$$\text{Beispiel}: [\text{Geschwindigkeit}] = \text{milli} * \text{meter/s} = \text{Millimeter/s}$$

Nach Definition der benutzerdefinierten Geschwindigkeitseinheit beziehen sich alle Geschwindigkeitsangaben zur Ansteuerung des Antriebs nur noch auf diese Einheit und werden intern automatisch umgerechnet unter Berücksichtigung von Getriebeuntersetzung (Gear ratio) und der Spindel bzw. Kundengetriebe (Feed constant).

Velocity factor (Objekt 6094h)

$$\text{Velocity factor} = \text{Velocity notation} \cdot \text{Velocity dimension} \cdot \text{Encoder resolution} \cdot \frac{\text{Gear ratio}}{\text{Feed constant}}$$

@16’384

$$\text{Beispiel : Velocity factor} = \frac{1}{60} \cdot \frac{1}{1000} \cdot \frac{\frac{4}{1} \cdot \frac{6125}{100}}{\frac{1}{400}} = 0.408$$

Velocity actual value (Objekt 606Ch) und Profile velocity (Objekt 6081h)

$$\text{Velocity actual value} = \frac{\text{Motordrehzahl in U/s}}{\text{Velocity factor}} \qquad \text{Profile velocity} = \frac{\text{Motordrehzahl in U/s}}{\text{Velocity factor}}$$

$$73\,\text{mm/min (anlagenseitig)} \Leftrightarrow 30\,\text{U/s (motorseitig)}$$

ACHTUNG: Falls als Geschwindigkeitseinheit Encoder-Schritte pro Sekunde eingestellt ist, dann gilt automatisch Velocity factor = 1/4. Gear ratio oder Feed constant werden dann in der Umrechnung nicht mit berücksichtigt.

4.3.2 Drehrichtung (Polarity)

Im Objekt 607Eh kann die Drehrichtung definiert werden.

4.3.3 Strom und Geschwindigkeit

Die Drehmoment-Begrenzung des Antriebs erfolgt über die Begrenzung des maximalen Motorstroms (siehe Objekt 6073h). Der Wert 0 entspricht der maximal zulässigen Strombegrenzung, Werte zwischen 1 mA und 20’000 mA sorgen für eine (Drehmoment-) Begrenzung.
Die Geschwindigkeiten werden wie die Positionen in benutzerdefinierten Einheiten angegeben. Somit wird die Dimension und Notation (Objekte 608Bh, 608Ch) der Geschwindigkeitsangabe als auch die Getriebe- und Encoder-Einstellungen (Objekte 608Fh, 6091h, 6092h) verrechnet.

4.3.4 Positioning Timeout

Falls dieser Wert nicht 0 ist, wird ein Fahrauftrag abgebrochen, wenn er länger als die angegebene Zeit in ms benötigt.

4.3.5 Spielausgleich

Im Objekt 2111h Subindex 1 kann angegeben werden, wie das Ziel angefahren werden soll. Entweder wird das Ziel direkt, von oben oder von unten angefahren. Zusätzlich wird in Subindex 2 definiert, wie weit der Antrieb zum Spielausgleich über die Sollposition hinausfahren soll.
Beispiel: Der Antrieb befindet sich auf aktueller Position 1’000. Die Sollposition ist 2’000. Wenn nun „Sollposition direkt anfahren“ aktiviert ist, dann wird die Sollposition von unten her angefahren. Ist dagegen „Sollposition von oben anfahren“ aktiviert, dann fährt der Antrieb zuerst auf Zwischenposition 2’100 (bei Backlash delta 100) und dann von oben nach unten auf Sollposition 2’000.

Tabelle 15: Beispiele für Antriebsparameter.

Parameter	Objekt	Subindex	Wert	Beschreibung
Position notation	6089h	0	FDh	milli (*10^{-3})
Position dimension	608Ah	0	01h	Meter (Längsbewegung)
Velocity notation	608Bh	0	FDh	milli (*10^{-3})
Velocity dimension	608Ch	0	A6h	Meter / min
Encoder increments	608Fh	1	4'000h	16'384 Encoder-Schritte pro Umdrehung
Motor revolutions	608Fh	2	01h	
Gear ratio – motor revolutions	6091h	1	17EDh	61,25 = 6125 / 100
Gear ratio – shaft revolutions	6091h	2	64h	
Feed constant - feed	6092h	1	01h	2,5 mm / Umdrehung
Feed constant – shaft revolutions	6092h	2	190h	= 1000 mm / 400 Umdrehungen = 1 Meter / 400 Umdrehungen

4.3.6 Einstellungen speichern

Über das Objekt 1010h wird das Speichern in den nicht flüchtigen Speicher (EEPROM) ausgelöst. Um ein unabsichtliches Speichern zu verhindern muss die Botschaft „save“ in den Subindex 1 geschrieben werden.

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
600h+Node-ID	8	23h	10h	10h	1h	73 ‚s’	61 ‚a’	76 ‚v’	65 ‚e’

Antwort:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	60h	10h	10h	1h	0	0	0	0

4.4 Antriebsposition referenzieren (Homing)

Der Antrieb benötigt einen Bezugspunkt zur Mechanik in der Anlage oder Applikation. Hierzu muss der Antrieb im Normalfall einmalig referenziert werden. Nach dem Einschalten befindet sich der Antrieb im Ready Mode. Das Referenzieren erfolgt im Modus Homing mit dem Objekt 6098h. Der Wechsel des Zustands erfolgt durch:

1. Objekt 6060h (Modes of operation) auf 6=Homing setzen
2. Objekt 6040h (Kontrollwort) auf 000Fh setzen
   SwitchOn, EnableVoltage, QuickStop und EnableOperation = 1

Anlagen-Bezugspunkt anfahren

Es gibt verschiedene Möglichkeiten zum Setzen der Referenzposition:

1. Herantasten an Bezugspunkt über Jogging, d.h. Ansteuerung der digitalen Hardware-Schalteingänge
   a. Externe Anschlüsse im Objekt 2111h, Subindex2 als Schalter für Jogging+/- definieren.
2. Herantasten an Bezugspunkt durch Vorgabe von Zielpositionen über den CAN-Bus (relativ / absolut)
   a. Sollposition vorgeben in benutzerdefinierter Einheit (siehe Objekt 607Ah)
   b. Relativ oder absolut in Objekt 6040h, Bit6 setzen
   c. Fahren mit Objekt 6040h, Bit4 (HomingStart)
3. Fahren auf Anschlag (Anschlag als Bezugspunkt)
   a. Strombegrenzung mit Objekt 6073h auf Minimum setzen
   b. Mit Objekt 6098h, Wert –40, -41 hin- und herfahren
   c. ACHTUNG: Durch die Getriebeuntersetzung können sehr grosse Kräfte wirken.
4. Fahren bis externer Referenzschalter anspricht (Referenzposition als Bezugspunkt)
   a. Externen Referenzschalter an Referenzposition anbringen
   b. Externen Hardware-Schalteingang im Objekt 2111h, Subindex2 als Hardware-Endschalter definieren
   c. Mit Objekt 6098h, Wert –30, -31 hin- und herfahren
   d. Nachdem die Referenzposition gefunden und gespeichert ist, kann der externe Referenzschalter demontiert werden

Referenzposition direkt setzen

ACHTUNG: Das direkte Setzen der Referenzposition kann gefährlich sein. Bei Eingabe einer inkorrekten Referenzposition ist anschliessend die Funktion der Endschalter wirkungslos. Der Antrieb kann dann gegen einen Anschlag fahren.
Aus diesem Grunde wurde von der CiA ein spezielles Vorgehen definiert. Hierbei muss zunächst die Sollposition auf die zu setzende Referenzposition geändert werden. Danach wird die aktuelle Position des Antriebs mit Objekt 6098, Wert –12 mit der Sollposition (Target position) überschrieben.
z.B. setze Position des Antriebs an der aktuellen Stelle auf 100 mm.

1. Setze Antrieb in den Zustand HomingReady
2. Setze Objekt 607Ah (Target position) auf 100 (Beispiel: Antrieb arbeitet in mm)
3. Setze nächste Referenzierung mit Objekt 6098 (Homing method) auf –12
4. Starte Referenzierung mit Objekt 6040h (Kontrollwort) = 001Fh

Tabelle 16: Übersicht über Referenzierungs-Methoden (Objekt 6098h) im HomingReady Mode.

Objekt 6098h		Beschreibung
0	0h	Keine Referenzierung aktiv
-10	F6h	Setze aktuelle Position als unteren Software-Endschalter
-11	F5h	Setze aktuelle Position als oberen Software-Endschalter

-12	F4h	Referenzieren mit Presetwert
-13	F3h	Referenzieren mit Einmessmethode
-20	ECh	Fahr zu unterem Software Endschalter
-21	EBh	Fahr zu oberem Software Endschalter
-22	EAh	Fahr zu Position Null
-30	E2h	Referenzieren auf unteren Referenzschalter
-31	E1h	Referenzieren auf oberen Referenzschalter
-40	D8h	Referenzieren auf unteren Anschlag
-41	D7h	Referenzieren auf oberen Anschlag

Mit einem Start-Befehl kann man die gewünschte Auswahl aktivieren bzw. starten. Im PositioningReady Mode kann das Ziel gesetzt und angefahren werden (siehe 2.4.2).

4.4.1 Position oder Referenzposition direkt setzen

Antrieb wird in den **HomingReady Mode** gesetzt:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0
600h+Node-ID	5	2Fh	60h	60h	0	06h

Antwort:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	60h	60h	60h	0	0	0	0	0

Es wird das Kommando xFh ins Kontrollwort geschrieben:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
600h+Node-ID	8	23h	40h	60h	0	1Fh	0	0	0

Antwort:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	60h	40h	60h	0	0	0	0	0

Setzen der Homing-Methode Referenzieren mit Presetwert: Weise Sollposition der aktuellen Position zu.

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0
600h+Node-ID	5	2Fh	98h	60h	0	F4h

Antwort:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	60h	98h	60h	0	0	0	0	0

Ziel (bzw. 0) setzen:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
600h+Node-ID	8	23h	7Ah	60h	0	x	x	x	x

Antwort:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	60h	7Ah	60h	0	0	0	0	0

Es wird ein Start ins Kontrollwort geschrieben:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
600h+Node-ID	8	23h	40h	60h	0	1Fh	0	0	0

Antwort:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	60h	40h	60h	0	0	0	0	0

Sobald aktuelle Position neu gesetzt ist, wird das Bit14 HomingOK im Statuswort (Objekt 6041h) gesetzt.

Antrieb wird in den PositioningReady Mode gesetzt:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0
600h+Node-ID	5	2Fh	60h	60h	0	01h

Antwort:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	60h	60h	60h	0	0	0	0	0

Es wird ein Stop ins Kontrollwort geschrieben:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
600h+Node-ID	8	23h	40h	60h	0	1Fh	01h	0	0

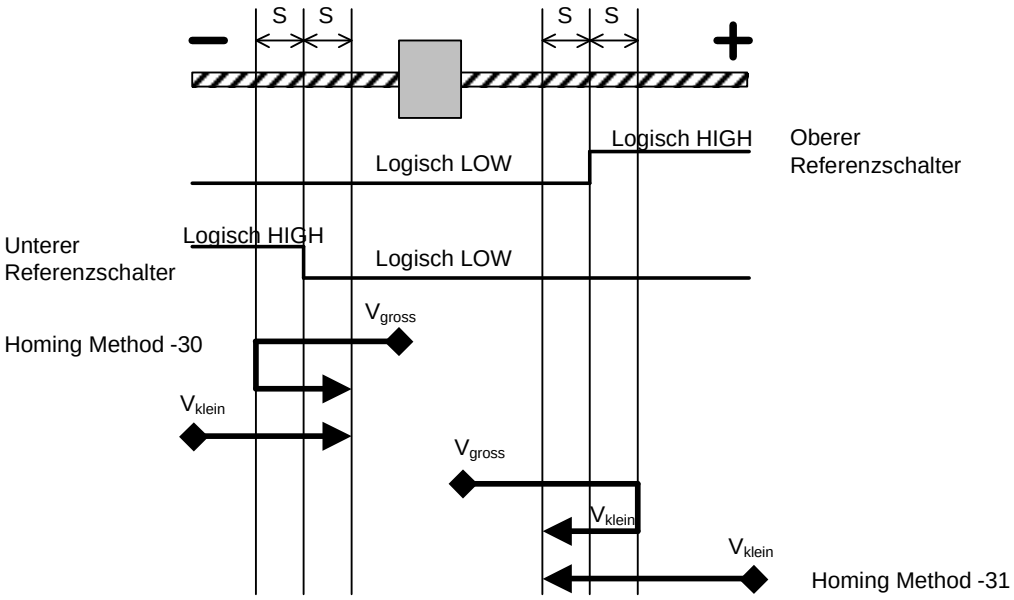
Antwort:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	60h	40h	60h	0	0	0	0	0

Nun befindet sich der Antrieb wieder im PositioningReady Mode und ist bereit für Fahraufträge.

4.4.2 Referenzieren auf Referenzschalter

Beim Übergang des Referenzschalters von logisch HIGH auf LOW wird die Referenzposition (Objekt 2111h-11h) der aktuellen Position (Objekt 6064h) zugewiesen. Sobald im Statuswort 6041h das Bit14 HomingOK gesetzt wurde, ist die Referenzierung abgeschlossen.

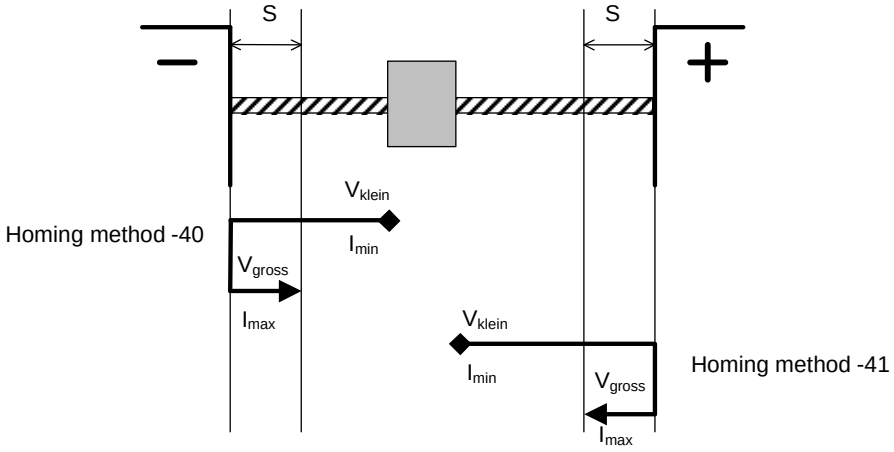


S: Free referencing distance (Objekt 2111-1Eh)
V_{gross} Referenzschalter suchen mit Profile velocity (Objekt 6081h)

Figur 14: Referenzieren auf unteren bzw. oberen Referenzschalter.

4.4.3 Referenzieren auf Anschlag

Der Antrieb fährt mit Homing speed auf den Anschlag. Wenn der Motor blockiert ist, wird die Referenzposition (Objekt 2111h-11h) der aktuellen Position (Objekt 6064h) zugewiesen. Anschliessend wird der Antrieb mit dem maximal zulässigem Strom um den Weg s frei gefahren. Sobald im Statuswort 6041h das Bit14 HomingOK gesetzt wurde, ist die Referenzierung abgeschlossen.



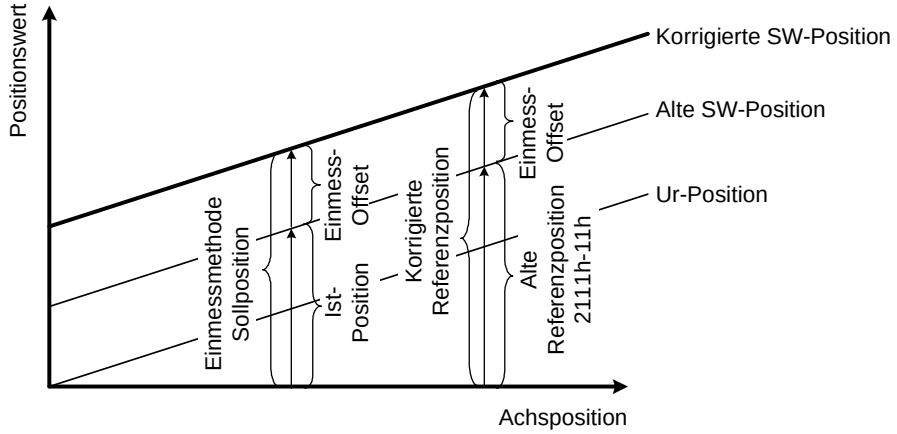
Anschlag suchen mit
V_{klein} kleiner Geschwindigkeit (Homing speed, Objekt 6099-1h)
I_{min} kleinem Strom (Max current, Objekt 6073h)

Freifahren mit
V_{gross} maximaler Geschwindigkeit (Profile velocity, Objekt 6081h)
I_{max} maximalem Strom (Max current, Objekt 6073h)
S: Freifahrweg (Free referencing distance, Objekt 2111-1Eh)

Figur 15: Referenzieren auf Anschlag in Abwärtsrichtung oder in Aufwärtsrichtung.

4.4.4 Referenzieren mit Einmessmethode

Vorraussetzung für diese Homing-Methode ist, dass Antrieb bereits referenziert wurde. Somit muss das Bit14 im Statuswort (Objekt 6041) HomingOK bereits gesetzt sein. Beim Starten dieser Homing-Methode wird zunächst der Einmess-Offset bestimmt. Anschliessend wird die Ist-Position mit der Sollposition (Target position) neu gesetzt. Zusätzlich wird die Referenzposition (2111h-11h) mit dem Einmess-Offset korrigiert. Nach Abschluss der Einmessmethode wird das Bit8 im Statuswort (6041h) CalibrationOK gesetzt.



Figur 16: Referenzieren mit der Einmessmethode.

4.4.5 Zurücksetzen der Statusbits HomingOK und CalibrationOK

Durch folgende Aktionen werden die Statusbits HomingOk (Bit14) und CalibrationOK (Bit8) auf Null zurück gesetzt:

- Wenn im Servicebetrieb ohne Buskommunikation verfahren wird.
- Nach dem Start folgender Homing-Methoden:

Objekt 6098h		Beschreibung
-12	F4h	Referenzieren mit Presetwert (überschreibe aktuelle Position mit Zielposition)
-13	F3h	Referenzieren mit Einmessmethode
-30	E2h	Referenzieren auf unteren Referenzschalter (Fahr zu unterem Referenzschalter)
-31	E1h	Referenziere auf oberen Referenzschalter (Fahr zu oberem Referenzschalter)
-40	D8h	Referenzieren auf unteren Anschlag (Fahr zu unterem Anschlag)
-41	D7h	Referenzieren auf oberen Anschlag (Fahr zu oberem Anschlag)

- Nach dem Überschreiben folgender Objekte:

Objekt	Beschreibung
607Eh-00h	Polarity
6091h-01h	Gear - Motor revolutions
6091h-02h	Gear - Gear shaft revolutions
6089h-00hh	Position notation
608Ah-00h	Position dimension
608Fh-01h	Position encoder resolution - Encoder increments
608Fh-02h	Position encoder resolution - Encoder revolutions
6092h-01h	Feed constant - Feed
6092h-02h	Feed constant - Spindle shaft revolutions

4.5 Positionen der Software-Endschalter setzen

Nachdem die Referenzposition gesetzt ist, können die Positionen der Software-Endschalter definiert werden. Die Positionen dieser Endschalter beziehen sich immer auf den Offsetwert. Werden Referenzposition oder Offset verschoben, werden auch die Positionen der Software-Endschalter mit verschoben.

Software-Endschalter anfahren

Die Positionen der Software-Endschalter können mit den gleichen Mitteln wie die Referenzposition angefahren werden.

Software-Endschalter setzen

Nachdem sich Antrieb an der Position eines Endschalters befindet, kann diese mit dem Objekt 6098h, Wert –10 oder –11 als Position für den unterer bzw. oberer Software-Endschalter übernommen werden.

Positionen der Software-Endschalter direkt eingeben

Als zusätzliche Variante können die Positionen der Software-Endschalter mit dem Objekt 607Dh, Subindex1 oder 2, direkt eingegeben werden.

Position für unteren Software-Endschalter direkt setzen:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
600h+Node-ID	8	23h	7Dh	60h	1	x	x	x	x

Antwort:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	60h	7Dh	60h	1	0	0	0	0

Position für oberen Software-Endschalter direkt setzen:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
600h+Node-ID	8	23h	7Dh	60h	2	x	x	x	x

Antwort:

COB-ID	DLC	Kommando	Objekt L	Objekt H	Subindex	Data 0	Data 1	Data 2	Data 3
580h+Node-ID	8	60h	7Dh	60h	2	0	0	0	0

4.6 Fahraufträge ausführen

Nachdem die Parameter definiert sind, kann ein Fahrauftrag erfolgen. Dazu muss zunächst ein Wechsel des Zustands vom Ready Mode zum Positioning Ready Mode erfolgen durch:

1. Objekt 6060h (Modes of operation) auf 1=Modus Positioning setzen
2. Objekt 6040h (Kontrollwort) auf 000Fh setzen
   SwitchOn, EnableVoltage, QuickStop und EnableOperation = 1

Anschliessend kann eine Sollposition vorgegeben werden durch Beschreiben des Objekts 607Ah relativ oder absolut (Bit6 des Objektes 6040h).
Zusätzliches Setzen von Bit4 im Objekt 6040h löst anschliessend den Fahrauftrag aus.
Das Setzen von Bit8 im Objekt 6040h stoppt den Antrieb (Zwischenstopp). Erneutes Fahren erfolgt durch Rücksetzen dieses Bits. Anschliessend wird die zuvor vorgegebene Sollposition angesteuert.

5 Fehlerdiagnose

5.1 Feldbus-Kommunikation

1. Falls der Antrieb über den CAN-Bus nicht angesprochen werden kann, sollten zuerst alle elektrischen Anschlüsse überprüft werden.

   Sind die Anschlüsse in Ordnung, sollte als nächstes der Feldbusbetrieb getestet werden. Dazu wird ein CAN-Monitor benötigt, welcher die CAN-Kommunikation aufzeichnet und die Telegramme darstellt.

2. Nun sollte der Antrieb beim Aus- und wieder Einschalten der Spannungsversorgung eine BootUp-Meldung absetzen.

   Sollte keine BootUp-Meldung erscheinen, so prüfen Sie, ob die Baudrate des Antriebs mit der Baudrate des CAN-Monitors übereinstimmt.

3. Wenn Sie Schwierigkeiten haben die Verbindung zu einem Teilnehmer aufzubauen, prüfen Sie die Node-ID und die Baudrate.

   Die Baudrate muss überall gleich eingestellt werden. Die Node-ID muss zwischen 1 und 127 liegen. Jeder Busteilnehmer muss eindeutig mit einer Node-ID definiert werden, d.h. es darf auf keinen Fall mehr als ein Busteilnehmer die selbe Node-ID besitzen.

5.2 Fehlerdiagnose am Antrieb

Der Antrieb verfügt über mehrere Objekte und Meldungen, welche seinen Status oder Fehlerzustände umschreiben:

- Error register (Objekt 1001h): Dieses Objekt ist ein Register für den Fehlerstatus des Antriebs.
- Objekt 1003h: In diesem Objekt werden die letzten acht Fehlercodes und Warnungen gespeichert.
- Objekt Emergency (80h + Node-ID): Hoch prioritäre Fehlermeldung eines Teilnehmers mit Error code und Error register.
- SDO Abort Message: Falls die SDO-Kommunikation nicht korrekt abläuft, enthält die SDO-Antwort einen Abort Code.

Tabelle 17: Error register (Objekt 1001h) in Emergency-Meldung Byte 2 (Byte 3...7 werden nicht verwendet).

Error register	Beschreibung
Bit 0 = 1	Allgemeiner Fehler
Bit 2 = 1	Spannungs-Fehler
Bit 3 = 1	Temperatur-Fehler
Bit 4 = 1	CAN-Bus Kommunikations-Fehler
Bit 5 = 1	Gerätespezifischer Fehler

Tabelle 18: Error code Definition für Emergency-Meldung Byte 0...1 und Byte 3...4 (siehe Kapitel Emergency-Dienst). Für zuletzt aufgetretenen Fehler, siehe Objekt 603Fh. Für letzte acht Fehler, siehe Objekt 1003h.

Error Code Byte 0...1	Zusatz Info. Byte 3...4	Fehler / Warnung	Beschreibung / Massnahme
0000h	0000h	Kein Fehler	-
3110h	0000h	Fehler Überspannung	Betriebsspannung überprüfen.
3120h	0000h	Fehler Unterspannung	Meldung während Abschalten. Kein Motorbetrieb mehr möglich. CAN-Kommunikation möglich, solange Betriebsspannung an +VsE anliegt.
4210h	0000h	Fehler Übertemperatur	Antrieb kann nicht weiter betrieben werden.

4210h	0001h	Warnung Übertemperatur	Antrieb kann weiter betrieben werden.
5441h	0000h	Warnung unterer Hardware-Endschalter aktiv	Fahren nur noch in positive Richtung möglich
5442h	0000h	Warnung oberer Hardware-Endschalter aktiv	Fahren nur noch in negative Richtung möglich
5530h	0000h	Warnung Speicher (EEPROM)	Fehler bei Speichervorgang. Speicherung wiederholen. Bei wiederholtem Auftreten Überprüfung des Antriebs im Werk notwendig.
6010h	0000h	Warnung Software (Watchdog)	Reset des Antriebs wurde intern ausgelöst. Antrieb kann weiter betrieben werden. Vorher alle Parameter überprüfen. Bei wiederholtem Auftreten Überprüfung des Antriebs im Werk notwendig.
7121h	0000h	Warnung Motor blockiert	Motor möglicherweise blockiert, Last überprüfen, Strombegrenzung (Objekt 6073h) überprüfen
7320h	0000h	Fehler Encoder	Positionsdaten nicht konsistent. Antrieb kann nicht weiter betrieben werden. Überprüfung des Antriebs im Werk notwendig.
8110h	0000h	Warnung CAN-Bus Kommunikation	Überlauf im CAN-Empfangspuffer. Möglicherweise gingen Daten gehen verloren. Kommunikation wiederholen. Buslast überprüfen.
8130h	0000h	Fehler Lifeguard oder Heartbeat	Verkabelung überprüfen. Funktion Busmaster überprüfen.
FF00h	0000h	Warnung Data valid multiturn (DVMT)	Spannung der Pufferbatterie für Positionsspeicherung hat Warngrenze erreicht. Antrieb kann noch einige Tage bis Wochen weiter betrieben werden. Danach Überprüfung des Antriebs im Werk notwendig.
FF02h	0000h	Fehler Positioning timeout	Motor möglicherweise blockiert, Last überprüfen, Strombegrenzung (Objekt 6073h) überprüfen
FF03h	0000h	Warnung Not moving timeout	Motor möglicherweise blockiert, Last überprüfen, Strombegrenzung (Objekt 6073h) überprüfen
FF10h	0000h	Warnung unterer Software-Endschalter aktiv	Fahren nur noch in positive Richtung möglich (ausser Modus Homing)
FF11h	0000h	Warnung oberer Software-Endschalter aktiv	Fahren nur noch in negative Richtung möglich (ausser Modus Homing)

Tabelle 19: SDO Abort Code Definitionen (siehe Kapitel Servicedaten-Kommunikation). Bei SDO Kommunikationsfehler wird als Antwort eine Abort Message mit Abort Code gesendet.

SDO Abort Code	Beschreibung
05040001h	Kommando-Byte wird nicht unterstützt
06010000h	Falscher Zugriff auf ein Objekt
06010001h	Lesezugriff auf Write Only
06010002h	Schreibzugriff auf Read Only
06020000h	Objekt wird nicht unterstützt
06090011h	Subindex wird nicht unterstützt
06090030h	Wert ausserhalb des zulässigen Wertebereichs
08000020h	Falsche Signatur bei Speichern oder Laden der Default-Parameter
08000021h	Speicherung konnte nicht abgeschlossen werden
08000022h	Sollposition kann nicht gesetzt werden, da Antrieb fahrend

5.3 EMV-gerechte Verdrahtung

Halten Sie die Spezifikation der CAN-Leitungen ein

Tabelle 20: Spezifikation der CAN-Leitungen.

Gesamtlänge Bussystem	< 300 m	< 1000 m
Kabeltyp	LIYCY 2 x 2 x 0,5 mm² (paarverseilt mit Abschirmung)	CYPIMF 2 x 2 x 0,5 mm² (paarverseilt mit Abschirmung)
Leitungswiderstand	≤ 40 Ohm/km	≤ 40 Ohm/km
Kapazitätsbelag	≤ 130 nF/km	≤ 60 nF/km
Anschluss	Paar 1 (weiss / braun): CAN-GND und +Vs Paar 2 (grün / gelb): CAN-HIGH und CAN-LOW	

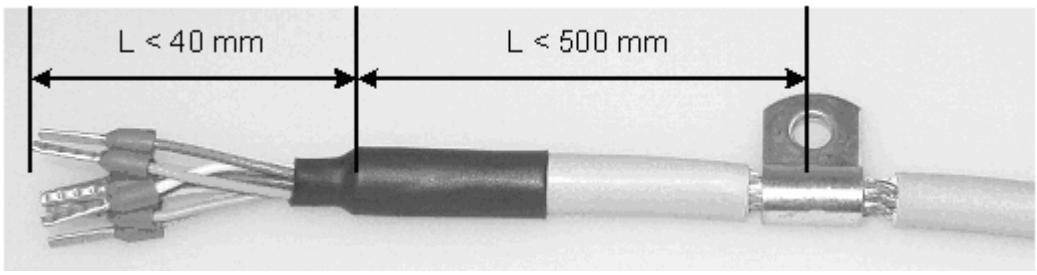
- Nur Leitungen verwenden, die ein zusätzliches Aderpaar für CAN-GND haben.
- Nur mit korrekt angeschlossenem CAN-GND ist ein störungsfreier Busbetrieb möglich.

Schliessen Sie die Bus-Abschlusswiderstände an
Am physikalischen Anfang und am physikalischen Ende des Bussystems muss jeweils ein Abschlusswiderstand
von 120 Ohm angeschlossen sein.

Schirmen Sie Busleitung und Motorleitung korrekt ab

- Den Schirm der Busleitung beidseitig auflegen.
- Schirmschellen verwenden, um den Schirm großflächig an PE anzubinden.
- Busleitung und Motorleitung so abisolieren, dass sie großflächigen Kontakt zur Schirmschelle hat.
- Schirmschelle der Busleitung und der Motorleitung mit einem Mindestabstand von 50 mm zueinander montieren.

Falls eine Erdung über einen der Antriebsflansche nicht sichergestellt werden kann, muss die Kabelabschirmung gemäss Fig. 18 geerdet werden. Bei beidseitiger Erdung eines Kabels muss der Potentialausgleich zur Verhinderung einer Erdschleife über eine separate Potential-Ausgleichsleitung mit ausreichendem Leiterquerschnitt sicher gestellt werden.



Figur 17: EMV-gerechte Verdrahtung durch korrektes Anschliessen der Kabelabschirmung.

Halten Sie unbedingt die zulässigen Bus-Leitungslängen und Stichleitungslängen nach Tabelle 21 ein

Die maximal zulässige Gesamt-Leitungslänge und Gesamt-Stichleitungslänge

- ist abhängig von der Baudrate und
- kann in mehrere Segmente bzw. Einzelstichleitungen aufgeteilt werden.

Tabelle 21: Maximale Gesamt-Busleitungslänge (mit Abschlusswiderstand) und maximale Stichleitungslänge (ohne Abschlusswiderstand) in Abhängigkeit von der Baudrate.

Baudrate [kBit/s]	10	20	50	100	125	250	500	800	1000
Gesamt-Buslänge	5000 m	3000 m	1000 m	500 m	400 m	200 m	75 m	30 m	25 m
Gesamt-Stichleitungslänge	1360 m	875 m	350 m	175 m	140 m	70 m	35 m	20 m	17 m
Einzel-Stichleitungslänge	270 m	175 m	70 m	35 m	28 m	14 m	7 m	4 m	3 m

Die maximal zulässige Segment-Leitungslänge nach Tabelle 22 ist abhängig

- von dem verwendetem Kabelquerschnitt
- von der Anzahl der Kommunikationsteilnehmer

Tabelle 22: Maximale Segment-Leitungslängen in Abhängigkeit von der Anzahl der Kommunikationsteilnehmer und dem Leitungsquerschnitt.

Anzahl der Kommunikationsteilnehmer	Leitungsquerschnitt		
	0,75 mm²	0,5 mm²	0,25 mm²
32	550 m	360 m	200 m
64	470 m	310 m	170 m
100	410 m	270 m	150 m

- Mit Repeatern teilen Sie die max. Gesamt-Leitungslänge in Segmente auf, wenn Sie größere Leitungslängen realisieren müssen, als nach Tabelle 22 zulässig sind.
- Repeater erfordern eine Verkürzung der max. Gesamt-Leitungslängen nach Tabelle 21. Diese ist abhängig vom Repeater-Typ. Pro Repeater beträgt die typische Verkürzung der max. Gesamt-Leitungslänge ca. 30 m.
- Die max. Gesamt-Leitungslänge mit Repeatern ergeben sich aus den Längen in Tabelle 21 abzüglich der Summe aller Leitungslängenverkürzungen durch Repeater.
  $L_{Bus\ max.} = L_{max.\ (Tab.\ 2)} - (\sum$ Leitungslängenverkürzung * Repeateranzahl)
- Ein Repeater wird auch empfohlen als
  Serviceschnittstelle zum störungsfreies Ankoppeln im laufenden Bus-Betrieb oder als Einmessschnittstelle zur galvanischen Trennung des Programmiergeräts.

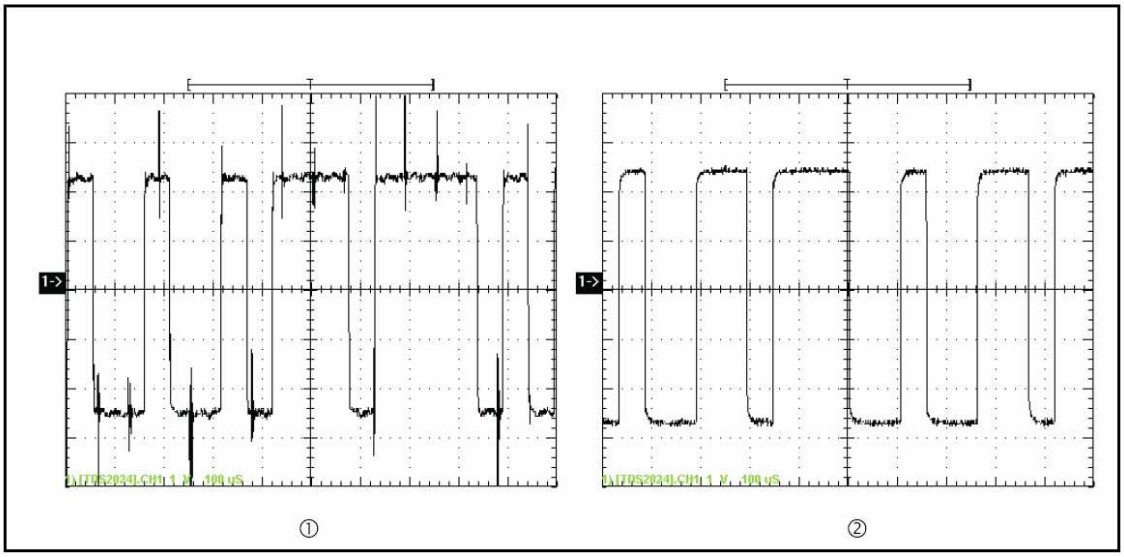
Vermeiden Sie Potentialdifferenzen zwischen den Knoten des CAN-Netzwerks

Potentialdifferenzen (Bezug zu PE) zwischen den Knoten des CAN-Netzwerks können Kommunikationsstörungen verursachen oder zur Zerstörung der CAN-Teilnehmer führen.

Vermeiden Sie Potentialdifferenzen, indem Sie

- jeden CAN-Teilnehmer auf dem kürzesten, möglichst niederohmigen Weg mit dem gleichen Erd-Bezugspotential (PE) der Maschine/Anlage verbinden.
- eine Potentialausgleichsleitung zwischen den Kommunikationsteilnehmern verwenden.
- den Erdbezug der Maschine/Anlage zur Gesamterde niederohmig ausführen.

Erkennen von EMV-Störungen im Signal-Oszillogramm



Figur 18: Oszillogramme der CAN-Signale (1) mit und (2) ohne Störspannung (Messpunkte: CAN_HIGH zu CAN_LOW).

Zur Quantifizierung von Störungen sind Messungen mit einem CAN-Analyser erforderlich. Hiermit können wichtige Bus-Parameter wie beispielsweise die Buslast oder die Anzahl von Errorframes ermittelt und weitergehende Analysen durchgeführt werden.

6 Technische Daten

6.1 Elektrische und mechanische Daten

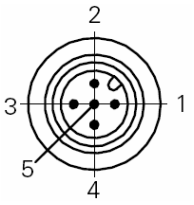
Tabelle 23: Elektrische und mechanische Daten (siehe auch Katalog bzw. Datenblatt).

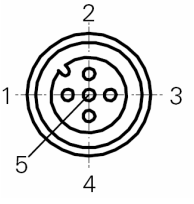
Technische Daten	MSIA 42	MSIA 53	MSIA 68	MSIA 83
Betriebsspannung	24 VDC ± 10%	24 VDC ± 10%	24 VDC ± 10%	24 VDC ± 10%
Max. Stromaufnahme	10 A	8 A	20 A	20 A
Betriebsart[2]	S1	S1	S1	S1
Nennstrom	2,3 A	1,7 A	5,8 A	7,8 A
Nennabgabeleistung[1]	36 W	22 W	80 W	144 W
Nenndrehzahl[1]	4'720 U/min	3'600 U/min	3'500 U/min	3'600 U/min
Nenndrehmoment[1, 3]	0,07 Nm	0,06 Nm	0,22 Nm	0,38 Nm
Positionierauflösung[1]	0,02°	0,02°	0,02°	0,02°
Positioniergenauigkeit[1]	± 1°	± 1°	± 1°	± 1°
Wiederholgenauigkeit[1]	0,3°	0,3°	0,3°	0,3°
erfassbare Umdrehungen[1]	262'144	262'144	262'144	262'144
Betriebsstunden	2'500 h	20'000 h	20'000 h	20'000 h
Material Gehäuse	Zink-Druckguss	Aluminium	Aluminium	Aluminium
Achsbelastung axial / radial	110 N / 300 N	40 N / 400 N	40 N / 400 N	40 N / 400 N
Kommunikation				
Feldbus	CANopen Version 2.0B			
Profil	CANopen/CIA DS-301 Version 4.02			
	DSP-305 (LSS), DSP-402 Version 2.0			
Node-ID	0...99, einstellbar (Default-Wert 1)			
max. Baudrate	1 MBit/s, einstellbar (Default-Wert 50 kBit/s)			
Schalteingänge / -ausgänge	4+1, Pull-Down Widerstand 5,6 kΩ, HIGH >18 V (>2 ms), LOW <4 V (>2 ms)			
Stromaufnahme	typ. 90 mA (siehe I_E Figur 19)			
Umgebungsbedingungen				
Schutzklasse	IP 54			
Umgebungstemperatur	-15...+40 ℃			
Störaussendung	EN 61000-6-4			
Störfestigkeit	EN 61000-6-2			
Vibration	IEC 60068-2-6			
Schock	IEC 60068-2-27			

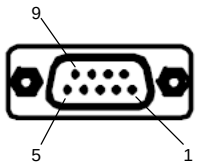
[1]ohne Getriebe
[2]S1 Dauerbetrieb (DIN EN 60034-1)
[3]bei Betriebsfaktor = 1
Nenndaten bei +40℃ Umgebungstemperatur

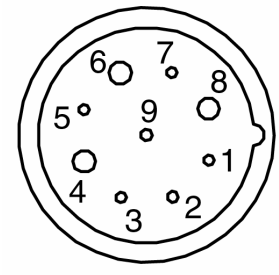
6.2 Anschlussbelegung

Tabelle 24: Anschlussbelegung.

CAN-Bus IN (Rundstecker 5-Pol M12)			
 A-codiert	Pin	Signal	Beschreibung
	1	n.c.	Not connected
	2	+VsE	Not connected
	3	CAN_GND	CAN-Ground / 0VsE
	4	CAN_H	Bus dominant HIGH
	5	CAN_L	Bus dominant LOW

CAN-Bus OUT (Runddose 5-Pol M12)			
 A-codiert	Pin	Signal	Beschreibung
	1	n.c.	Not connected
	2	+VsE	Not connected
	3	CAN_GND	CAN-Ground / 0VsE
	4	CAN_H	Bus dominant HIGH
	5	CAN_L	Bus dominant LOW

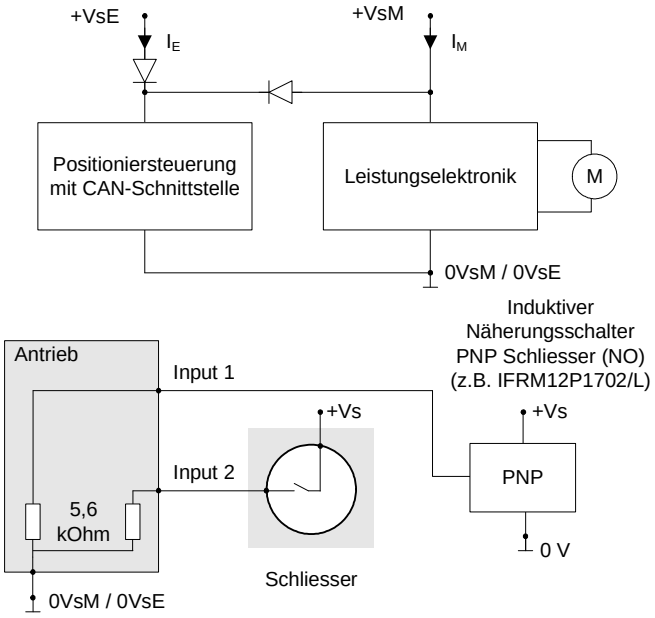
Speisung (9-Pol D-Sub-Stecker)			
	Pin	Signal	Beschreibung
	1	+VsE	+24 VDC Betriebsspannung Elektronik
	2	Input 1	Eingang programmierbar
	3	Input 2	Eingang programmierbar
	4	Input 3	Eingang programmierbar
	5	Input 4	Eingang programmierbar
	6	0 VME	0 VDC Betriebsspannung Motor / Elektronik
	7	0 VME	0 VDC Betriebsspannung Motor / Elektronik
	8	+VsM	+24 VDC Betriebsspannung Motor
	9	+VsM	+24 VDC Betriebsspannung Motor

Speisung (9-Pol M23-Stecker)			
	Pin	Signal	Beschreibung
	1	Input 1	Eingang programmierbar
	2	Input 2	Eingang programmierbar
	3	Input 3	Eingang programmierbar
	4	PE	Erdung
	5	Input 4	Eingang programmierbar
	6	0 VME	0 VDC Betriebsspannung Motor / Elektronik
	7	Output 1	Ausgang programmierbar
	8	+VsM	+24 VDC Betriebsspannung Motor
	9	+VsE	+24 VDC Betriebsspannung Elektronik

6.3 Blockschaltbild und Schalteingänge

Der Antrieb verfügt über zwei Anschlüsse zur Spannungsversorgung (Figur 19). Bereits bei Anschluss der Betriebsspannung an +VsM ist der Antrieb voll funktionsfähig. Bei Abschalten von +VsM beispielsweise über

einen Not-Aus-Kreis kann die Positioniersteuerung weiterhin über +VsE gespeist werden zur Aufrechterhaltung z.B. der Kommunikation. Ein Anlaufen des Motor ist in diesem Fall ausgeschlossen, ein Fehler wird nicht angezeigt.



Figur 19: Blockschaltbild des Antriebs und Beispiel für Beschaltung der Schalteingänge.

Zusätzlich verfügt der Antrieb über 2 digitale Schalteingänge mit pnp-Charakteristik (Funktion programmierbar, Objekt 2111h-0Dh) zum Anschluss handelsüblicher Positionssensoren. Mögliche Funktionen sind:

- Aktivierung des Tippbetriebs (Jogging+, Jogging-)
- Hardware-Endschalter oder Referenzierung
- Steuerung des Kontrollworts (Objekt 6040h)

6.4 Rückspeiseschutz beim Abbremsen des Motors

Bei geregeltem Abbremsen des Antriebs, insbesondere in einer dynamischem Applikation mit geringer Reibung kann die entstehende elektrische Energie die Spannungsversorgung beeinflussen und zu einer Erhöhung der Betriebsspannung über den zulässigen Bereich von etwa 28 VDC führen. In diesem Fall muss ein Rückspeiseschutz für die Bremsenergie vorgesehen werden, die von den folgenden Faktoren abhängt:

- Vorhandene Trägheitsmomente (Antriebe und deren Lasten)
- Reibung und Wirkungsgrad der Mechanik
- Anzahl der gleichzeitig abgebremsten Antriebe an der selben Spannungsversorgung (beispielsweise betriebsbedingt durch Not-Aus oder gleichzeitiges Verfahren)

Die Bremsenergie eines Antriebs ohne Berücksichtigung von Verlusten berechnet sich nach folgender Formel:

$$W = \frac{1}{2} \cdot J \cdot \omega^2 = \frac{\pi^2}{1800} \cdot J \cdot n^2$$

W : Bremsenergie $[Ws = kgm^2/s^2]$

J : Massenträgheitsmoment $[kgm^2]$

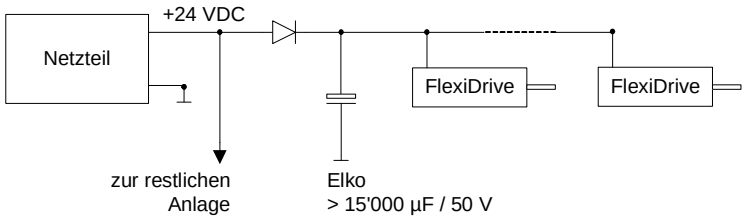
ω : Kreisfrequenz $= (2 \cdot \pi \cdot n) / 60$ [1/s] mit n [U/min]

Die typische Bremsenergie beträgt etwa 0,5 Ws beim MSIA 42 und 2 Ws beim MSIA 83 (Bremsen aus Nenndrehzahl im S3-Betrieb, nur Berücksichtigung nur des Motorträgheitsmoments, typische Verluste).

Trägheitsmoment und Bremsenergie sind direkt proportional, d.h. doppeltes Trägheitsmoment bedeutet doppelte Bremsenergie.

Zur Umwandlung der Bremsenergie gibt es folgende Möglichkeiten:

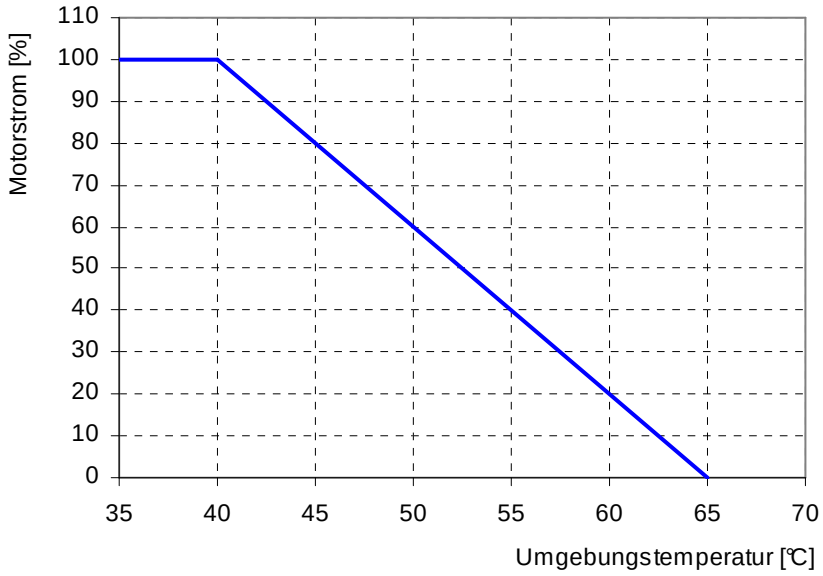
- Rückspeiseschutz mit Diode und Kondensator (siehe Figur 20). Die Bremsenergie wird im Kondensator zwischengespeichert.
- Verwendung eines Netzteils mit Rückspeisemöglichkeit bzw. eines Power-Management-Moduls (z.B. Siemens DC-PMM/24V)



Figur 20: Beispiel Rückspeiseschutz mit Diode und Kondensator.

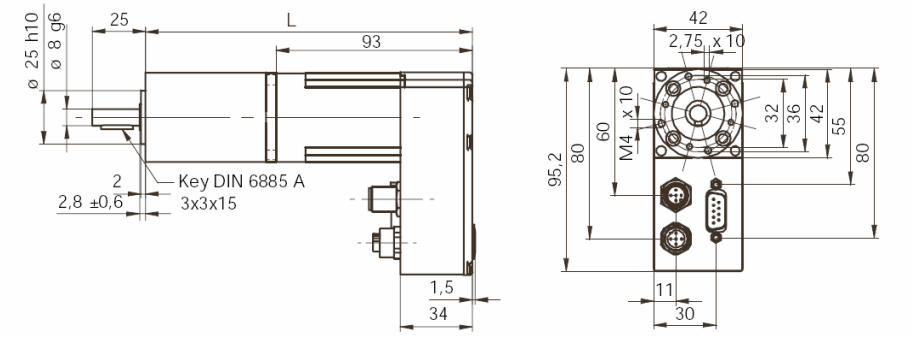
6.5 Motorstromreduktion bei erhöhter Umgebungstemperatur

Der Motorstrom ist abhängig von der Umgebungstemperatur. Betrieb des Antriebs bis 40℃ gemäss Nenndaten. Bei erhöhter Umgebungstemperatur kann der Antrieb dennoch betrieben werden bis 65 ℃ mit entsprechender Reduktion des Motordauerstroms (siehe Figur 21).

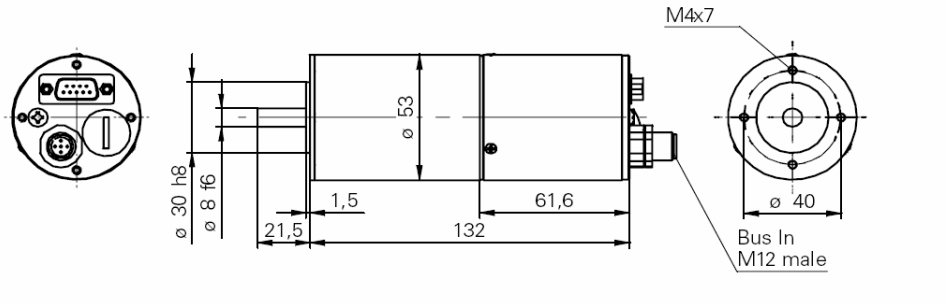


Figur 21: Motorstromreduktion bei erhöhter Umgebungstemperatur in Prozent des Nennstroms bei 40°C.

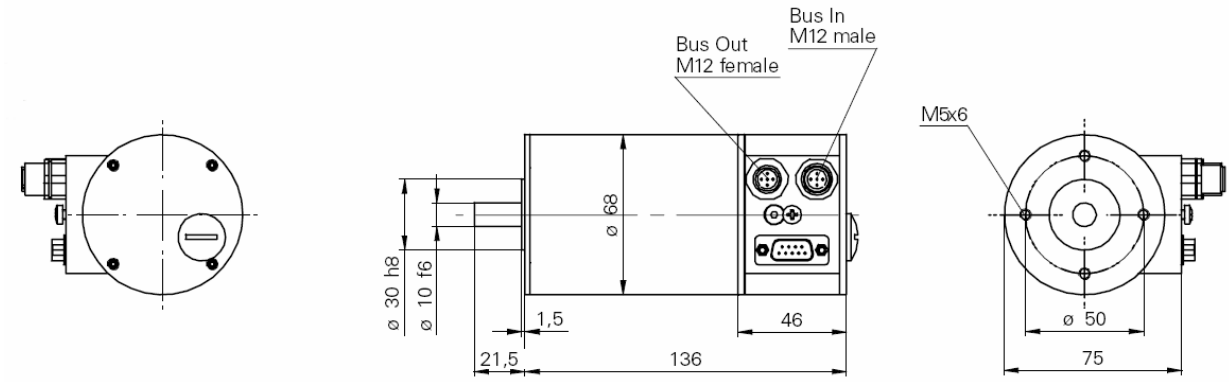
6.6 Abmessungen und Einbaumasse



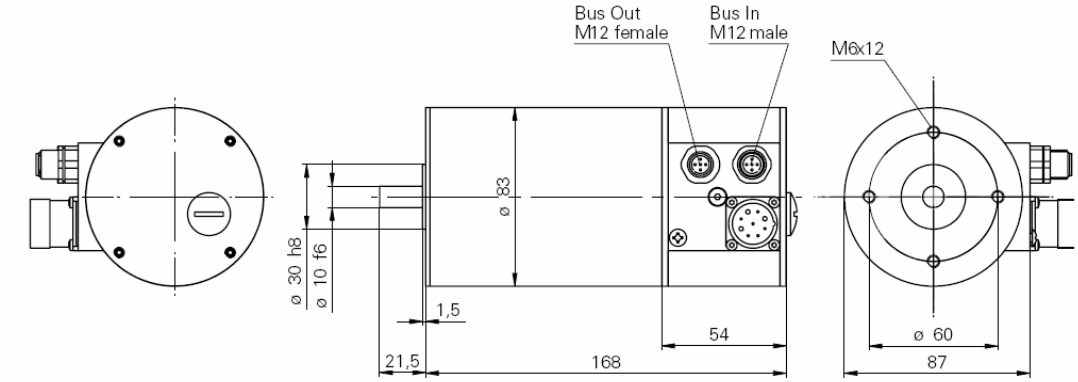
Figur 22: Abmessungen MSIA 42C2PL12-C43xP4xxx.



Figur 23: Abmessungen MSIA 53C2PA11-N52xK0xxx.



Figur 24: Abmessungen MSIA 68C2PR12-N64xK0xxx.



Figur 25: Abmessungen MSIA 83C2PR32-N86xK0xxx.

6.7 Befehlssequenzen als Beispiel

Tabelle 25: Befehlssequenz zur Identifikation des Antriebs.

Master (PLD) an Slave (MSIA)	Objekt	Wert	Slave (MSIA) an Master PLD	Wert
SDO Upload Request (Get Hardware version)	1009h	00000000h	SDO Upload Response	xxxxxxxxh
SDO Upload Request (Get Software version COM)	100Ah	00000000h	SDO Upload Response	xxxxxxxxh
SDO Upload Request (Get Identity object, Product code)	1018h-02h	00000000h	SDO Upload Response	00000100h
SDO Upload Request (Get Identity object, Revision number)	1018h-03h	00000000h	SDO Upload Response	xxxxxxxxh
SDO Upload Request (Get Identity object, Serial number)	1018h-04h	00000000h	SDO Upload Response	xxxxxxxxh

Tabelle 26: Befehlssequenz zum Setzen der Applikationsparameter.

Master (PLD) an Slave (MSIA)	Objekt	Wert	Slave (MSIA) an Master PLD	Wert
SDO Download Request (Set Maximum current)	6073h	03E8h	SDO Download Response	00000000h
SDO Download Request (Set Polarity)	607Eh	00h	SDO Download Response	00000000h
SDO Download Request (Set Position notation)	6089h	FDh	SDO Download Response	00000000h
SDO Download Request (Set Position dimension)	608Ah	01h	SDO Download Response	00000000h
SDO Download Request (Set Velocity notation)	608Bh	00h	SDO Download Response	00000000h
SDO Download Request (Set Velocity dimension)	608Ch	A3h	SDO Download Response	00000000h
SDO Download Request (Set Acceleration notation)	608Dh	00h	SDO Download Response	00000000h
SDO Download Request (Set Acceleration dimension)	608Eh	A6h	SDO Download Response	00000000h
SDO Download Request (Set Position encoder resolution, Encoder increments)	608Fh-01h	00004000h	SDO Download Response	00000000h
SDO Download Request (Set Position encoder resolution, Encoder revolutions)	608Fh-02h	00000001h	SDO Download Response	00000000h
SDO Download Request (Set Gear ratio, Motor revolutions)	6091h-01h	00000012h	SDO Download Response	00000000h
SDO Download Request (Set Gear ratio, Gear shaft revolutions)	6091h-02h	00000001h	SDO Download Response	00000000h
SDO Download Request (Set Feed constant, Feed)	6092h-01h	00000001h	SDO Download Response	00000000h
SDO Download Request (Set Feed constant, Spindle shaft revolutions)	6092h-02h	00000001h	SDO Download Response	00000000h
SDO Download Request (Set Software position limit, Minimum software position limit)	607Dh-01h	00000000h	SDO Download Response	00000000h
SDO Download Request (Set Software position limit, Maximum software position limit)	607Dh-02h	00000000h	SDO Download Response	00000000h
SDO Download Request (Set Positioning parameter, Backlash compensation)	2111h-01h	00h	SDO Download Response	00000000h
SDO Download Request (Set Positioning parameter, Backlash delta)	2111h-02h	00000064h	SDO Download Response	00000000h
SDO Download Request (Store application parameters)	1010h-03h	65766173h	SDO Download Response	00000000h

Tabelle 27: Befehlssequenz zum Setzen der Kommunikationsparameter.

Master (PLD) an Slave (MSIA)	Objekt	Wert	Slave (MSIA) an Master PLD	Wert
SDO Download Request (Set Receive PDO1 parameter, R_PDO1 type)	1400h-02h	FEh	SDO Download Response	00000000h
SDO Download Request (Set Receive PDO2 parameter, R_PDO2 type)	1401h-02h	FEh	SDO Download Response	00000000h
SDO Download Request (Set Transmit PDO1 parameter, T_PDO1 type)	1800h-02h	FEh	SDO Download Response	00000000h
SDO Download Request (Set Transmit PDO1 parameter, T_PDO1 Event Timer)	1800h-05h	0203h	SDO Download Response	00000000h
SDO Download Request (Set T_PDO1 add-on)	2800h	01h	SDO Download Response	00000000h
SDO Download Request (Set Transmit PDO2 parameter, T_PDO2 type)	1801h-02h	FEh	SDO Download Response	00000000h
SDO Download Request (Set Transmit PDO2 parameter, T_PDO2 Event Timer)	1801h-05h	0100h	SDO Download Response	00000000h
SDO Download Request (Set T_PDO2 add-on)	2801h	01h	SDO Download Response	00000000h
SOD Download Request (Set Consumer heartbeat time)	1016h-01h	yyyyzzzzh	SDO Download Response	00000000h
SDO Download Request (Set Producer heartbeat time)	1017h	xxxxh	SDO Download Response	00000000h
SDO Download Request (Store communication parameters)	1010h-02h	65766173h	SDO Download Response	00000000h

Tabelle 28: Wechsel vom Zustand Ready in den Zustand PositioningReady (einmalig nach Einschalten oder Reset des Antriebs).

Master (PLD) an Slave (MSIA)	Objekt	Wert	Slave (MSIA) an Master PLD	Wert
SDO Download Request (Set Mode of operation)	6060h	01h	SDO Download Response	00000000h
SDO Download Request (Set Controlword)	6040h	000Fh	SDO Download Response	00000000h

Tabelle 29: Tippen in positiver Richtung (Antrieb im Zustand PositioningReady).

Master (PLD) an Slave (MSIA)	Objekt	Wert	Slave (MSIA) an Master PLD	Wert
SDO Download Request (Set Controlword)	6040h	081Fh	SDO Download Response	00000000h

Tabelle 30: Tippen in negativer Richtung (Antrieb im Zustand PositioningReady).

Master (PLD) an Slave (MSIA)	Objekt	Wert	Slave (MSIA) an Master PLD	Wert
SDO Download Request (Set Controlword)	6040h	101Fh	SDO Download Response	00000000h

Tabelle 31: Positionieren über SDO (Antrieb im Zustand PositioningReady).

Master (PLD) an Slave (MSIA)	Objekt	Wert	Slave (MSIA) an Master PLD	Wert
NMT Cmd (Start Node, only once after power-on / reset)		01h	(T_PDO1, T_PDO2)	
SDO Download Request (Set Target position)	607Ah	xxxxxxxxh	SDO Download Response	00000000h
SDO Download Request (Set Controlword)	6040h	001Fh	SDO Download Response	00000000h
SDO Upload Request (Get Statusword)	6041h		SDO Upload Response	4237h
-			T_PDO1	5237h
-			T_PDO2	xxxxxxxxh
SDO Upload Request (Get Statusword)	6041h		SDO Upload Response	5237h
-			(T_PDO1, T_PDO2)	
-			T_PDO1	4637h

Tabelle 32: Positionieren über R_PDO (Antrieb im Zustand PositioningReady).

Master (PLD) an Slave (MSIA)	Objekt	Wert	Slave (MSIA) an Master PLD	Wert
NMT Cmd (Start Node, only once after power-on / reset)		01h	(T_PDO1, TPDO2)	
R_PDO2 (Set Target position + Controlword)		xxxxxxxxh 001Fh	T_PDO1	5237h
SDO Upload Request (Get Statusword)	6041h		SDO Upload Response	4237h
-			T_PDO2	xxxxxxxxh
SDO Upload Request (Get Statusword)	6041h		SDO Upload Response	5237h
-			(T_PDO1, T_PDO2)	
-			T_PDO1	4637h

Tabelle 33: Stoppen eines Fahrauftrags.

Master (PLD) an Slave (MSIA)	Objekt	Wert	Slave (MSIA) an Master PLD	Wert
SDO Download Request (Set Controlword)	6040h	011Fh	SDO Download Response	00000000h

Tabelle 34: Quittieren eines Fehlers.

Master (PLD) an Slave (MSIA)	Objekt	Wert	Slave (MSIA) an Master PLD	Wert
SDO Download Request (Set Controlword)	6040h	008Fh	SDO Download Response	00000000h

Tabelle 35: Referenzieren mit Presetwert.

Master (PLD) an Slave (MSIA)	Objekt	Wert	Slave (MSIA) an Master PLD	Wert
NMT Cmd (Start Node)		01h	(T_PDO1, T_PDO2)	
SDO Download Request (Set Mode of operation)	6060h	06h	SDO Download Response	00000000h
SDO Download Request (Set Controlword)	6040h	000Fh	SDO Download Response	00000000h
SDO Download Request (Set Homing method)	6098h	F4h	SDO Download Response	00000000h
SDO Download Request (Set Target position)	607Ah	xxxxxxxxh	SDO Download Response	00000000h
SDO Download Request (Set Controlword)	6040h	001Fh	SDO Download Response	00000000h
-			(T_PDO1, T_PD02)	
-			T_PDO2	xxxxxxxxh
-			T_PDO1	4237h
SDO Download Request (Store application parameters)	1010h-03h	65766173h	SDO Download Response	00000000h
SDO Download Request (Set Mode of operation)	6060h	01h	SDO Download Response	00000000h
SDO Download Request (Set Controlword)	6040h	000Fh	SDO Download Response	00000000h

Tabelle 36: Referenzieren auf Referenzschalter in negativer Richtung.

Master (PLD) an Slave (MSIA)	Objekt	Wert	Slave (MSIA) an Master PLD	Wert
NMT Cmd (Start Node)		01h	(T_PDO1, T_PDO2)	
SDO Download Request (Set Positioning parameter, Input1)	2111h-0Dh	12h	SDO Download Response	00000000h
SDO Download Request (Set Positioning parameter, Reference Position)	2111h-11h	xxxxxxxxh	SDO Download Response	00000000h
SDO Download Request (Set Mode of operation)	6060h	06h	SDO Download Response	00000000h
SDO Download Request (Set Controlword)	6040h	000Fh	SDO Download Response	00000000h
SDO Download Request (Set Homing method)	6098h	E2h	SDO Download Response	00000000h
SDO Download Request (Set Controlword)	6040h	001Fh	SDO Download Response	00000000h
-			T_PDO1	1237h
-			T_PDO2	xxxxxxxxh
-			(T_PDO1, T_PDO2)	
-			T_PDO1	4237h
SDO Download Request (Store application parameters)	1010h-03h	65766173h	SDO Download Response	00000000h
SDO Download Request (Set Mode of operation)	6060h	01h	SDO Download Response	00000000h
SDO Download Request (Set Controlword)	6040h	000Fh	SDO Download Response	00000000h

Tabelle 37: Referenzierung auf unteren Anschlag.

Master (PLD) an Slave (MSIA)	Objekt	Wert	Slave (MSIA) an Master PLD	Wert
NMT Cmd (Start Node)		01h	(T_PDO1, T_PDO2)	
SDO Download Request (Set Positioning parameter, Reference Position)	2111h-11h	xxxxxxxxh	SDO Download Response	00000000h
SDO Download Request (Set Maximum current)	6073h	01F4h	SDO Download Response	00000000h
SDO Download Request (Set Mode of operation)	6060h	06h	SDO Download Response	00000000h
SDO Download Request (Set Controlword)	6040h	000Fh	SDO Download Response	00000000h
SDO Download Request (Set Homing method)	6098h	D8h	SDO Download Response	00000000h
SDO Download Request (Set Controlword)	6040h	001Fh	SDO Download Response	00000000h
-			T_PDO1	1237h
-			T_PDO2	xxxxxxxxh
-			(T_PDO1, T_PDO2)	
-			Emergency Message	7121h
-			Emergency Message	0000h
-			(T_PDO1, T_PDO2)	
-			T_PDO1	4237h
-			T_PDO2	xxxxxxxxh
SDO Download Request (Store application parameters)	1010h-03h	65766173h	SDO Download Response	00000000h
SDO Download Request (Set Mode of operation)	6060h	01h	SDO Download Response	00000000h
SDO Download Request (Set Controlword)	6040h	000Fh	SDO Download Response	00000000h

Tabelle 38: Referenzierung mit Einmessmethode.

Master (PLD) an Slave (MSIA)	Objekt	Wert	Slave (MSIA) an Master PLD	Wert
NMT Cmd (Start Node)		01h	(T_PDO1, T_PDO2)	
SDO Download Request (Set Target position)	607Ah	xxxxxxxxh	SDO Download Response	00000000h
SDO Download Request (Set Mode of operation)	6060h	06h	SDO Download Response	00000000h
SDO Download Request (Set Controlword)	6040h	000Fh	SDO Download Response	00000000h
SDO Download Request (Set Homing method)	6098h	F3h	SDO Download Response	00000000h
SDO Download Request (Set Controlword)	6040h	001Fh	SDO Download Response	00000000h
-			T_PDO1	4337h
-			T_PDO2	xxxxxxxxh
SDO Download Request (Store application parameters)	1010h-03h	65766173h	SDO Download Response	00000000h
SDO Download Request (Set Mode of operation)	6060h	01h	SDO Download Response	00000000h
SDO Download Request (Set Controlword)	6040h	000Fh	SDO Download Response	00000000h