

# CAN-Bus®

## Feldbus Schnittstelle für S300 / S700



Ausgabe: August 2020  
Originalanleitung



Für einen ordnungsgemäßen und sicheren Gebrauch diesen Anleitungen folgen. Für künftige Verwendung aufbewahren.

## Bisher erschienene Ausgaben :

Ausgabe	Bemerkung
10 / 2005	Erstausgabe (gültig ab Firmware 1.32 - CAN Version 0.41)
11 / 2005	Kombiniert: Referenz und Beispiele
09 / 2006	Neues Design, SDO umbenannt in Objekt, Index erweitert
11 / 2006	Aktualisierung für S700 Familie, Abschlusswiderstände (Interface) korrigiert, neue Objekte
08 / 2007	Symbole, Normen
01 / 2008	Korrekturen, neue Objekte, Symbole nach ANSI Z535
12 / 2008	Kleine Korrekturen
07 / 2009	Statusmaschine Bit 6 korrigiert, CAN Logo, Produkt Logo
12 / 2009	Kleinere Korrekturen
06 / 2010	SDOs für Safety Erweiterungskarte neu, diverse Korrekturen
09 / 2010	neue Objekte
12 / 2010	Objekt 6094, neuer Firmenname
05 / 2012	Beispiele korrigiert, Design Titelseite
07 / 2014	Warnhinweise aktualisiert, Design Titelseite
10 / 2014	Safetykarten S3 und S4 hinzugefügt, Referenzfahrtarten aktualisiert
04 / 2016	Warnsymbole, Europäische Richtlinien aktualisiert, sichere Spannung 50V
07 / 2016	ASCII Kommando Referenz aktualisiert, bestimmungsgemäße Verwendung aktualisiert
11 / 2018	Layout Warnhinweise und Lesehinweis Titelseite verändert, Anforderung Fachleute aktualisiert
12 / 2019	Layout aktualisiert
08 / 2020	Layout aktualisiert

## Warenzeichen

- Windows ist ein eingetragenes Warenzeichen der Microsoft Corporation.
- CANopen ist ein eingetragenes Warenzeichen des CAN in Automation (CiA) e.V.

## Technische Änderungen zur Verbesserung der Leistung der Geräte ohne vorherige Ankündigung vorbehalten.

Dieses Dokument ist geistiges Eigentum von Kollmorgen. Alle Rechte vorbehalten. Kein Teil dieses Werkes darf in irgendeiner Form (Fotokopie, Mikrofilm oder in einem anderen Verfahren) ohne schriftliche Genehmigung von Kollmorgen reproduziert oder elektronisch verarbeitet, vervielfältigt oder verbreitet werden.

# 1 Inhaltsverzeichnis

<b>1 Inhaltsverzeichnis</b> .....	<b>3</b>
<b>2 Allgemeines</b> .....	<b>9</b>
2.1 Über dieses Handbuch .....	9
2.2 Zielgruppe .....	9
2.3 Hinweise für die Online-Ausgabe (PDF-Format) .....	9
2.4 Bestimmungsgemäße Verwendung .....	10
2.5 Verwendete Symbole .....	10
2.6 Verwendete Abkürzungen .....	10
2.7 Realisierte Grundfunktionen über CANopen .....	11
2.8 Übertragungsgeschwindigkeit und -verfahren .....	11
2.9 Verhalten bei Kommunikationsstörungen BUSOFF .....	11
<b>3 Installation / Inbetriebnahme</b> .....	<b>12</b>
3.1 Wichtige Hinweise .....	12
3.2 Einstellen der Stationsadresse und der Übertragungsrate .....	12
3.3 CAN-Bus Interface (X6) .....	13
3.4 CAN Busleitung .....	13
3.5 Leitfaden zur Inbetriebnahme .....	14
3.6 Wichtige Konfigurationsparameter für den CAN-Bus-Betrieb .....	14
<b>4 CANopen Kommunikationsprofil</b> .....	<b>15</b>
4.1 Allgemeine Erläuterungen zu CAN .....	15
4.2 Aufbau des Kommunikationsobjekt Identifiers .....	16
4.3 Definition der verwendeten Datentypen .....	17
4.3.1 Basisdatentypen .....	17
4.3.1.1 Unsigned Integer .....	17
4.3.1.2 Signed Integer .....	17
4.3.2 Zusammengesetzte Datentypen .....	17
4.3.3 Erweiterte Datentypen .....	18
4.3.3.1 Octet String .....	18
4.3.3.2 Visible String .....	18
4.4 Kommunikationsobjekte .....	18
4.4.1 Network Management Objects (NMT) .....	19
4.4.2 Synchronization Object (SYNC) .....	19
4.4.3 Time-Stamp Objekt (TIME) .....	19
4.4.4 Emergency Object (EMCY) .....	19
4.4.4.1 Verwendung des Emergency Object .....	20
4.4.4.2 Aufbau des Emergency Object .....	20
4.4.5 Service Daten Objects (SDO) .....	21
4.4.5.1 Aufbau eines Service Daten Object .....	21
4.4.5.2 Initiate SDO Download Protocol .....	22
4.4.5.3 Download SDO Segment Protocol .....	22
4.4.5.4 Initiate SDO Upload Protocol .....	22
4.4.5.5 Upload SDO Segment Protocol .....	22
4.4.5.6 Abort SDO Protocol .....	23
4.4.6 Process Daten Objekt (PDO) .....	23
4.4.6.1 Übertragungsmodi .....	24
4.4.6.2 Triggermodi .....	24
4.4.7 Nodeguard .....	25
4.4.8 Heartbeat .....	26
<b>5 CANopen Antriebsprofil</b> .....	<b>27</b>
5.1 Emergency Messages .....	27

5.2 Allgemeine Definitionen .....	28
5.2.1 Allgemeine Objekte .....	28
5.2.1.1 Objekt 1000h: Device Type (DS301) .....	28
5.2.1.2 Objekt 1001h: Error register (DS301) .....	28
5.2.1.3 Objekt 1002h: Manufacturer Status Register (DS301) .....	29
5.2.1.4 Objekt 1003h: Predefined Error Field (DS301) .....	30
5.2.1.5 Objekt 1005h: COB-ID of the SYNC Message (DS301) .....	31
5.2.1.6 Objekt 1006h: Communication Cycle Period (DS301) .....	31
5.2.1.7 Objekt 1008h: Manufacturer Device Name (DS301) .....	31
5.2.1.8 Objekt 1009h: Manufacturer Hardware Version .....	32
5.2.1.9 Objekt 100Ah: Manufacturer Software Version (DS301) .....	32
5.2.1.10 Objekt 100Ch: Guard Time (DS301) .....	32
5.2.1.11 Objekt 100Dh: Lifetime Factor (DS301) .....	32
5.2.1.12 Objekt 1010h: Store Parameters (DS301) .....	33
5.2.1.13 Objekt 1011h: Restore Default Parameters DS301 .....	34
5.2.1.14 Objekt 1014h: COB-ID for Emergency Message (DS301) .....	34
5.2.1.15 Objekt 1016h: Consumer Heartbeat Time .....	35
5.2.1.16 Objekt 1017h: Producer Heartbeat Time .....	35
5.2.1.17 Objekt 1018h: Identity Objekt (DS301) .....	36
5.2.1.18 Objekt 1026h: OS Prompt .....	37
5.2.1.19 Objekt 2000h: Manufacturer Warnings .....	38
5.2.1.20 Objekt 2014-2017h: 1st-4th Mask 1 to 4 for Transmit-PDO .....	39
5.2.1.21 Objekt 2030h: DP-RAM Variables (write only) .....	39
5.2.1.22 Objekt 2040h: Gearing factor for electronic gearing .....	40
5.2.1.23 Objekt 2041h: Electric gearing actual value .....	41
5.2.1.24 Objekt 2051h: Configuration of Position Registers .....	42
5.2.1.25 Objekt 2052h: Position Registers, absolute .....	43
5.2.1.26 Objekt 2053h: Positions Registers, relative .....	43
5.2.1.27 Objekt 2061h: Current limitation for velocity mode .....	44
5.2.1.28 Objekt 2080h: Motion Task Select .....	44
5.2.1.29 Objekt 2081h: Active Motion Task Display .....	44
5.2.1.30 Objekt 2082h: Copy motion tasks .....	45
5.2.1.31 Objekt 2083h: Delete Motion tasks .....	45
5.2.1.32 Objekt 2090h: DP-RAM Variables (read only) .....	45
5.2.1.33 Objekt 20A0h: Latch position 1, positive edge .....	46
5.2.1.34 Objekt 20A1h: Latch position 1, negative edge .....	46
5.2.1.35 Objekt 20A2h: Latch position 2, positive edge .....	46
5.2.1.36 Objekt 20A3h: Latch position 2, negative edge .....	47
5.2.1.37 Objekt 20A4h: Latch Control Register .....	47
5.2.1.38 Objekt 20B0h: Trigger Variable Digital Input 20 .....	47
5.2.1.39 Objekt 20B1h: Control Word Digital Inputs 5...20 .....	48
5.2.1.40 Objekt 20B2h: Analog Inputs .....	48
5.2.1.41 Objekt 2100h: Write Dummy .....	49
5.2.1.42 Objekt 2101h: Read Dummy .....	49
5.2.1.43 Objekt 60FDh: Digital inputs (DS402) .....	49
5.2.1.44 Objekt 6502h: Supported drive modes (DS402) .....	50
5.3 PDO Konfiguration .....	50
5.3.1 Receive PDOs (RXPDO) .....	51
5.3.1.1 Objekte 1400-1403h: 1st-4th RxPDO communication parameter (DS301) .....	51
5.3.1.2 Objekte 1600-1603h: 1st-4th RxPDO Mapping Parameter (DS301) .....	52
5.3.1.3 Default RXPDO Definition .....	52
5.3.2 Transmit PDOs (TXPDO) .....	53
5.3.2.1 Objekte 1800-1803h: 1st-4th TxPDO communication parameter (DS301) .....	53
5.3.2.2 Objekte 1A00-1A03h: 1st-4th TxPDO mapping parameter (DS301) .....	55

5.3.2.3 Default TXPDO definition .....	55
5.4 Device control (dc) .....	56
5.4.1 Zustandsmaschine (DS402) .....	56
5.4.1.1 Zustände der Zustandsmaschine .....	56
5.4.1.2 Übergänge der Zustandsmaschine .....	57
5.4.2 Objekt Beschreibung .....	58
5.4.2.1 Objekt 6040h: Control word (DS402) .....	58
5.4.2.2 Objekt 6041h: Status word (DS402) .....	60
5.4.2.3 Objekt 6060h: Modes of Operation (DS402) .....	61
5.4.2.4 Objekt 6061h: Modes of Operation Display (DS402) .....	62
5.5 Factor Groups (fg) (DS402) .....	62
5.5.1 Allgemeine Informationen .....	62
5.5.1.1 Faktoren .....	62
5.5.1.2 Beziehung zwischen physikalischen und antriebsinternen Einheiten .....	62
5.5.2 Objekte für Positionsberechnungen .....	63
5.5.2.1 Objekt 6089h: position notation index (DS402) .....	63
5.5.2.2 Objekt 608Ah: Position Dimension Index (DS402) .....	63
5.5.2.3 Objekt 608Fh: Position encoder resolution (DS402) .....	64
5.5.2.4 Objekt 6091h: Gear Ratio (DS402) .....	65
5.5.2.5 Objekt 6092h: Feed constant (DS402) .....	66
5.5.2.6 Objekt 6093h: Position factor (DS402) .....	67
5.5.2.7 Objekt 6094h: Velocity encoder factor (DS402) .....	68
5.5.3 Objekte für Geschwindigkeitsberechnungen .....	69
5.5.3.1 Objekt 608Bh: velocity notation index (DS402) .....	69
5.5.3.2 Objekt 608Ch: velocity dimension index (DS402) .....	69
5.5.4 Objekte für Beschleunigungsberechnungen .....	70
5.5.4.1 Objekt 608Dh: acceleration notation index (DS402) .....	70
5.5.4.2 Objekt 608Eh: acceleration dimension index (DS402) .....	70
5.5.4.3 Objekt 6097h: Acceleration factor (DS402) .....	71
5.6 Profile Velocity Mode (pv) (DS402) .....	72
5.6.1 Allgemeine Informationen .....	72
5.6.2 Objekte, die in diesem Kapitel definiert werden .....	72
5.6.3 Objekte, die in anderen Kapiteln definiert werden .....	72
5.6.4 Objektbeschreibung .....	72
5.6.4.1 Objekt 606Ch: Velocity actual value (DS402) .....	72
5.6.4.2 Objekt 60FFh: Target velocity (DS402) .....	72
5.7 Profile Torque Mode (tq) (DS402) .....	73
5.7.1 Allgemeine Informationen .....	73
5.7.2 Objekte, die in diesem Kapitel definiert werden .....	73
5.7.3 Objektbeschreibung .....	73
5.7.3.1 Objekt 6071h: Target torque (DS402) .....	73
5.7.3.2 Objekt 6073h: Max current (DS402) .....	73
5.7.3.3 Objekt 6077h: Torque actual value (DS402) .....	73
5.8 Position Control Function (pc) (DS402) .....	74
5.8.1 Allgemeine Informationen .....	74
5.8.2 Objekte, die in diesem Kapitel definiert werden .....	74
5.8.3 Objekte, die in anderen Kapiteln definiert werden .....	74
5.8.4 Objektbeschreibung .....	74
5.8.4.1 Objekt 6063h: position actual value* (DS402) .....	74
5.8.4.2 Objekt 6064h: position actual value (DS402) .....	75
5.8.4.3 Objekt 6065h: Following error window .....	75
5.8.4.4 Objekt 6067h: Position window (DS402) .....	75
5.8.4.5 Objekt 6068h: Position window time (DS402) .....	76
5.8.4.6 Objekt 60F4h: Following error actual value (DS402) .....	76

5.9 Interpolated Position Mode (ip) (DS402)	77
5.9.1 Allgemeine Informationen	77
5.9.2 Objekte, die in diesem Kapitel definiert werden	77
5.9.3 Objektbeschreibung	77
5.9.3.1 Objekt 60C0h: Interpolation sub mode select	77
5.9.3.2 Objekt 60C1h: Interpolation data record	78
5.9.3.3 Objekt 60C2h: Interpolation time period	79
5.9.3.4 Objekt 60C3h: Interpolation sync definition	80
5.9.3.5 Objekt 60C4h: Interpolation data configuration	80
5.10 Homing Mode (hm) (DS402)	82
5.10.1 Allgemeine Informationen	82
5.10.2 Objekte, die in diesem Kapitel definiert werden	82
5.10.3 Objekte, die in anderen Kapiteln definiert werden	82
5.10.4 Objektbeschreibung	82
5.10.4.1 Objekt 607Ch: Homing offset (DS402)	82
5.10.4.2 Objekt 6098h: Homing method (DS402)	83
5.10.4.3 Objekt 6099h: Homing speeds (DS402)	84
5.10.4.4 Objekt 609Ah: Homing acceleration (DS402)	84
5.10.5 Homing Mode Sequence	85
5.11 Profile Position Mode (pp)	85
5.11.1 Allgemeine Informationen	85
5.11.2 Objekte, die in diesem Kapitel definiert werden	85
5.11.3 Objekte, die in anderen Kapiteln definiert werden	85
5.11.4 Objektbeschreibung	86
5.11.4.1 Objekt 607Ah: Target position (DS402)	86
5.11.4.2 Objekt 607Dh: Software position limit (DS402)	86
5.11.4.3 Objekt 607Fh: Max profile velocity (DS402)	87
5.11.4.4 Objekt 6080h: Max motor speed (DS402)	87
5.11.4.5 Objekt 6081h: Profile velocity (DS402)	87
5.11.4.6 Objekt 6083h: Profile acceleration (DS402)	88
5.11.4.7 Objekt 6084h: Profile deceleration (DS402)	88
5.11.4.8 Objekt 6085h: Quick stop deceleration	88
5.11.4.9 Objekt 6086h: Motion Profile T type (DS402)	89
5.11.4.10 Objekt 60C5h: Max acceleration	89
5.11.5 Funktionelle Beschreibung	90
<b>6 Anhang</b>	<b>92</b>
6.1 Der Objektkanal	92
6.1.1 Objekte >3500h Herstellerspezifischer Objektkanal	92
6.1.2 ASCII Kommando Referenz	94
6.1.3 Objektverzeichnis	103
6.2 CANopen SDOs für Safety Erweiterungskarten S1/S2/S1-2/S2-2	110
6.2.1 Objekt 2400h: Safety card serial number	110
6.2.2 Objekt 2401h: Safety card status	110
6.2.3 Objekt 2402h: Safety card I/O status	111
6.2.4 Objekt 2403h: Safety card error register	112
6.2.5 Objekt 2404h: Safety card error stack error number	113
6.2.6 Objekt 2405h: Safety card error stack error time	113
6.2.7 Objekt 2406h: Safety card error stack error index	114
6.2.8 Objekt 2407h: Safety card error stack error info	114
6.2.9 Objekt 2408h: Safety card error stack error parameter 1	115
6.2.10 Objekt 2409h: Safety card error stack error parameter 2	115
6.2.11 Objekt 240Ah: Safety card error stack error parameter 3	116
6.2.12 Objekt 240Bh: Safety card error stack error parameter 4	116
6.2.13 Objekt 240Ch: Actual speed	117

---

6.3 Beispiele .....	117
6.3.1 Prinzipielle Prüfung des CAN Verbindungsaufbaus .....	117
6.3.2 Beispiel: Bedienung der Zustandsmaschine .....	118
6.3.3 Beispiel: Tipbetrieb über SDO .....	119
6.3.4 Beispiel: Drehmoment-Modus über SDO .....	119
6.3.5 Beispiel: Tipbetrieb über PDO .....	120
6.3.6 Beispiel: Drehmoment-Modus über PDO .....	121
6.3.7 Beispiel: Referenzfahrt über SDO .....	122
6.3.8 Beispiel: Starte Fahrauftrag über SDO aus dem internen Speicher des S300/S700 .....	123
6.3.9 Beispiel: Bedienung des Profile Position Mode .....	124
6.3.10 Beispiel: ASCII Kommunikation über SDO .....	127
6.3.11 Test für SYNC Telegramme .....	128
6.3.12 Anwendung: Elektrisches Getriebe .....	129
6.3.13 Anwendungen: Externe Trajektorie mit Interpolated Position Modus .....	130
<b>7 Index .....</b>	<b>135</b>

---/---



## 2 Allgemeines

### 2.1 Über dieses Handbuch

Dieses Handbuch beschreibt Inbetriebnahme, Funktionsumfang und Softwareprotokoll des Kommunikationsprofils CANopen bei Servoverstärkern der Serien S300/S700. Es ist Bestandteil der Gesamtdokumentation der digitalen Servoverstärker S300/S700.

Installation und Inbetriebnahme der Servoverstärker, sowie alle Standardfunktionen werden in der zugehörigen Betriebsanleitung beschrieben.

Sonstige Bestandteile der Gesamtdokumentation der digitalen Servoverstärker-Familien:

Titel	Herausgeber
Betriebsanleitungen S300/S700	Kollmorgen
Online-Hilfe mit Objekt Referenz in der Inbetriebnahme-Software	Kollmorgen

Weiterführende Dokumentation:

Titel	Herausgeber
CAN Application (CAL) for Industrial Applications	CiA e.V.
Draft Standards 301 (ab Version 4.0), 402	CiA e.V.
CAN Specification Version 2.0	CiA e.V.

ISO 11898 ... Controller Area Network (CAN) for high-speed communication

### 2.2 Zielgruppe

Dieses Handbuch richtet sich mit folgenden Anforderungen an Fachpersonal:

<b>Transport:</b>	Personal mit Kenntnissen in der Behandlung elektrostatisch gefährdeter Bauelemente.
<b>Auspacken:</b>	Fachleute mit elektrotechnischer Ausbildung.
<b>Installation:</b>	Fachleute mit elektrotechnischer Ausbildung.
<b>Inbetriebnahme:</b>	Fachleute mit weitreichenden Kenntnissen in den Bereichen Elektrotechnik / Antriebstechnik.
<b>Programmierung:</b>	Softwareentwickler, CAN-Bus Projektoren.

Das Fachpersonal muss IEC 60364/60664 und nationale Unfallverhütungsvorschriften beachten.

#### Nur geschultes Personal einsetzen!

Während des Betriebes der Geräte besteht die Gefahr von gesundheitlichen oder materiellen Schäden.

- Der Betreiber muss sicherstellen, dass die Sicherheitshinweise in diesem Handbuch beachtet werden.
- Der Betreiber muss sicherstellen, dass alle mit Arbeiten am Servoverstärker betrauten Personen die Betriebsanleitung gelesen und verstanden haben.

Wir bieten auf Anfrage Schulungs- und Einarbeitungskurse an.

### 2.3 Hinweise für die Online-Ausgabe (PDF-Format)

Das Dokument bietet verschiedene Funktionen, um die Navigation zu vereinfachen.

<b>Lesezeichen</b>	Das Inhaltsverzeichnis und der Index enthalten aktive Lesezeichen.
<b>Inhaltsverzeichnis und Index im Text</b>	Die Zeilen im Inhaltsverzeichnis und Index sind aktive Querverweise. Klicken Sie auf eine Zeile, um zur entsprechenden Seite zu gelangen.
<b>Seitennummern im Text</b>	Seitennummern im Text mit Querverweisen sind aktive Verknüpfungen.









## 2.4 Bestimmungsgemäße Verwendung

Beachten Sie die Kapitel "Bestimmungsgemäße Verwendung" und "Nicht bestimmungsgemäße Verwendung" in der Betriebsanleitung des Servoverstärkers.

Das CANopen Interface dient allein dem Anschluss des Servoverstärkers an einen Master mit CANopen Anbindung (CAN-Bus). Die Servoverstärker werden als Komponenten in elektrischen Anlagen oder Maschinen eingebaut und dürfen nur als integrierte Komponenten der Anlage in Betrieb genommen werden.

Wir garantieren nur bei Verwendung der von uns genannten Komponenten und Einhaltung der Installationsvorschriften die Konformität der Servoverstärker zu den in der EU Konformitäts-erklärung genannten europäischen Richtlinien.

## 2.5 Verwendete Symbole

Symbol	Bedeutung
	Weist auf eine gefährliche Situation hin, die, wenn sie nicht vermieden wird, zum Tode oder zu schweren, irreversiblen Verletzungen führen wird.
	Weist auf eine gefährliche Situation hin, die, wenn sie nicht vermieden wird, zum Tode oder zu schweren, irreversiblen Verletzungen führen kann.
	Weist auf eine gefährliche Situation hin, die, wenn sie nicht vermieden wird, zu leichten Verletzungen führen kann.
	Dieses Symbol weist auf eine Situation hin, die, wenn sie nicht vermieden wird, zu Beschädigung von Sachen führen kann.
	Dieses Symbol weist auf wichtige Informationen hin.
	Warnung vor einer Gefahr (allgemein). Die Art der Gefahr wird durch den nebenstehenden Warntext spezifiziert.
	Warnung vor gefährlicher elektrischer Spannung und deren Wirkung.
	Warnung vor Gefahr durch automatischem Anlauf.

## 2.6 Verwendete Abkürzungen

Kürzel	Bedeutung
BTB/RTO	Betriebsbereit
COB	Communication Object
COB-ID	Communication Object Identifier
EEPROM	Elektrisch löschbarer Festspeicher
EMV	Elektromagnetische Verträglichkeit
ISO	International Standardization Organisation
LED	Leuchtdiode
MB	MegaByte
NSTOP	Endschalteneingang Drehrichtung links
PC	Personal Computer
PDO	Prozessdatenobjekt
PSTOP	Endschalteneingang Drehrichtung rechts
RAM	flüchtiger Speicher
ROD	Inkrementelle Positionsangabe
RXPDO	Receive- (Empfangs-) PDO
SDO	Servicadatenobjekt
TXPDO	Transmit- (Sende-) PDO

## 2.7 Realisierte Grundfunktionen über CANopen

In Zusammenhang mit dem im digitalen Servoverstärker S300/S700 integrierten Lageregler werden folgende Funktionen bereitgestellt:

### Einricht- und Allgemeine Funktionen:

- Referenzfahren, Referenzpunkt setzen
- Vorgabe digitaler Sollwerte für die Drehzahl und Momentenregelung
- Unterstützung folgender Betriebsarten des CANopen Profils DS402:
  - profile position mode
  - homing mode
  - profile torque mode
  - interpolated position mode
  - profile velocity mode

### Positionierfunktionen:

- Ausführen eines Fahrauftrages aus dem Fahrsatzspeicher des Servoverstärkers
- Ausführen eines Direktfahrauftrages
- Trajektorie absolut, ip-Mode

### Datentransferfunktionen:

- Übertragen eines Fahrauftrages in den Fahrsatzspeicher des Servoverstärkers Ein Fahrauftrag besteht aus folgenden Elementen::
  - Positionssollwert (Absolutauftrag) oder Wegsollwert (Relativauftrag)
  - Geschwindigkeitssollwert
  - Beschleunigungszeit, Bremszeit
  - Fahrauftragsart (absolut/relativ)
  - Nummer eines Folgefahrauftrags (mit oder ohne Zwischenstopp)
- Lesen eines Fahrauftrages aus dem Fahrsatzspeicher des Servoverstärkers
- Lesen von Istwerten
- Lesen der Fehlerregister
- Lesen der Statusregister
- Lesen / Schreiben von Regelparametern

## 2.8 Übertragungsgeschwindigkeit und -verfahren

- Busanpassung und Busmedium: CAN-Standard ISO 11898 (CAN-Highspeed)
- Übertragungsgeschwindigkeit: max. 1MBit/s Einstellmöglichkeiten des Servoverstärkers: 10, 20, 50, 100, 125, 250, 333, 500 (default), 666, 800, 1000 kBaud

## 2.9 Verhalten bei Kommunikationsstörungen BUSOFF

Die Kommunikationsstörung BUSOFF wird direkt von der Schicht 2 (CAN controller) überwacht und gemeldet. Diese Meldung kann unterschiedliche Ursachen haben.

Hier einige Beispiele:

- Telegramm wird gesendet, obwohl kein weiterer CAN – Knoten angeschlossen ist
- CAN Knoten weisen unterschiedliche Baudraten auf
- Busleitung defekt
- Reflexionen auf den Leitungen aufgrund fehlerhafter Leitungsabschlüsse.

Ein BUSOFF wird vom S300/S700 nur gemeldet, wenn ein weiterer CAN Knoten angeschlossen ist und mindestens ein Objekt zu Beginn erfolgreich abgesetzt werden konnte. Der Zustand BUSOFF wird mit der Fehlermeldung F23 signalisiert. Sollte beim Auftreten dieses Fehlers die Endstufe freigegeben (enable) sein und eine Fahrfunktion ausgeführt werden, wird der Antrieb mit der Notbremsrampe angehalten und die Endstufe gesperrt (disable).

## 3 Installation / Inbetriebnahme

### 3.1 Wichtige Hinweise



#### **GEFAHR** Hohe Spannung bis 900 V!

Lebensgefahr durch elektrischen Schlag. Restladungen in den Kondensatoren können auch mehrere Minuten nach Abschalten der Netzspannung gefährliche Werte aufweisen. Steuer- und Leistungsanschlüsse können Spannung führen, auch wenn sich der Motor nicht dreht.

- Installieren und verdrahten Sie nur abgeschaltete Geräte.
- Achten Sie darauf, dass die Anlage sicher freigeschaltet ist (Absperrung, Warnzeichen usw.). Erst bei der Inbetriebnahme werden die einzelnen Spannungen eingeschaltet.
- Messen Sie zur Sicherheit die Spannung am DC-Bus-Zwischenkreis, und warten Sie, bis die Spannung unter 50 V gesunken ist.



#### **WARNUNG** Automatischer Wiederanlauf!

Es besteht die Gefahr von tödlichen oder schweren Verletzungen für Personen, die in der Maschine arbeiten. Antriebe mit Servoverstärkern in Netzwerken sind fernbediente Maschinen. Sie können sich jederzeit ohne vorherige Ankündigung in Bewegung setzen.

- Stellen Sie durch entsprechende Schutzmaßnahmen sicher, dass ein ungewolltes Anlaufen der Maschine nicht zu Gefahrensituationen für Mensch und Maschine führen kann.
- Der Anwender ist dafür verantwortlich, dass bei Ausfall des Servoverstärkers der Antrieb in einen funktional sicheren Zustand geführt wird, z.B. mit einer sicheren mechanischen Bremse.
- Softwareendschalter ersetzen nicht die Hardwareendschalter der Maschine.

#### **ACHTUNG**

Montieren Sie den Servoverstärker wie in den Betriebsanleitungen S300/S700 beschrieben. Beachten Sie alle Sicherheitshinweise in der zum Servoverstärker gehörenden Betriebsanleitung. Beachten Sie alle Hinweise zu Einbaulage, Umgebungsbedingungen und Verdrahtung, sowie Absicherung.

### 3.2 Einstellen der Stationsadresse und der Übertragungsrage

Bei der Inbetriebnahme ist es sinnvoll, die Stationsadressen der einzelnen Verstärker und die Baudrate für die Kommunikation vorab über die Frontplattentastatur einzustellen (siehe Kapitel "Inbetriebnahme" in der Betriebsanleitung)

#### **INFO**

Nach Verändern der Stationsadresse und Baudrate müssen Sie die 24V-Hilfsspannung der Servoverstärker ausschalten und wieder einschalten.

Die **Stationsadresse** (Bereich 1...127) können Sie auf drei Arten einstellen:

- Mit der Tastatur in der Frontplatte (siehe Kapitel "Inbetriebnahme" in der Betriebsanleitung)
- In der Inbetriebnahme-Software DRIVEGUI.EXE auf der Bildschirmseite "CAN / Feldbus"
- Über die serielle Schnittstelle mit der Abfolge der ASCII-Kommandos:  
ADDR nn → SAVE → COLDSTART (mit nn = Adresse).

Die **Übertragungsgeschwindigkeit** (Baudrate) können Sie auf drei Arten einstellen:

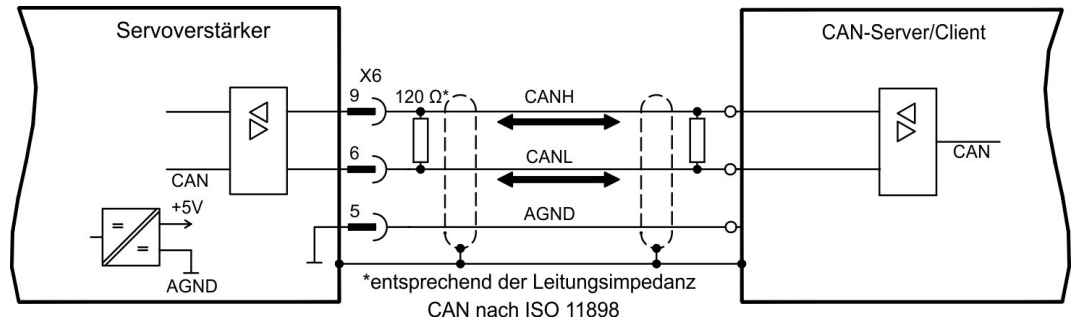
- Mit der Tastatur in der Frontplatte (siehe Kapitel "Inbetriebnahme" in der Betriebsanleitung)
- In der Inbetriebnahme-Software DRIVEGUI.EXE auf der Bildschirmseite "CAN / Feldbus"
- Über die serielle Schnittstelle mit der Abfolge der ASCII-Kommandos:  
CBAUD bb → SAVE → COLDSTART (mit bb = Baudrate in kbit/s).

Kodierung der Baudrate im LED-Display :

Kodierung	Baud Rate in kbit/s	Kodierung	Baud Rate in kbit/s
1	10	33	333
2	20	50	500
5	50	66	666
10	100	80	800
12	125	100	1000
25	250		

### 3.3 CAN-Bus Interface (X6)

Interface zum Anschluss an den CAN Bus (default : 500 kBaud). Die Schnittstelle liegt auf dem gleichen Potential wie das RS232-Interface. Die analogen Sollwerteingänge sind weiterhin nutzbar. Wir liefern auf Anfrage spezielle Klemmbuchsenstecker, die für den Busbetrieb leicht konfektioniert werden können.



### 3.4 CAN Busleitung

Nach ISO 11898 sollten Sie eine Busleitung mit einem Wellenwiderstand von 120Ω verwenden. Die verwendbare Leitungslänge für eine sichere Kommunikation nimmt mit zunehmender Übertragungsrate ab. Als Anhaltspunkte können folgende bei uns gemessenen Werte dienen, die allerdings nicht als Grenzwerte zu verstehen sind:

#### Leistungsdaten:

- Wellenwiderstand: 100-120 Ω
- Leitungskapazität: max. 60 nF/km
- Leiterwiderstand (Schleife): 159,8 Ω/km

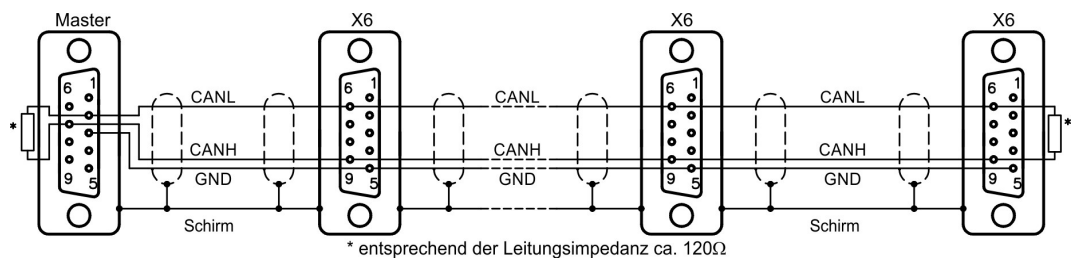
#### Leitungslängen in Abhängigkeit von der Übertragungsrate

Übertragungsrate (kBaud)	max. Leitungslänge (m)
1000	10
500	70
250	115

Mit geringerer Leitungskapazität (max. 30 nF/km) und geringerem Leiterwiderstand (Schleife, 115 Ω/km) können größere Übertragungsweiten erreicht werden. (Wellenwiderstand  $150 \pm 5\Omega$  → Abschlusswiderstand  $150 \pm 5\Omega$ ).

An das SubD-Steckergehäuse sind aus EMV-Gründen folgende Anforderung gestellt:

- Metallisches oder metallisch beschichtetes Gehäuse.
- Anschlussmöglichkeit für den Leitungsschirm im Gehäuse, großflächige Verbindung.



### 3.5 Leitfaden zur Inbetriebnahme

#### ACHTUNG

Nur Fachpersonal mit fundierten Kenntnissen in Regelungstechnik und Antriebstechnik darf den Servoverstärker in Betrieb nehmen.



#### **! WARNUNG** Automatischer Wiederanlauf!

Es besteht die Gefahr von tödlichen oder schweren Verletzungen für Personen, die in der Maschine arbeiten. Während der Inbetriebnahme ist nicht auszuschließen, dass der Antrieb ungeplant eine Bewegung durchführt.

Stellen Sie sicher, dass auch bei ungewollter Bewegung des Antriebs keine Gefährdung von Personen oder Sachen eintreten kann. Die Maßnahmen, die Sie dazu in Ihrer Anwendung treffen müssen, ergeben sich aus der Risikobeurteilung der Anwendung.

<b>Montage / Installation prüfen</b>	Prüfen Sie, ob alle Sicherheitshinweise in der Betriebsanleitung des Servoverstärkers und im vorliegenden Handbuch beachtet und umgesetzt wurden. Kontrollieren Sie Adresse und Baudrate.
<b>PC anschließen, Inbetriebnahmesoftware starten</b>	Zum Parametrieren des Servoverstärkers verwenden Sie die Inbetriebnahmesoftware DRIVEGUI.EXE.
<b>Grundfunktionen in Betrieb nehmen</b>	Nehmen Sie nun die Grundfunktionen des Servoverstärkers in Betrieb und optimieren Sie Strom- und Drehzahlregler. Dieser Teil der Inbetriebnahme ist in der Betriebsanleitung und in der Onlinehilfe der Inbetriebnahmesoftware genauer beschrieben.
<b>Parameter speichern</b>	Speichern Sie die Parameter nach erfolgter Optimierung im Servoverstärker.
<b>Buskommunikation in Betrieb nehmen</b>	Die geänderten Parameter werden erst nach einem Software-Reset (Warmboot) wirksam. Betätigen Sie dazu den Reset-Button in der Symbolleiste der Inbetriebnahmesoftware. Es ist erforderlich, dass das ab Seite (→ # 27) beschriebene Software-Protokoll ist auf dem Master realisiert ist. Baudrate des S300/S700 an die des Masters anpassen.
<b>Test der Kommunikation</b>	Prüfen Sie die Boot-Up Meldung, wenn Sie den Verstärker einschalten. Führen Sie einen SDO Lesezugriff auf Index 1000 Subindex 0 aus (Gerätetyp). Siehe auch Beispiele im Anhang (→ # 117).
<b>Lageregler in Betrieb nehmen</b>	Nehmen Sie den Lageregler in Betrieb, wie in der Online-Hilfe der Inbetriebnahmesoftware beschrieben.

### 3.6 Wichtige Konfigurationsparameter für den CAN-Bus-Betrieb

Die folgenden Parameter sind für den CAN-Betrieb wichtig:

1. **CBAUD** : Übertragungsrate für den CAN-Bus
2. **ADDR** : Mit dem Kommando ADDR wird die Feldbus-Adresse des Verstärkers definiert. Nach der Änderung der Adresse sollten alle Parameter im EEPROM abgespeichert werden und der Verstärker ausund eingeschaltet werden.
3. **AENA**: Hiermit kann der Zustand des Software-Enable beim Einschalten des Verstärkers definiert werden. Mit dem Software-Enable wird einer externen Steuerung die Möglichkeit gegeben die Endstufe softwaremässig zu sperren bzw. freizugeben. Bei Geräten, die mit einem analogen Sollwert arbeiten (OPMODE=1,3) wird beim Einschalten des Verstärkers das Software-Enable automatisch gesetzt, so dass diese Geräte sofort betriebsbereit sind (Hardware-Enable vorausgesetzt). Bei allen anderen Geräten wird beim Einschalten das Software-Enable auf den Wert von AENA gesetzt. Die Variable AENA hat auch eine Funktion beim Reset des Verstärkers nach einem Fehler (über digitalen Eingang 1 bzw. mit dem ASCII-Kommando CLRFAULT). Bei Fehlern, die mit der Software zurückgesetzt werden können, wird, nachdem der Fehler gelöscht wurde, der Software-Enable auf den Zustand von AENA gesetzt. Auf diese Weise ist das Verhalten des Verstärkers beim Software-Reset analog zu dem Einschaltverhalten.

## 4 CANopen Kommunikationsprofil

Dieses Kapitel beschreibt die grundlegenden Dienste und Kommunikationsobjekte des CANopen Kommunikationsprofils DS 301, die vom S300/S700 verwendet werden.

### INFO

Es wird vorausgesetzt, dass das Kommunikationsprofil in seiner grundlegenden Funktionsweise bekannt ist und als Referenzdokumentation zur Verfügung steht.

Da die Geschäftssprache der CiA englisch ist, werden feststehende Begriffe im Original verwendet (z.B. Objekt 1000h: Device Type).

### 4.1 Allgemeine Erläuterungen zu CAN

Das hier verwendete Übertragungsverfahren ist in der ISO 11898 (Controller Area Network [CAN] for high-speed communication) definiert.

Das in allen CAN-Bausteinen implementierte Schicht 1/2-Protokoll (Physical Layer/Daten Link Layer) stellt u. a. die Anforderung von Daten zur Verfügung.

Datentransport bzw. Datenanforderung erfolgen über ein Datentelegramm (Daten Frame) mit bis zu 8 Byte Nutzdaten bzw. über ein Datenanforderungstelegramm (Remote Frame).

Kommunikationsobjekte (COBs) werden durch einen 11 Bit Identifier (ID) gekennzeichnet, der auch die Priorität von Objekten bestimmt.

Um die Applikation von der Kommunikation zu entkoppeln, wurde ein Schicht-7-Protokoll (Anwendungsschicht) entwickelt. Die von der Anwendungsschicht bereitgestellten Dienstelemente ermöglichen die Realisierung einer über das Netzwerk verteilten Applikation. Diese Dienstelemente sind im "CAN Applikation Layer (CAL) for Industrial Applications" beschrieben.

Auf die CAL ist das Kommunikationsprofil CANopen und das Antriebsprofil aufgesetzt. Das folgende Diagramm zeigt die grundlegende Struktur eines Kommunikationsobjekts:

S		R				A	
O	COB-ID	T	CTRL	Daten Segment	CRC	C	EOM
M		R				K	

SOM	Start of message
COB-ID	Communication Object Identifier (11-bit)
RTR	Remote Transmission Request
CTRL	Control Field (u.a. Daten Length Code)
Daten Segment	0 ... 8 Byte (Daten-COB) 0 Byte (Remote-COB)
CRC	Zyklische Prüfsequenz
ACK	Acknowledge slot
EOM	End of message

## 4.2 Aufbau des Kommunikationsobjekt Identifiers

Die folgende Grafik zeigt den Aufbau des COB – Identifiers (COB-ID). Der "Function Code" legt die Bedeutung und die Priorität des jeweiligen Objekts fest.

10	9	8	7	6	5	4	3	2	1	0
Function-Code					Module-ID					

Bit 0 .. 6:

Modul ID (CAN-Bus Adresse des Servoverstärkers, Bereich 1..127; wird in der Inbetriebnahme-Software oder am Servoverstärker eingestellt, (→ # 13).

Bit 7... 10:

Function Code (Nummer des im Server definierten Kommunikationsobjektes)

### INFO

Wird eine ungültige Stationsnummer eingestellt (=0 oder >127), so wird die Modul-ID intern auf 1 gesetzt.

Die folgenden Tabellen zeigen die Default-Belegung der COB-Identifier nach dem Einschalten des Servoverstärkers. Die Objekte, die mit einem Index (Communication Parameters at Index) versehen sind, können nach der Initialisierungsphase mit einem neuen Identifier versehen werden. Die Indizes in Klammern sind optional.

#### Vordefinierte "Broadcast" Objekte (Senden an alle)

Objekt	Function code (binär)	Resultierende COB-IDs		Kommunikation Parameter bei Index
		Dez.	Hex.	
NMT	0000	0	0h	—
SYNC	0001	128	80h	(1005h)
TIME	0010	256	100h	nicht unterstützt

#### Vordefinierte "Peer to Peer" Objekte (Senden Station zu Station)

Objekt	Function code (binär)	Resultierende COB-IDs		Kommunikation Parameter bei Index	Priority	
		Dez.	Hex.			
EMERGENCY	0001	129..255	81h..FFh	—	high	
TPDO 1	0011	385..511	181h..1FFh	1800h	↓	
RPDO 1	0100	513..639	201h..27Fh	1400h		
TPDO 2	0101	641..767	281h..2FFh	1801h		
RPDO 2	0110	769..895	301h..37Fh	1401h		
TPDO 3	0110	897..1023	381h..3FFh	1802h		
RPDO 3	1000	1025..1151	401h..47Fh	1402h		
TPDO 4	1001	1153..1279	481h..4FFh	1803h		
RPDO 4	1010	1281..1407	501h..57Fh	1403h		
SDO (tx*)	1011	1409..1535	581h..5FFh			
SDO (rx*)	1100	1537..1663	601h..67Fh			
Nodeguard	1110	1793..1919	701h..77Fh	(100Eh)		low

\*tx = Senderichtung : S300/S700 → Master

rx = Senderichtung : Master → S300/S700



### 4.3 Definition der verwendeten Datentypen

Dieses Kapitel definiert die verwendeten Datentypen. Jeder Datentyp kann mit Hilfe von Bit-Sequenzen beschrieben werden. Diese Bit-Sequenzen werden in "Octets" (Bytes) zusammengefasst. Für numerische Datentypen wird das "Little-Endian"-Format (Intel) verwendet (siehe auch DS301 Application Layer "General Description of Datentyps and Encoding Rules").

#### 4.3.1 Basisdatentypen

##### 4.3.1.1 Unsigned Integer

Daten vom Basis Datentyp UNSIGNEDn definiert ausschließlich positive Integer. Der Wertebereich ist  $0 \dots 2^n - 1$ . Die Bitsequenz  $b = b_0 \dots b_{n-1}$  definiert den Wert  $UNSIGNEDn(b) = b_{n-1} 2^{n-1} + \dots + b_1 2^1 + b_0 2^0$

Beispiel: Der Wert 266 = 10Ah wird mit dem Datentyp UNSIGNED16 mit Hilfe von zwei "octets" übertragen (1. Octet = 0Ah, 2. Octet = 01h).

##### Übertragungssyntax für den Datentyp UNSIGNEDn

Octet Nummer	1.	2.	3.	4.	5.	6.	7.	8.
UNSIGNED8	b <sub>7</sub> ..b <sub>0</sub>							
UNSIGNED16	b <sub>7</sub> ..b <sub>0</sub>	b <sub>15</sub> ..b <sub>8</sub>						
UNSIGNED24	b <sub>7</sub> ..b <sub>0</sub>	b <sub>15</sub> ..b <sub>8</sub>	b <sub>23</sub> ..b <sub>16</sub>					
UNSIGNED32	b <sub>7</sub> ..b <sub>0</sub>	b <sub>15</sub> ..b <sub>8</sub>	b <sub>23</sub> ..b <sub>16</sub>	b <sub>31</sub> ..b <sub>24</sub>				
UNSIGNED40	b <sub>7</sub> ..b <sub>0</sub>	b <sub>15</sub> ..b <sub>8</sub>	b <sub>23</sub> ..b <sub>16</sub>	b <sub>31</sub> ..b <sub>24</sub>	b <sub>39</sub> ..b <sub>32</sub>			
UNSIGNED48	b <sub>7</sub> ..b <sub>0</sub>	b <sub>15</sub> ..b <sub>8</sub>	b <sub>23</sub> ..b <sub>16</sub>	b <sub>31</sub> ..b <sub>24</sub>	b <sub>39</sub> ..b <sub>32</sub>	b <sub>47</sub> ..b <sub>40</sub>		
UNSIGNED56	b <sub>7</sub> ..b <sub>0</sub>	b <sub>15</sub> ..b <sub>8</sub>	b <sub>23</sub> ..b <sub>16</sub>	b <sub>31</sub> ..b <sub>24</sub>	b <sub>39</sub> ..b <sub>32</sub>	b <sub>47</sub> ..b <sub>40</sub>	b <sub>55</sub> ..b <sub>48</sub>	
UNSIGNED64	b <sub>7</sub> ..b <sub>0</sub>	b <sub>15</sub> ..b <sub>8</sub>	b <sub>23</sub> ..b <sub>16</sub>	b <sub>31</sub> ..b <sub>24</sub>	b <sub>39</sub> ..b <sub>32</sub>	b <sub>47</sub> ..b <sub>40</sub>	b <sub>55</sub> ..b <sub>48</sub>	b <sub>63</sub> ..b <sub>56</sub>

##### 4.3.1.2 Signed Integer

Daten vom Basis Datentyp INTEGERn definiert positive und negative Integer. Der Wertebereich ist  $-2^{n-1} \dots 2^{n-1} - 1$ . Die Bitsequenz  $b = b_0 \dots b_{n-1}$  definiert den Wert  $INTEGERn(b) = b_{n-2} 2^{n-2} + \dots + b_1 2^1 + b_0 2^0$  mit  $b_{n-1} = 0$

Negative Zahlen werden im zweier Komplement dargestellt. Somit gilt:

$INTEGERn(b) = -INTEGERn(b)-1$  mit  $b_{n-1} = 1$

Beispiel: Der Wert -266 = FEF6h wird mit dem Datentyp INTEGER16 mit Hilfe von zwei "Octets" übertragen (1. Octet = F6h, 2. Octet = FEh).

##### Übertragungssyntax für den Datentyp INTEGERn

Octet Nummer	1.	2.	3.	4.	5.	6.	7.	8.
INTEGER8	b <sub>7</sub> ..b <sub>0</sub>							
INTEGER16	b <sub>7</sub> ..b <sub>0</sub>	b <sub>15</sub> ..b <sub>8</sub>						
INTEGER24	b <sub>7</sub> ..b <sub>0</sub>	b <sub>15</sub> ..b <sub>8</sub>	b <sub>23</sub> ..b <sub>16</sub>					
INTEGER32	b <sub>7</sub> ..b <sub>0</sub>	b <sub>15</sub> ..b <sub>8</sub>	b <sub>23</sub> ..b <sub>16</sub>	b <sub>31</sub> ..b <sub>24</sub>				
INTEGER40	b <sub>7</sub> ..b <sub>0</sub>	b <sub>15</sub> ..b <sub>8</sub>	b <sub>23</sub> ..b <sub>16</sub>	b <sub>31</sub> ..b <sub>24</sub>	b <sub>39</sub> ..b <sub>32</sub>			
INTEGER48	b <sub>7</sub> ..b <sub>0</sub>	b <sub>15</sub> ..b <sub>8</sub>	b <sub>23</sub> ..b <sub>16</sub>	b <sub>31</sub> ..b <sub>24</sub>	b <sub>39</sub> ..b <sub>32</sub>	b <sub>47</sub> ..b <sub>40</sub>		
INTEGER56	b <sub>7</sub> ..b <sub>0</sub>	b <sub>15</sub> ..b <sub>8</sub>	b <sub>23</sub> ..b <sub>16</sub>	b <sub>31</sub> ..b <sub>24</sub>	b <sub>39</sub> ..b <sub>32</sub>	b <sub>47</sub> ..b <sub>40</sub>	b <sub>55</sub> ..b <sub>48</sub>	
INTEGER64	b <sub>7</sub> ..b <sub>0</sub>	b <sub>15</sub> ..b <sub>8</sub>	b <sub>23</sub> ..b <sub>16</sub>	b <sub>31</sub> ..b <sub>24</sub>	b <sub>39</sub> ..b <sub>32</sub>	b <sub>47</sub> ..b <sub>40</sub>	b <sub>55</sub> ..b <sub>48</sub>	b <sub>63</sub> ..b <sub>56</sub>

#### 4.3.2 Zusammengesetzte Datentypen

Zusammengesetzte Datentypen nutzen Basisdatentypen (INTEGERn, UNSIGNEDn, REAL). Es werden zwei Datentypen unterschieden:

- STRUCT  
Dieser Datentyp setzt sich aus unterschiedlichen Basisdatentypen zusammen.
- ARRAY  
Dieser Datentyp setzt sich aus gleichen Basisdatentypen zusammen.

### 4.3.3 Erweiterte Datentypen

Erweiterte Datentypen werden aus Basisdatentypen und zusammengesetzten Datentypen abgeleitet. Im Folgenden werden die unterstützten Datentypen definiert.

#### 4.3.3.1 Octet String

Der Datentyp "OCTET\_STRING" definiert sich aus dem Datentyp ARRAY. "Length" ist die Länge des "octet string".

ARRAY[length] OF UNSIGNED8 = OCTET\_STRINGlength

#### 4.3.3.2 Visible String

Der Datentyp "VISIBLE\_" kann sich aus dem Datentyp UNSIGNED8 oder aus dem Datentyp ARRAY definieren. Die zulässigen Werte sind 00h und der Bereich von 20h bis 7Eh.

Die Daten werden als 7 Bit ASCII Code interpretiert (ISO 646-1973(E)). "Length" ist die Länge des "visible string".

UNSIGNED8 = VISIBLE\_CHAR

ARRAY[length] OF VISIBLE\_CHAR = VISIBLE\_STRINGlength

## 4.4 Kommunikationsobjekte

Die Kommunikationsobjekte werden mit Hilfe von Dienstelementen und Protokollen beschrieben. Es können im wesentlichen zwei Dienstelemente unterschieden werden:

- Unbestätigte Dienste PDO
- Bestätigte Dienste SDO

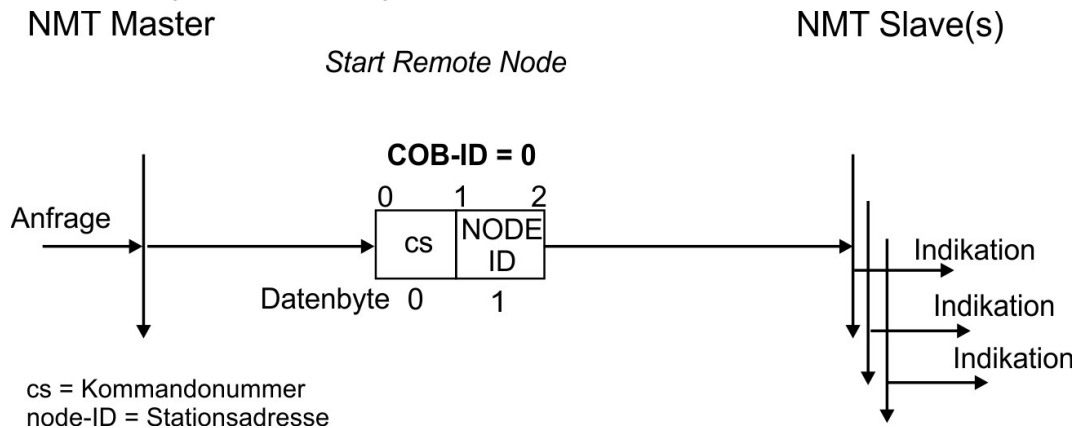
Sämtliche Dienste setzen voraus, das "Daten Link" und "Physical Layer" fehlerfrei arbeiten.

Der S300/S700 unterstützt die Kommunikationsobjekte, die in den folgenden Kapiteln detailliert beschrieben werden:

- Network Management Objects (NMT)
- Synchronization Objekt (SYNC)
- Emergency Objekt (EMCY)
- Process Daten Objekt (PDO)
- Service Daten Objekt (SDO)
- Nodeguard

#### 4.4.1 Network Management Objects (NMT)

Das NMT-Telegramm sieht wie folgt aus:



Der Antrieb unterstützt folgende Funktionen des Netzwerkmanagements:

- |  |  |
|--|--|
| <b>cs = 129, reset node:</b>               | führt zu einem Kaltstart des Antriebs, damit werden alle im RAM abgelegte Parameter gelöscht und wieder die im EEPROM gespeicherten Werte eingestellt.   |
| <b>cs = 130, reset communication node:</b> | stoppt die PDO Kommunikation, erzeugt eine neue Boot-up Meldung.   |
| <b>cs = 1, start remote node:</b>          | startet den CAN-Knoten, d.h. gibt die PDOs des Antriebs zur Bedienung frei. Ab diesem Zeitpunkt werden Sende-PDOs ereignisgesteuert gesendet, bzw. kann der zyklische Prozessdatenbetrieb starten. |
| <b>cs = 2, stop remote node:</b>           | stoppt den CAN-Knoten, d.h. der Antrieb reagiert nicht mehr auf empfangene PDOs und sendet keine mehr.   |

#### 4.4.2 Synchronization Object (SYNC)

Das SYNC Objekt wird meist als periodisches "Broadcast" – Objekt verwendet und gibt den Basis-Bustakt vor. Um einen zeitlich äquidistanten Abstand zu ermöglichen, besitzt das SYNC eine hohe Priorität. Die Anwendung dieses Protokolls wird im Anhang (→ # 117) beschrieben. Mit Hilfe des SYNC-Objekt können z.B. auch Fahraufträge mehrerer Achsen gleichzeitig gestartet werden.

#### 4.4.3 Time-Stamp Objekt (TIME)

Dieses Kommunikationsobjekt wird von S300/S700 nicht unterstützt.

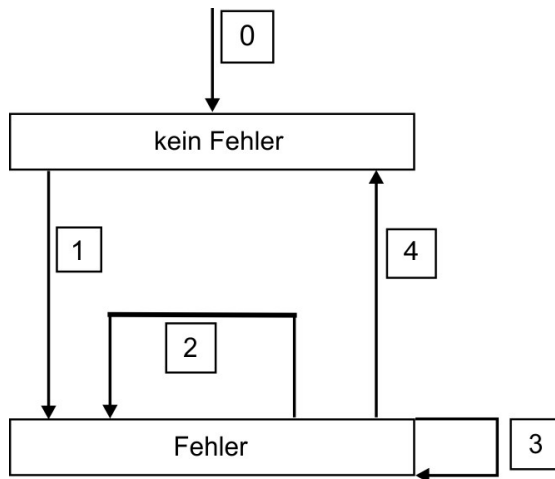
#### 4.4.4 Emergency Object (EMCY)

Das EMCY wird ereignisgetriggert aufgrund einer internen Fehlersituation generiert. Für jeden Fehler wird dieses Objekt erneut übertragen. Da die Fehlercodes geräteabhängig sind, werden sie im Kapitel "CANopen Antriebsprofil" (→ # 27) beschrieben. Die letzten 8 Fehlercodes können über Objekt 1003 gelesen werden.

#### 4.4.4.1 Verwendung des Emergency Object

Das Verhalten im Fehlerfall ist von der Fehlerart abhängig und somit unterschiedlich. Aus diesem Grund wird das Verhalten mit Hilfe eine Fehlerzustandsmaschine beschrieben. Es werden die Fehlerzustände "error free" und "error occurred" unterschieden. Es sind die folgenden Übergänge definiert:

0. Nach der Initialisierung wird der Zustand "error free" eingenommen, falls kein Fehler erkannt wurde. In diesem Zustand wird keine Fehlermeldung generiert.
1. Der S300/S700 erkennt einen internen Fehler und zeigt diesen in den ersten drei Bytes des "emergency" Telegramms an ("error code" Byte 0,1 und "error register" Byte 2). Da der S300/S700 mehrere Fehlerarten unterscheidet, wird Byte 3 aus dem herstellerspezifischen Fehlerfeld dazu genutzt, um die Fehlerkategorie anzuzeigen.
2. Einer, aber nicht alle Fehler sind zurückgesetzt worden. Das EMCY Telegramm beinhaltet den "Error Code" 0000h, das "Error Register" zeigt die restlichen noch anstehenden Fehler an. Der herstellerspezifische Bereich ist zu Null gesetzt.
3. Ein neuer Fehler ist aufgetreten. Der S300/S700 verbleibt im Zustand "Error State" und überträgt ein EMCY Objekt mit dem entsprechenden "Error Code". Der neue "Error Code" wird in den Bytes 0 und 1 eingetragen.
4. Alle Fehler sind zurückgesetzt worden. Das EMCY Telegramm beinhaltet den "Error Code" 0000h, das "Error Register" zeigt keine Fehler an. Der herstellerspezifische Bereich ist zu Null gesetzt.



#### 4.4.4.2 Aufbau des Emergency Object

Das Emergency-Objekt setzt sich aus 8 Byte zusammen und ist folgendermaßen aufgeteilt:

Byte	0	1	2	3	4	5	6	7
Inhalt	Emergency error code (→ # 27)		Error register (Objekt 1001h)	Kategorie	Reserviert			

Wurde ein Emergency Objekt generiert, wird im Anschluss daran der Fehlerzustand der Zustandsmaschine gemeldet (error free / error occurred), indem ein zweites Emergency-Objekt generiert wird. Hier sind nur die ersten 4 Bytes relevant (Emergency ErrorCode, Error register, Kategorie). Byte 0/1 enthält den "Error Reset Code" (0000h) und Byte 2 zeigt an, dass eventuell noch ein Fehler ansteht. Ist das Error Register 00h, ist der Fehlerzustand "error free".

Byte 3 enthält die Kategorie. Die Bedeutung der Fehlernummern (Error Code) und der Fehlerkategorie werden im Kapitel "Emergency Message" beschrieben (→ # 27). Das Fehlerregister wird über das Objekt 1001h "Error register" definiert.

#### 4.4.5 Service Daten Objects (SDO)

Mit Hilfe der SDOs wird der Zugriff auf das Objektverzeichnis realisiert. Die SDOs werden zur Parametrierung und zur Statusabfrage verwendet. Der Zugriff auf ein einzelnes Objekt erfolgt mit einem Multiplexer über Index und Subindex des Objektverzeichnisses. Es werden die folgenden Kommunikationsprotokolle unterstützt:

- Initiate SDO Download Protocol
- Download SDO Segment Protocol
- Initiate SDO Upload Protocol
- Upload SDO Segment Protocol
- Abort SDO Transfer Protocol

Die Definition der einzelnen Kommunikationsdienste und der Protokolle sind dem DS301 zu entnehmen. Beispiele zur Handhabung von SDOs sind im Anhang ab Seite (→ # 117).

#### INFO

Da es sich bei einem SDO um einen bestätigten Dienst handelt, muss immer auf das SDO-Antworttelegramm gewartet werden, bevor ein neues Telegramm gesendet werden darf.

##### 4.4.5.1 Aufbau eines Service Daten Object

Ein SDO setzt sich aus folgenden Komponenten zusammen:

Byte	1	2	3	4	5	6	7	8
Inhalt	rw	Index		Subindex		Daten		

##### 1. Das KontrollByte (Byte 1):

Mit Hilfe des KontrollBytes wird bestimmt, ob über das SDO schreibend oder lesend auf den Objektverzeichniseintrag zugegriffen wird. Eine Darstellung des gesamten Objektverzeichnis finden Sie ab Seite (→ # 103).

Der Datenaustausch mit dem S300/S700 hält sich an den Standard "CMS multiplexed domain protocols", wie er im CAN-Standard DS 202 beschrieben wird. Um Daten zu lesen, muss das KontrollByte entsprechend folgender Darstellung beschrieben werden:

Bit	7	6	5	4	3	2	1	0
Inhalt	ccs*=2			X	X	X	X	X

\*ccs → client command specifier (ccs = 2 → initiate upload request)

X → beliebig

Der Wert 0100 0000 (binary) oder 40h muss im Control-Byte übertragen werden. Der Serververstärker antwortet entsprechend mit einem AntwortByte:

Bit	7	6	5	4	3	2	1	0
Inhalt	scs*=2			X	n		e	s

\*scs → server command specifier (scs = 2 → initiate upload response)

n → nur gültig bei e = s = 1.

Ist dies der Fall, so steht in n die Zahl der Bytes, die keine Daten enthalten.

X → beliebig

Wenn der Lesezugriff erfolgreich war, sind im AntwortByte immer die Bits 0 und 1 auf 1 gesetzt (e=s=1). Kodierte Byte-Länge in der SDO Antwort:

0x43-4 Bytes

0x47-3 Bytes

0x4B-2 Bytes

0x4F-1 Byte.

Wenn ein Fehler auftritt, wird scs auf 4 gesetzt, das Antwort-Byte ist 0x80 und die Fehlerinformation ist im 4-Byte Datenfeld enthalten. Aufschlüsselung des Fehlers: (→ # 23).

Um Daten zu schreiben, muss das KontrollByte wie folgt beschrieben werden:

Client Byte	Initiate Domain Download								Server								
	7	6	5	4	3	2	1	0		2	3	4	5	6	7	8	
request	cos = 1			X	n	e	s		m				d				indication
	==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==> ==>																

n,e und s sind wie im Lesefall definiert., m: Index + Subindex, d: Datenfeld mit 4 Bytes Die Datenlänge eines Objekts finden Sie im Objekt-Verzeichnis im Anhang.

Das Control Byte soll sein:

0x23 für 4-Byte Zugriff

0x27 für 3-Byte Zugriff

0x2B für 2-Byte Zugriff

0x2F für 1-Byte Zugriff

Client Byte	Initiate Domain Download								Server								
	7	6	5	4	3	2	1	0		2	3	4	5	6	7	8	
confirm	scs = 3			X					m				reserviert				response
	<== <== <== <== <== <== <== <== <== <== <== <== <== <== <==																

### 2. Der Index (Bytes 2 und 3):

Der Index ist der Haupteintrag im Objektverzeichnis und teilt die Parameter in Gruppen ein (Beispiel: Index 1018h ist das Identitäts Objekt).

Wie bei allen CAN -Daten wird der Index in umgekehrter Reihenfolge in den Bytes abgelegt, Beispiel: Index 6040h → Byte 2 = 40h, Byte 3 = 60h).

### 3. Der Subindex (Byte 4):

Der Subindex unterteilt die Parameter innerhalb einer Parametergruppe.

### 4. Das Datenfeld (Bytes 5 ... 8):

Über diese Komponenten werden die eigentlichen Nutzdaten ausgetauscht. In den Telegrammen zur Leseaufforderung an den S300/S700 sind diese auf 0 zu setzen, in der Schreibbestätigung vom S300/S700 haben diese Daten bei erfolgreichem Transfer keinen Inhalt, bei fehlerhaftem Schreiben enthalten sie einen Fehlercode (→ # 23).

#### 4.4.5.2 Initiate SDO Download Protocol

Das "Initiate SDO Download" – Protokoll wird für Schreibzugriffe auf Objekte mit bis zu 4 Byte Nutzdaten verwendet ("expedited transfer") oder zum Einleiten eines Segmenttransfers ("normal transfer").

Der "expedited transfer" wird auch für Objekte genutzt, die nur eine Kommandoeigenschaft aufweisen (z.B. ASCII: SAVE) und somit keine weiteren Nutzdaten benötigen.

#### 4.4.5.3 Download SDO Segment Protocol

Das "Download SDO Segment" – Protokoll wird für Schreibzugriffe auf Objekte mit mehr als 4 Byte Nutzdaten verwendet ("normal transfer"). Dieser Dienst wird vom S300/S700 z.Zt. nicht unterstützt, da keine Objekte existieren, die mehr als 4 Byte Nutzdaten verarbeiten.

#### 4.4.5.4 Initiate SDO Upload Protocol

Das "Initiate SDO Upload" – Protokoll wird für Lesezugriffe auf Objekte mit bis zu 4 Byte Nutzdaten ("expedited transfer") oder zum Einleiten eines Segmenttransfers ("normal transfer") verwendet .

#### 4.4.5.5 Upload SDO Segment Protocol

Das "Upload SDO Segment" – Protokoll wird für Lesezugriffe auf Objekte mit mehr als 4 Byte Nutzdaten verwendet ("normal transfer"). Dieser Dienst wird vom S300/S700 z.Zt. nicht unterstützt, da keine Objekte existieren, die mehr als 4 Byte Nutzdaten verarbeiten.

#### 4.4.5.6 Abort SDO Protocol

Das "Abort SDO" Protokoll wird bei einer fehlerhaften SDO-Übertragung vom S300/S700 verwendet und zeigt mit Hilfe des "abort code" (Fehlercode) den Fehler an, der zum Abbruch der Übertragung führte. Der Fehlercode ist als UNSIGNED32-Wert kodiert. Die folgende Tabelle zeigt die möglichen Ursachen für einen Abbruch:

Abort Code	Beschreibung
0601 0000h	Zugriff auf dieses Objekt wird nicht unterstützt
0601 0001h	Versuchter Lesezugriff auf ein Objekt, das nur geschrieben werden kann
0601 0002h	Versuchter Schreibzugriff auf ein Objekt, das nur gelesen werden kann
0602 0000h	Objekt existiert im Objektverzeichnis nicht
0604 0041h	Objekt kann nicht in ein PDO "gemappt" werden
0604 0042h	Größe und Anzahl der "gemappten" Objekte übersteigt die mögliche PDO Länge
0604 0043h	Allgemeine Parameter-Inkompatibilität
0607 0010h	Datentyp passt nicht, Länge des Service Parameters passt nicht
0609 0011h	Subindex existiert nicht
0609 0030h	Wertebereich des Parameters verlassen (nur für Schreibzugriffe)
0609 0031h	Wert des Parameters zu groß
0609 0032h	Wert des Parameters zu klein
0800 0020h	Datum kann nicht übertragen oder gespeichert werden
0800 0022h	Datum kann aufgrund des Gerätezustandes nicht übertragen / gespeichert werden
FF03 0000h	OS Kommandospeicher voll

Nicht aufgeführte "Abort Codes" sind reserviert.

#### 4.4.6 Process Daten Objekt (PDO)

PDOs werden für die Echtzeit-Datenkommunikation verwendet. Zum Beispiel lassen sich über PDOs ähnliche Regler aufbauen wie sie bei analog realisierten Antrieben verwendet werden. Statt der dort verwendeten +/- 10V Sollwertvorgabe und ROD-Istwertausgabe können hier digitaler Drehzahlsollwert und Istpositionsausgabe über PDOs realisiert werden. Die PDO-Daten werden dazu ohne Protokoll-Overhead übertragen und der Empfang nicht bestätigt.

Dieses Kommunikationsobjekt nutzt einen unbestätigten Kommunikationsdienst.

PDOs definieren sich über das Objektverzeichnis des S300/S700. Das "Mapping" erfolgt mit Hilfe von SDOs während der Konfigurationsphase. Die Länge wird über die gemappten Objekte definiert.

Die Definition des PDO Dienstes und des Protokolls sind dem DS301 zu entnehmen. Beispiele zur Handhabung von PDOs sind im Anhang ab Seite (→ # 92) zu finden.

Grundsätzlich können zwei Arten von PDOs in Abhängigkeit der Übertragungsrichtung unterschieden werden:

- Transmit-PDOs (TPDOs) (S300/S700 → Master)  
Die TPDOs übertragen Daten vom S300/S700 zur Steuerung (z.B. Istwertobjekte, Gerätezustände).
- Receive-PDOs (RPDOs) (Master → S300/S700)  
Die RPDOs übertragen Daten von der Steuerung zum S300/S700 (z.B. Sollwerte)

Der S300/S700 unterstützt für jede Übertragungsrichtung vier unabhängige PDO-Kanäle. Die Kanäle sind mit den Kanalnummern 1 bis 4 gekennzeichnet.

Zur Konfiguration der vier möglichen PDOs stehen jeweils zwei Parametersätze zur Verfügung, die über entsprechende SDOs eingestellt werden können:

1. Mappingparameter, über die ermittelt werden kann, welche Daten im ausgewählten PDO vorhanden (= gemappt) sind und über die bestimmt werden kann, welche Daten enthalten sein sollen.
2. Kommunikationsparameter, die festlegen, ob die PDOs im synchronisierten Betrieb oder ereignisgesteuert arbeiten sollen: (Objekte 1400h bis 1403h, 1800h bis 1803h).

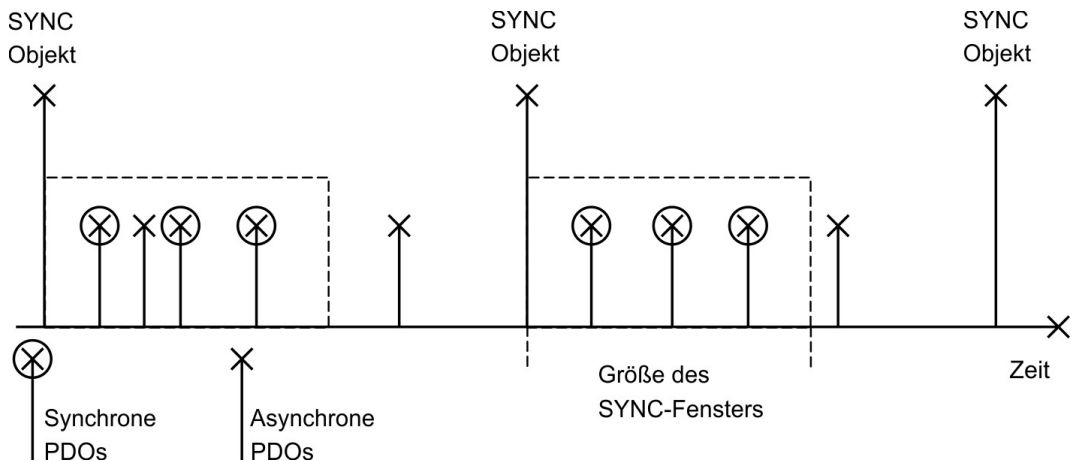
#### 4.4.6.1 Übertragungsmodi

Es werden die folgenden PDO Übertragungsmodi (transmission mode) unterschieden:

- Synchroner Übertragung
- Asynchroner Übertragung

Um Antriebe zu synchronisieren, wird periodisch das vordefinierte SYNC – Objekt übertragen (Bustakt). Synchroner PDOs werden innerhalb eines vordefinierten Zeitfensters im Anschluss an das SYNC Objekt übertragen.

Die Einstellung der Übertragungsmodi wird mit Hilfe der PDO Kommunikationsparameter vorgenommen.



#### 4.4.6.2 Triggermodi

Es werden drei Triggermodi (Triggering Modes) unterschieden:

- **Event driven**  
Die Übertragung der Telegramme wird über ein objektspezifisches Ereignis getriggert.
- **Time driven**  
Bei ereignisgesteuerten Signalen, die eine zu hohe Buslast erzeugen würden, kann mit Hilfe der "inhibit time" (Kommunikationsparameter, Subindex 03h) bestimmt werden, nach welcher Mindestzeit wieder ein PDO gesendet werden darf.
- **Event Timer driven**  
Wenn ein PDO (auch ohne Änderung) während eines bestimmten Zeitintervalls gesendet werden soll, kann dieses Intervall mit einem speziellen SDO definiert werden.



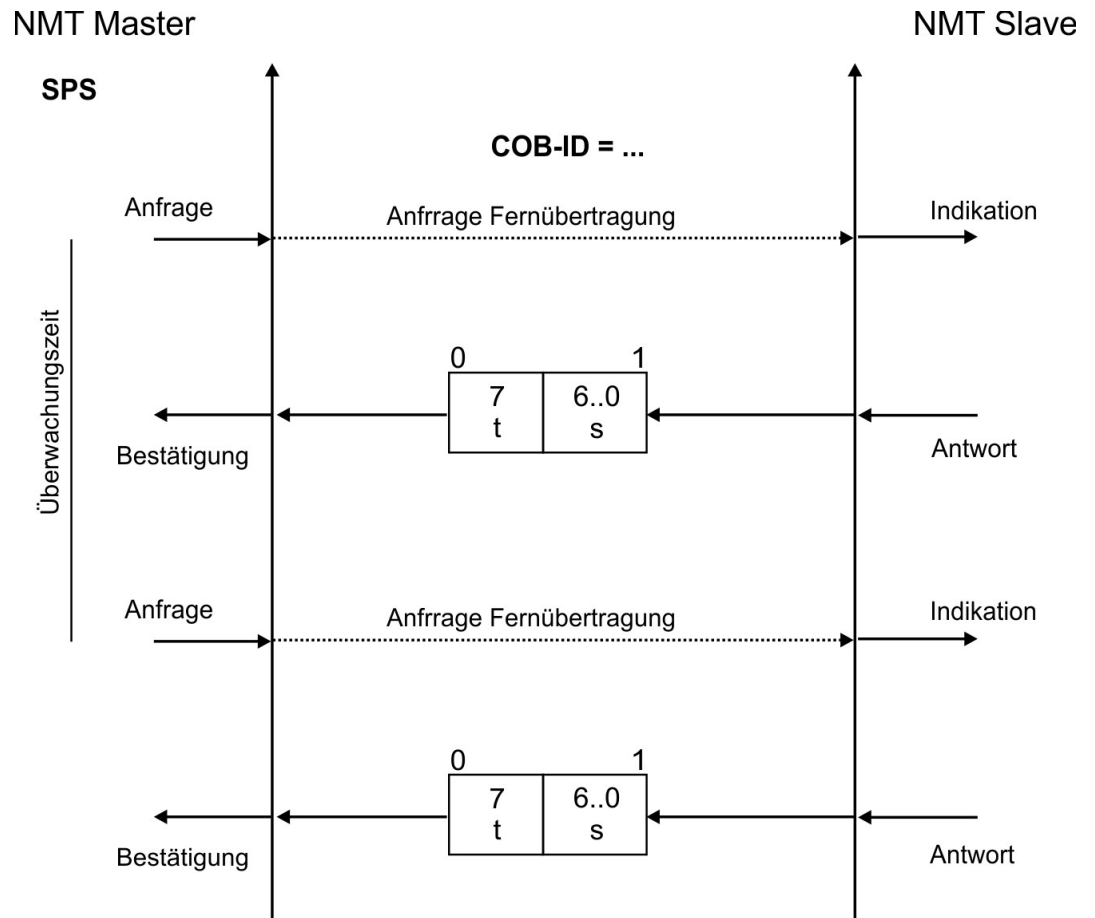
#### 4.4.7 Nodeguard

Das Node Guarding Protokoll dient der Funktionsüberwachung des Antriebs. Dazu wird der Antrieb in äquidistanten Zeitabständen vom CANopen-Master angesprochen.

Der maximale zeitliche Abstand, der zwischen zwei Nodeguard-Telegrammen liegen darf, wird durch das Produkt der Guard Time (Objekt 100Ch (→ # 32)) und des Life Time Factors (Objekt 100Dh, (→ # 32)) bestimmt. Ist einer dieser beiden Werte 0, ist die Ansprechüberwachung deaktiviert.

Wird der Antrieb innerhalb der Zeit, die durch die Objekte 100Ch und 100Dh definiert ist, nicht angesprochen, tritt die Warnung N04 (Ansprechüberwachung) am Antrieb auf, der Antrieb wird mittels Quickstop-Rampe abgebremst und jede weitere Bewegung wird verhindert (Parameter DECSTOP, Objekt 6085 sub0 (→ # 88)).

Die zeitliche Folge des Nodeguardings sieht wie folgt aus:



t = toggle Bit, ändert seinen Zustand mit jedem Slave-Telegramm

s = Zustand der NMT Slave-Zustandsmaschine

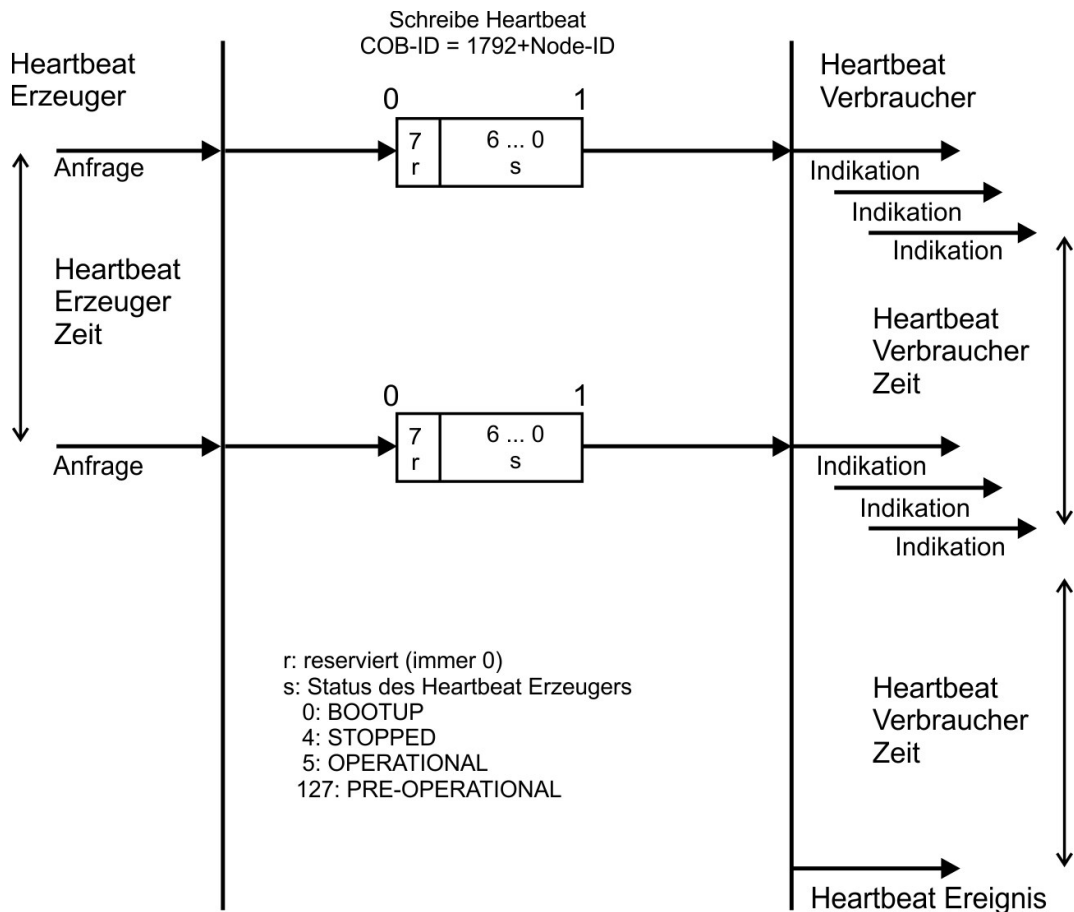
Das Nodeguarding wird vom Master mit RTR-Telegrammen mit dem COB-ID 700h + Slave-Knotenadresse ausgeführt.

### 4.4.8 Heartbeat

Das Heartbeat Protokoll definiert einen Fehlerüberwachungsdienst ohne Remote Frame. Ein Heartbeat Producer überträgt zyklisch eine Heartbeat Nachricht. Einer oder mehrere Heartbeat Consumer empfangen die Meldung. Die Beziehung zwischen Producer und Consumer ist mit Hilfe von Objekt 1016h (→ # 35) / 1017h (→ # 35) konfigurierbar.

Der Heartbeat Consumer überwacht den Empfang des Heartbeat während der Heartbeat Consumer Time. Wenn der Heartbeat während dieser Zeit nicht empfangen wird, wird ein Heartbeat Event erzeugt.

Heartbeat Protokoll:



## 5 CANopen Antriebsprofil

### 5.1 Emergency Messages

"Emergency Messages" werden durch interne Gerätefehler ausgelöst. Sie haben eine hohe ID – Priorität, um einen schnellen Buszugriff sicherzustellen. Die "Emergency Message" beinhaltet ein Fehlerfeld mit vordefinierten Fehlernummern (2 Byte), Fehlerregister (1Byte) und die Fehlerkategorie (1Byte) und zusätzlichen Informationen (→ # 15). Das höherwertige Byte der Fehlernummer gibt die Fehlerklasse und das niederwertige Byte die Fehlernummer der Klasse an.

Fehlernummern von 0000h bis 7FFFh sind im Kommunikations- oder Antriebsprofil definiert. Fehlernummern von FF00h bis FFFFh sind herstellerspezifische Definitionen. Mit Hilfe der Fehlerkategorie können auftretende Fehler in ihrer Bedeutung klassifiziert werden. Es sind die folgenden Fehlerkategorien definiert:

1. Fehler, die nur durch einen Reset (Kommando "COLDSTART" oder Bit 7 im controlword (→ # 58) gelöscht werden können. Wird ein Reset bei einem Fehler dieser Kategorie ausgeführt, wird ein Coldstart des Geräts ausgeführt.
2. Fehler, die durch Bit 7 im controlword (→ # 58).
3. Fehlermeldungen, die bei der Verarbeitung eines PDOs auftreten können.
4. Fehler, die **nicht** durch den Anwender behoben werden können.
5. Bedienungsfehler / Warnungen

Die Fehler der Kategorien 1 und 2 werden durch Blinken der Anzeige in der Frontplatte angezeigt

Error Code	Kategorie	Fehlernummer	Beschreibung
0000h	—	—	ErrorReset oder kein Fehler (erforderlich)
1000h	—	—	Generischer Fehler (erforderlich)
1080h	5	—	Kein BTB vorhanden (Zustand "not ready for operation")
2330h	2	F22	Fehler im Erdanschluss
2380h	1	F12	Fehler im Motor Anschluss (Phasenfehler)
3100h	2	F16	Kein Netz-BTB
3110h	2	F02	Überspannung Zwischenkreis
3120h	2	F05	Unterspannung Zwischenkreis
3130h	2	F19	Eine Netzphase fehlt (bei PMODE = 2)
4110h	2	F13	Umgebungstemperatur überschritten
4210h	1	F01	Kühlkörpertemperatur überschritten
4310h	1	F06	Motortemperatur überschritten
5111h	1	F07	Fehler ±15VHilfsspannung
5380h	1	F17	Fehler A/D-Konverter
5400h	1	F14	Endstufenfehler
5420h	1	F18	Ballast (Chopper)
5441h	1	F27	Fehler bei Bedienung der AS-Option
5530h	1	F09	Seriellles EEPROM
6320h	3	—	Parameterfehler
7111h	1	F11	Bremsenfehler
7122h	1	F25	Kommutierungsfehler
7181h	5	—	S300/S700 konnte nicht enabled werden
7303h	1	F04	Fehler Rückführeinheit
7305h	1	F10	Signalfehler digitaler Encoder Eingang
8182h	1	F23	CAN-Busoff
8331h	2	F15	I2t (torque fault)
8480h	2	F08	Überdrehzahl (overspeed)
8611h	2	n03/F03	Schleppfehler
8681h	5	—	Ungültige Fahrauftragsnummer
FF01h	1	F32	Schwerwiegender Ausnahmefehler
FF02h	3	—	Fehler in PDO-Komponente
FF04h	1	F20	Slotfehler
FF05h	1	1	Handlingfehler
FF06h	2	F24	Warnungsanzeige als Fehler
FF07h	2	F26	Fehler Referenzfahrt (HW-Endschalter erreicht)
FF08h	2	F29	Sercos-Fehler
FF11h	2	F30	Emergency Timeout Ausfall

## 5.2 Allgemeine Definitionen

In diesem Kapitel werden allgemeingültige Objekte beschrieben (z.B. Objekt 1000h "Device Type"). Im Anschluss daran wird die freie Konfiguration von Prozessdatenobjekten ("Freies Mapping") erläutert.

### 5.2.1 Allgemeine Objekte

#### 5.2.1.1 Objekt 1000h: Device Type (DS301)

Dieses Objekt beschreibt den Gerätetyp (Servoantrieb) und die Gerätefunktionalität (DS402 Antriebsprofil). Es setzt sich folgendermaßen zusammen:

MSB				LSB			
zusätzliche Informationen				Geräte-Profil-Nummer			
Modus Bits		Typ		402d=192h			
31	24	23	16	15	0		

Die Geräteprofilnummer ist DS402, der Typ ist 2 für Servoverstärker, die Betriebsart-Bits 28 bis 31 sind herstellerspezifisch und können von aktuellem Wert auf 0 geändert werden. Lesezugriff liefert zur Zeit 0x00002192.

Index	<b>1000h</b>
Name	device type
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	nein

#### 5.2.1.2 Objekt 1001h: Error register (DS301)

Fehlerregister für das Gerät. Das Gerät kann interne Fehler in dieses Register eintragen. Dieses Objekt ist Teil des Fehlerobjekts (Emergency Message).

Index	<b>1001h</b>
Name	Error register
Objektcode	VAR
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED8
Defaultwert	nein

Im Folgenden wird die Bitbelegung des Fehlerregisters beschrieben. Ein gesetztes Bit zeigt ein Fehlerereignis an. Bit 0 wird in jedem Fehlerfall gesetzt.

Bit	Beschreibung	Bit	Beschreibung
0	generic error	4	communication error (overrun, error state)
1	current	5	device profile specific
2	voltage	6	reserviert (always 0)
3	temperature	7	herstellerspezifisch

### 5.2.1.3 Objekt 1002h: Manufacturer Status Register (DS301)

Das herstellerspezifische Statusregister enthält wichtige Verstärker-Informationen.

Index	<b>1002h</b>
Name	Manufacturer Status Register
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	UNSIGNED32
Defaultwert	nein

Die folgende Tabelle zeigt die Bitbelegung des Statusregisters:

Bit	Beschreibung	Bit	Beschreibung
0	1 = Bewegung aktiv (Positionierung, Referenzfahrt)	16	1 = Referenzfahrt aktiv
1	1 = Referenzpunkt gesetzt	17	1 = Tippbetrieb aktiv
2	1 = Referenzschalter betätigt (Home Position)	18	1 = Position Latch an Eingang 2 (negativer Übergang)
3	1 = In Position	19	1 = Notstopp aktiv
4	1 = Position Latch an Eingang 2 (positiver Übergang)	20	1 = Position Latch an Eingang 1 (positiver Übergang)
5	reserviert	21	1 = Position Latch an Eingang 1 (negativer Übergang)
6	reserviert	22	1 = Feed forward ausgeschaltet
7	reserviert	23	1 = Referenzfahrt beendet
8	reserviert	24	1 = ein aktueller Fehler wird beim Reset einen Kaltstart auslösen
9	reserviert	25	1 = Digitaler Eingang 1 gesetzt
10	1 = Initialisierungsphase beendet	26	1 = Digitaler Eingang 2 gesetzt
11	reserviert	27	1 = Digitaler Eingang 3 gesetzt
12	1 = Motorstillstands-Meldung (threshold VELO)	28	1 = Digitaler Eingang 4 gesetzt
13	1 = Sicherheitsrelais angewählt (AS)	29	1 = Digitaler Eingang Hardware Enable gesetzt
14	1 = Endstufe freigegeben	30	reserviert
15	1 = Fehlerstatus	31	reserviert

#### 5.2.1.4 Objekt 1003h: Predefined Error Field (DS301)

Objekt 1003h liefert eine Fehlerhistorie mit maximal 8 Einträgen.

Subindex 0 beinhaltet die Anzahl der aufgetretenen Fehler seit dem letzten Reset der Fehlerhistorie (entweder beim Start des Verstärkers oder durch Schreiben einer 0 in Subindex 0).

Eine neue Fehlermeldung wird in Subindex 1 geschrieben, der alte Eintrag wird dabei in den nächsthöheren Index geschoben. Der Inhalt von Subindex 8 geht verloren.

Die UNSIGNED32 Information, die in die Sub-Indizes geschrieben wird, ist im Felder Error Code in der Tabelle bei der Beschreibung der Emergency Messages (→ # 27) definiert.

Index	<b>1003h</b>
Name	pre-defined Error Field
Objektcode	ARRAY
Datentyp	UNSIGNED32
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	0 bis 8
Defaultwert	0
Subindex	<b>1 bis 8</b>
Beschreibung	Standard Fehlerfeld (→ # 27)
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	nein

**5.2.1.5 Objekt 1005h: COB-ID of the SYNC Message (DS301)**

Dieses Objekt definiert die COB-ID des SYNC-Objekts.

Index	<b>1005h</b>
Name	COB-ID of the SYNC message
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	konditional
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	nein

Bit kodierte Information:

Bit	Wert	Bedeutung
31 (MSB)	X	—
30	0	Gerät erzeugt keine SYNC-Message
	1	Gerät erzeugt SYNC-Message
29	0	11 Bit ID (CAN 2.0A)
	1	29 Bit ID (CAN 2.0B)
28 to 11	X	wenn Bit 29=1 => Bit 11 ... 28 der 29-bit SYNC COB-ID
	0	wenn Bit 29=0
10 to 0 (LSB)	X	Bit 0..10 von SYNC COB-ID

Das Gerät unterstützt nicht die Erzeugung von SYNC-Nachrichten und nur die 11-Bit-IDs. Die Bits 11 bis 30 sind also immer 0.

**5.2.1.6 Objekt 1006h: Communication Cycle Period (DS301)**

Mit diesem Objekt kann die Sendeperiode des SYNC-Telegramms in µs festgelegt werden.

Index	<b>1006h</b>
Name	Period of the communication cycle
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	Optional
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	00h

**5.2.1.7 Objekt 1008h: Manufacturer Device Name (DS301)**

Der Gerätenamen besteht aus vier ASCII-Zeichen und hat die Buchstaben "S3xx" zum Inhalt, wobei xx für die Stromstärke der Endstufe steht.

Index	<b>1008h</b>
Name	Manufacturer Device Name
Objektcode	VAR
Datentyp	Visible String
Kategorie	Optional
Zugriff	const
PDO Mapping	nicht möglich
Wertebereich	S301-S3xx
Defaultwert	nein

**5.2.1.8 Objekt 1009h: Manufacturer Hardware Version**

Die Hardware Version des Servoverstärkers wird angezeigt.

Index	<b>1009h</b>
Name	manufacturer hardware version
Objektcode	VAR
Datentyp	Visible String
Kategorie	Optional
Zugriff	const
PDO Mapping	nicht möglich
Wertebereich	-
Defaultwert	nein

**5.2.1.9 Objekt 100Ah: Manufacturer Software Version (DS301)**

Das Objekt beinhaltet die Version der Herstellersoftware (hier: CANopen Teil der Verstärker-Firmware)

Index	<b>100Ah</b>
Name	Manufacturer Software Version
Objektcode	VAR
Datentyp	Visible String
Kategorie	Optional
Zugriff	const
PDO Mapping	nicht möglich
Wertebereich	0,01 bis 9,99
Defaultwert	nein

**5.2.1.10 Objekt 100Ch: Guard Time (DS301)**

Das Produkt der Objekte 100Ch "Guard Time" und 100Dh "Life Time Factor" ergibt die Ansprechüberwachungszeit. Die "Guard Time" wird in Millisekunden angegeben. Die Ansprechüberwachung wird erst mit dem ersten "Nodeguard-Objekt" aktiv (→ # 25). Wird der Wert des Objekts "Guard Time" zu Null gesetzt, ist die Ansprechüberwachung inaktiv.

Index	<b>100Ch</b>
Name	Guard Time
Objektcode	VAR
Datentyp	UNSIGNED16
Kategorie	konditional; obligatorisch, wenn Heartbeat nicht unterstützt wird
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED16
Defaultwert	0

**5.2.1.11 Objekt 100Dh: Lifetime Factor (DS301)**

Das Produkt aus Guard Time und Life Time Faktor ergibt die Lebenszeit für das NodeGuarding Protokoll. Wenn dies 0 ergibt, wird das Protokoll nicht verwendet.

Index	<b>100Dh</b>
Name	Lifetime Factor
Objektcode	VAR
Datentyp	UNSIGNED8
Kategorie	konditional; (obligatorisch, wenn Heartbeat nicht unterstützt wird)
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED8
Defaultwert	0



### 5.2.1.12 Objekt 1010h: Store Parameters (DS301)

Mit Hilfe dieses Objekts können die Parameter in einem Flash-EEPROM im Servoverstärker abgespeichert werden. Nur Subindex 1 wird unterstützt (alle Parameter sichern, ähnlich wie das Sichern als Datei mit Hilfe der Inbetriebnahmesoftware).

Index	<b>1010h</b>
Name	store parameters
Objektcode	ARRAY
Datentyp	UNSIGNED32
Kategorie	optional
Subindex	<b>0</b>
Name	Anzahl Einträge
Objektcode	VAR
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	1
Defaultwert	1
Subindex	<b>1</b>
Name	save all parameters
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	1

Datendefinition:

Bit	Wert	Bedeutung
31 to 2	0	reserviert (=0)
1	0	Gerät speichert Parameter nicht selbständig
	1	Gerät speichert Parameter selbständig
0	0	Gerät speichert Parameter nicht auf Befehl
	1	Gerät speichert Parameter auf Befehl

Beim Lesezugriff aus Subindex 1 liefert der Servoverstärker Informationen über seine Speicherfunktionalität.

Der S300/S700 liefert einen konstanten Wert 1 beim Lesezugriff. Das bedeutet, alle Parameter können durch Schreiben auf Objekt1010 sub 1 gespeichert werden. Normalerweise speichert der Verstärker die Parameter nicht selbständig, spezielle Ausnahmesituation ist z.B. Homing von Multiturn Absolut-Encodern.

Die Parameter werden nur gespeichert, wenn eine spezielle Zeichenkette ("save") in Subindex 1 geschrieben wird. "save" entspricht der Unsigned32-Zahl 65766173h.

**5.2.1.13 Objekt 1011h: Restore Default Parameters DS301**

Mit diesem Objekt werden die Defaultwerte der Parameter bezogen auf die Kommunikation oder das Geräteprofil wiederhergestellt. Der S300/S700 ermöglicht die Wiederherstellung aller Defaultwerte.

Index	<b>1011h</b>
Name	restore default parameters
Objektcode	ARRAY
Datentyp	UNSIGNED32
Kategorie	optional

Subindex	<b>0</b>
Name	Anzahl Einträge
Objektcode	VAR
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	1
Defaultwert	1

Subindex	<b>1</b>
Name	restore all default parameters
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	1 (Gerät stellt Parameter wieder her)

Die Wiederherstellung wird nur ausgeführt, wenn eine spezielle Signatur ("load") in Subindex 1 geschrieben wird. "load" muss als Unsigned32-Zahl 64616F6Ch übertragen werden.

**5.2.1.14 Objekt 1014h: COB-ID for Emergency Message (DS301)**

Dieses Objekt definiert die COB-ID der Emergency Message.

Index	<b>1014h</b>
Name	COB-ID emergency message
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	konditional; obligatorisch, wenn Emergency unterstützt wird
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	80h + Node-ID

**5.2.1.15 Objekt 1016h: Consumer Heartbeat Time**

Die Consumer Heartbeat Time definiert die erwartete Heartbeat-Zykluszeit und muss größer sein als die zugehörige "Producer Heartbeat Time", die auf dem Gerät konfiguriert wurde, die sie erzeugt. Die Überwachung startet nach Empfang des ersten Heartbeat. Ist die Consumer Heartbeat Time gleich 0, wird der Eintrag nicht benutzt. Die Zeit ist definiert in ms.

Index	<b>1016h</b>
Name	consumer heartbeat time
Objektcode	ARRAY
Datentyp	UNSIGNED32
Kategorie	optional

Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	1
Defaultwert	1

Subindex	<b>1</b>
Beschreibung	Consumer heartbeat time
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	unsigned 32
Defaultwert	0

Definition der Werte in Subindex 1

	MSB			LSB		
Wert	reserviert (Wert: 00)		Node-ID		Heartbeat Zeit	
Kodiert als	-		UNSIGNED8		UNSIGNED16	
Bit	31	24	23	16	15	0

**5.2.1.16 Objekt 1017h: Producer Heartbeat Time**

Die Producer Heartbeat Time definiert die Zykluszeit des Heartbeat in ms. Bei 0 wird sie nicht verwendet.

Index	<b>1017h</b>
Name	Producer heartbeat time
Objektcode	VAR
Datentyp	UNSIGNED16
Kategorie	konditional; obligatorisch, wenn Guarding nicht unterstützt wird
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED16
Defaultwert	0

**5.2.1.17 Objekt 1018h: Identity Objekt (DS301)**

Das Identitätsobjekt beinhaltet allgemeine Geräteinformationen.

Index	<b>1018h</b>
Name	Identity Objekt
Objektcode	RECORD
Datentyp	Identity
Kategorie	obligatorisch

Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	1 to 4
Defaultwert	4

Subindex 1 ist eine eindeutige Nummer für einen Gerätehersteller.

Subindex	<b>1</b>
Beschreibung	Vendor ID
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	0x6Ah (Kollmorgen)

Subindex 2 enthält die Gerätebezeichnung mit Info über DC-Bus Spannung und Stromklasse.

Subindex	<b>2</b>
Beschreibung	Product Code
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	301 ... 346
Defaultwert	nein

Subindex 3 besteht aus zwei Revisionsnummern:

- die Haupt-Revisionsnummer im oberen Wort enthält die CAN-Version
- die Neben-Revisionsnummer enthält die Firmware-Version

Beispiel: der Wert 0x0022 0079 bedeutet: CAN-Version 0.34 und Firmware-Version 1.21.

Subindex	<b>3</b>
Beschreibung	Revision Number
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	nein

Subindex 4 enthält die Seriennummer des Gerätes.

Subindex	<b>4</b>
Beschreibung	Serial Number
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	nein

**5.2.1.18 Objekt 1026h: OS Prompt**

Der OS Prompt wird für den Aufbau eines ASCII Kommunikationskanals zum Verstärker benutzt.

Index	<b>1026h</b>
Name	OS Prompt
Objektcode	ARRAY
Datentyp	UNSIGNED8
Kategorie	optional

Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	2
Defaultwert	2

Subindex 1 wird für das Senden eines Zeichens an den Verstärker verwendet.

Subindex	<b>1</b>
Beschreibung	StdIn
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	möglich
Wertebereich	UNSIGNED8
Defaultwert	—

Subindex 2 wird für den Empfang eines Zeichens vom Verstärker verwendet.

Subindex	<b>2</b>
Beschreibung	StdOut
Kategorie	obligatorisch
Zugriff	w
PDO Mapping	möglich
Wertebereich	UNSIGNED8
Defaultwert	0

**5.2.1.19 Objekt 2000h: Manufacturer Warnings**

Dieses Objekt liefert Informationen über geräteinterne Warnungen.

Index	2000h
Name	Manufacturer Warnings
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	UNSIGNED32
Defaultwert	-

Bit-kodierte Warnungen:

Bit	Bedeutung
0	n1: I <sup>t</sup> -Meldeschwelle überschritten
1	n2: Ballastleistung erreicht
2	n3: Schleppfehler
3	n4: Ansprechüberwachung aktiv
4	n5: Netzphase fehlt
5	n6: Software-Endschalter 1 hat angesprochen
6	n7: Software-Endschalter 2 hat angesprochen
7	n8: Fehlerhafter Fahrauftrag gestartet
8	n9: Beim Fahrauftrag-Start war kein Referenzpunkt gesetzt
9	n10: PSTOP aktiv
10	n11: NSTOP aktiv
11	n12: Motordefaultwerte wurden geladen
12	n13: Erweiterungskarte arbeitet nicht ordnungsgemäß
13	n14: Motorphase
14	n15: fehlerhafter VCT-Eintrag angewählt
15	n16: Summenwarnung für n17 bis n31
16	n17: CAN-Sync ist nicht eingeloggt (mit SYNCSRC = 3)
17	n18 -Multiturn overflow: Max. Anzahl von Umdrehungen überschritten
18 ... 30	n19-n31: reserviert
31	n32 -Firmware ist eine nicht freigegebene Beta-Version

\* = Diese Warnmeldungen führen zu einem geführten Stillsetzen des Antriebs (Bremsung mit Notrampe)

**5.2.1.20 Objekt 2014-2017h: 1st-4th Mask 1 to 4 for Transmit-PDO**

Um die Buslast bei ereignisgesteuerten PDOs zu senken, kann die Überwachung für einzelne Bits im PDO mit Hilfe der Masken abgeschaltet werden. Beispielsweise kann so auch erreicht werden, dass Positionswerte immer nur einmal pro Umdrehung gemeldet werden.

Diese Objekte maskieren die PDO-Kanäle 1 bis 4. Sollten nur 2 Byte in einem PDO definiert sein, überlagert die Maske auch nur zwei Byte obwohl 4 Byte Maskeninformation übertragen worden ist. Ein aktiviertes Bit in der Maske bedeutet, dass die Überwachung für das korrespondierende Bit im PDO aktiv ist, d. h. überwacht wird.

Index	<b>2014h, 2015h, 2016h, 2017h</b>
Name	tx_mask 1 to 4
Objektcode	ARRAY
Datentyp	UNSIGNED32
Subindex	<b>1</b>
Beschreibung	tx_mask1 to 4_low
Mode	independent
Zugriff	rw
PDO Mapping	nicht möglich
Einheit	—
Wertebereich	UNSIGNED32
Defaultwert	FFFFFFFFh
Subindex	<b>2</b>
Beschreibung	tx_mask1 to 4_high
Mode	independent
Zugriff	rw
PDO Mapping	nicht möglich
Einheit	—
Wertebereich	UNSIGNED32
Defaultwert	FFFFFFFFh

**5.2.1.21 Objekt 2030h: DP-RAM Variables (write only)**

Index	<b>2030h</b>
Name	DP-Ram Variables, write only (PDO)
Objektcode	ARRAY
Datentyp	INTEGER32
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	9
Defaultwert	9
Subindex	<b>1...8</b>
Beschreibung	DP-Ram Variables 9...16
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	INTEGER32
Defaultwert	0

**5.2.1.22 Objekt 2040h: Gearing factor for electronic gearing**

Dieses Objekt definiert die Übersetzung für das elektronische Getriebe zwischen Master und Slave-Verstärker, die über Inkrementalgeber (ROD) verbunden sind. Diese Objekte sind nur relevant für OPMODE 4 bzw. CANopen-Mode 0x84.

Index	<b>2040h</b>
Name	Electronic gearing factor
Objektcode	RECORD
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	2
Defaultwert	2

Subindex 1 bezieht sich auf das Eingangssignal vom Master abhängig vom gewählten Feedback und GEARMODE.

Subindex	<b>1</b>
Beschreibung	Gearing Input
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	INTEGER32
Defaultwert	1024

Subindex 2 liefert die Bewegung des Slave in Abhängigkeit von den Master Impulsen.

Subindex	<b>2</b>
Beschreibung	Gearing Output
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGN32ED
Defaultwert	1



**5.2.1.23 Objekt 2041h: Electric gearing actual value**

Dieses Objekt dient zur Anzeige der eingelesenen Mastergeschwindigkeit, der internen Slave-Sollgeschwindigkeit als Richtgröße für den Slave des elektrischen Getriebes und dem internen Steuerwort für die Ein- / Aussynchronisierung des elektrischen Getriebes. Ein Beispiel zur Verwendung des elektrischen Getriebes siehe (→ # 129).

Index	<b>2041h</b>
Name	Electric gearing actual value
Objektcode	RECORD
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	3
Defaultwert	3

Subindex 1 dient zur Anzeige des aus der Mastergeschwindigkeit über das Übersetzungsverhältnis (Objekt 2040h) ermittelten internen Soll-Slavegeschwindigkeit.

Subindex	<b>1</b>
Beschreibung	Slave velocity set point
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	INTEGER32
Defaultwert	0

Subindex 2 zeigt die ermittelte Mastergeschwindigkeit an.

Subindex	<b>2</b>
Beschreibung	Master velocity of electric gearing
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	INTEGER32
Defaultwert	0

Subindex 3 dient zur Anzeige des Synchronisationszustandes der Slave-Achse des elektrischen Getriebs. Zur Bedeutung siehe Beispiel (→ # 129).

Subindex	<b>3</b>
Beschreibung	Internal control word for electric gearing
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	0, 1, 2, 5, 6
Defaultwert	0

### 5.2.1.24 Objekt 2051h: Configuration of Position Registers

Dieses Objekt dient zur Freigabe, Einstellung der Triggerart und der Polarität der 16 Positionsregister. Über die Objekte 2052h und 2053h wird vorgegeben, ob die Position absolut oder relativ zur Istposition ausgewertet werden sollen.

Index	<b>2051h</b>
Name	Configuration of Position Registers
Objektcode	ARRAY
Datentyp	UNSIGNED32
Kategorie	optional

Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	3
Defaultwert	3

Über den Subindex 1 können die 16 P1 .. P16 Positionsregister über die zugeordneten Bits 0..15 einzeln freigegeben oder gesperrt werden. Freigabe erfolgt mit Setzen des jeweiligen Bits.

Subindex	<b>1</b>
Beschreibung	Enable position registers
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	0..65535
Defaultwert	0

Subindex 2 definiert die Triggerart der 16 Positionsregister P1-P16. Wird das entsprechende Bit auf 0 gesetzt, wird die Position ständig überwacht. Wird es auf 1 gesetzt, wird nach Ansprechen des Positionsregisters das entsprechende Freigabebit aus Subindex 1 zurückgesetzt und damit die Überwachung abgeschaltet.

Subindex	<b>2</b>
Beschreibung	Position register mode
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	0..65535
Defaultwert	0

Mit Hilfe des Subindex 3 wird die Polarität der Überwachung der 16 Positionsregister P1-P16 eingestellt. Bei Einstellung auf 0 wird das Überschreiten der Position überwacht, bei 1 das Unterschreiten der Position des entsprechenden Positionsregisters.

Subindex	<b>3</b>
Beschreibung	Position register polarity
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	0..65535
Defaultwert	0

**5.2.1.25 Objekt 2052h: Position Registers, absolute**

Über dieses Objekt werden die Werte der 16 Positionsregister P1-P16 als absolute Positionsangaben eingetragen.

Index	<b>2052h</b>
Name	Position registers, absolute
Objektcode	ARRAY
Datentyp	INTEGER32
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	16
Defaultwert	16

Über die Subindex 1 bis 16 werden die zu überwachenden Positionen der Positionsregister P1...P16 angesprochen.

Subindex	<b>1...16</b>
Beschreibung	Position registers 1..16, absolute
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	INTEGER32
Defaultwert	0

**5.2.1.26 Objekt 2053h: Positions Registers, relative**

Über dieses Objekt werden die Werte der 16 Positionsregister P1-P16 als Positionsangaben relativ zur aktuellen Istposition eingetragen.

Index	<b>2053h</b>
Name	Position registers, relative
Objektcode	ARRAY
Datentyp	INTEGER32
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	16
Defaultwert	16

Über die Subindex 1 bis 16 werden die zu überwachenden Positionen der Positionsregister P1...P16 angesprochen.

Subindex	<b>1...16</b>
Beschreibung	Positions registers 1..16, relative
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	INTEGER32
Defaultwert	0

**5.2.1.27 Objekt 2061h: Current limitation for velocity mode**

Dieses Objekt dient zur schnellen Strombegrenzung im Drehzahlmode (0x3). Ein Wert von 3280 entspricht dabei dem maximalen Gerätestrom, der über DIPEAK abfragbar ist. Dieses Objekt wirkt auf den ASCII-Parameter DPRILIMIT. Damit es wirksam werden kann, muss der Konfigurationsparameter DILIM auf 1 stehen.

Index	<b>2061h</b>
Name	Current limitation in velocity mode
Objektcode	VAR
Datentyp	UNSIGNED16
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	0 .. 3280
Defaultwert	0

**5.2.1.28 Objekt 2080h: Motion Task Select**

Dieses Objekt ist eine Erweiterung des Profile Position Mode. Wenn der Wert ungleich 0 ist, wird der adressierte Fahrsatz mit der nächsten steigenden Flanke des Bit "New Setpoint" (Bit 4) des Controlword gestartet, sofern auch das Bit "Change Set Immediately" (Bit 5) gesetzt ist. Nach Start des Fahrsatzes wird der Wert des Objekts automatisch wieder auf 0 zurückgesetzt.

Index	<b>2080h</b>
Name	Motion tasks in PP-Mode
Objektcode	VAR
Datentyp	INTEGER16
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	1 to 300
Defaultwert	0

**5.2.1.29 Objekt 2081h: Active Motion Task Display**

Dieses Objekt zeigt den letzten Fahrauftrag an, der im Verstärker gestartet wurde. Fahrsatznummern von 1 bis 200 zeigen Flash-EEPROM Fahraufträge, Nummern von 201 bis 300 zeigen RAM Fahraufträge an.

Wenn Objekt 2080h keinen Wert beinhaltet und ein Fahrauftrag über den "New-setpoint/setpoint acknowledge" Mechanismus des Profile Position Mode gestartet wurde, wird Fahrauftrag 0 verwendet und angezeigt.

Wenn Sie eine Serie von Folgefahraufträgen starten (Bit 3 des Fahrauftrag Control Word O\_C, Objekt 35B9 sub1 gesetzt), wird der aktive Fahrauftrag angezeigt.

Index	<b>2081h</b>
Name	Active Motion Task Display
Objektcode	VAR
Datentyp	UNSIGNED16
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	1...300
Defaultwert	0

**5.2.1.30 Objekt 2082h: Copy motion tasks**

Mit diesem Objekt können Fahraufträge im Servoverstärker kopiert werden. Der Fahrauftrag, der im Low Word adressiert ist, wird zu dem Fahrsatz kopiert, der im High Word adressiert ist.

**INFO**

EEPROM Fahraufträge 1...200 können nur bei gesperrter Endstufe geschrieben werden.

Index	<b>2082h</b>
Name	Copy motion tasks
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	wo
PDO Mapping	nicht möglich
Wertebereich	High Word: 0 .. 300, low word: 0 .. 300
Defaultwert	-

**5.2.1.31 Objekt 2083h: Delete Motion tasks**

Das Objekt ermöglicht das Löschen aller Flash-EEPROM Fahrsätze. Dieser Vorgang kann nur gestartet werden, wenn eine spezielle Signatur ("prom") geschrieben wurde.

"prom" muss als Unsigned32-Zahl 6D6F7270h übertragen werden.

Der Löschvorgang ist nur möglich, wenn die Endstufe gesperrt ist und der NMT-Status PREOPERATIONAL ist.

Index	<b>2083h</b>
Name	delete motion tasks
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	wo
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	-

**5.2.1.32 Objekt 2090h: DP-RAM Variables (read only)**

Index	<b>2090h</b>
Name	DP-Ram Variables, read only (PDO)
Objektcode	ARRAY
Datentyp	INTEGER32
Kategorie	optional

Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	9
Defaultwert	9

Subindex	<b>1...8</b>
Beschreibung	DP-Ram Variables 9...16
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	INTEGER32
Defaultwert	0

**5.2.1.33 Objekt 20A0h: Latch position 1, positive edge**

Über dieses Objekt wird die Position ausgelesen, an der die erste positive Flanke am digitalen Eingang 1 nach der Latchfreigabe (s.a. Objekt20A4) aufgetreten war. Durch das Auslesen wird die Latchmimik wieder für einen Vorgang aktiv geschaltet. Voraussetzung ist die Konfiguration des Eingangs 1 als Latcheingang (IN1MODE 26).

Index	<b>20A0h</b>
Name	Latch position 1 positive edge
Objektcode	VAR
Datentyp	INTEGER32
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	INTEGER32
Defaultwert	0

**5.2.1.34 Objekt 20A1h: Latch position 1, negative edge**

Über dieses Objekt wird die Position ausgelesen, an der die erste negative Flanke am digitalen Eingang 1 nach der Latchfreigabe (s.a. Objekt 20A4) aufgetreten war. Durch das Auslesen wird die Latchmimik wieder für einen Vorgang aktiv geschaltet. Voraussetzung ist die Konfiguration des Eingangs 1 als Latcheingang (IN1MODE 26).

Index	<b>20A1h</b>
Name	Latch position 1 negative edge
Objektcode	VAR
Datentyp	INTEGER32
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	INTEGER32
Defaultwert	0

**5.2.1.35 Objekt 20A2h: Latch position 2, positive edge**

Über dieses Objekt wird die Position ausgelesen, an der die erste positive Flanke am digitalen Eingang 2 nach der Latchfreigabe (s.a. Objekt20A4) aufgetreten war. Durch das Auslesen wird die Latchmimik wieder für einen Vorgang aktiv geschaltet. Voraussetzung ist die Konfiguration des Eingangs 2 als Latcheingang (IN2MODE 26).

Index	<b>20A2h</b>
Name	Latch position 2 positive edge
Objektcode	VAR
Datentyp	INTEGER32
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	INTEGER32
Defaultwert	0

**5.2.1.36 Objekt 20A3h: Latch position 2, negative edge**

Über dieses Objekt wird die Position ausgelesen, an der die erste negative Flanke am digitalen Eingang 2 nach der Latchfreigabe (s.a. Objekt 20A4) aufgetreten war. Durch das Auslesen wird die Latchmimik wieder für einen Vorgang aktiv geschaltet. Voraussetzung ist die Konfiguration des Eingangs 2 als Latcheingang (IN2MODE 26).

Index	<b>20A3h</b>
Name	Latch position 2 negative edge
Objektcode	VAR
Datentyp	INTEGER32
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	INTEGER32
Defaultwert	0

**5.2.1.37 Objekt 20A4h: Latch Control Register**

Das Latch-Kontrollregister dient zum Freischalten der Latch-Überwachung der Digitaleingänge 1 und 2. Freigeschaltet wird mit einem 1-Signal, gesperrt mit einem 0-Signal. Ob ein Latchereignis aufgetreten ist, lässt sich über den Herstellerstatus (siehe Tabelle), Objekt 1002, erkennen.

Index	<b>20A4h</b>
Name	Latch Control Register
Objektcode	VAR
Datentyp	UNSIGNED8
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	0 to 15
Defaultwert	0

Bit	Flanke	Bit in Objekt 1002
0	Freigabe der positiven Flanke an Digitaleingang 1	20
1	Freigabe der negativen Flanke an Digitaleingang 1	21
2	Freigabe der positiven Flanke an Digitaleingang 2	4
3	Freigabe der negativen Flanke an Digitaleingang 2	18
4 to 7	reserviert	

**5.2.1.38 Objekt 20B0h: Trigger Variable Digital Input 20**

Über dieses Objekt kann die Triggervariable für den Softwareeingang 20 eingestellt werden. Dieses auch mappbare Objekt kann für den Start-/Stop-Vorgang im elektrischen Getriebe verwendet werden (bei IN20MODE = 51 oder 53).

Index	<b>20B0h</b>
Name	Trigger variable digital input 20
Objektcode	VAR
Datentyp	INTEGER32
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	INTEGER32
Defaultwert	0

**5.2.1.39 Objekt 20B1h: Control Word Digital Inputs 5..20**

Über die unteren 16 Bits dieses Objekt können die digitalen Softwareeingänge 5 bis 20 angesteuert werden.

Index	<b>20B1h</b>
Name	Control word digital inputs 5 .. 20
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	0...65535
Defaultwert	0

**5.2.1.40 Objekt 20B2h: Analog Inputs**

Mit Hilfe dieses Objekts können die Spannungen der Analogeingänge 1 und 2 eingelesen werden.

Index	<b>20B2h</b>
Name	Analog Inputs
Objektcode	ARRAY
Datentyp	INTEGER16
Kategorie	optional
<b>Subindex</b>	<b>0</b>
Beschreibung	Anzahl der Einträge
Datentyp	UNSIGNED8
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	2
Defaultwert	2

Subindex 1 liefert den Digitalwert des Analogeingangs 1. (10 V ~ 2000h).

<b>Subindex</b>	<b>1</b>
Beschreibung	Voltage analog input 1
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	INTEGER16
Defaultwert	-

Subindex 2 liefert den Digitalwert des Analogeingangs 2. (10 V ~ 2000h).

<b>Subindex</b>	<b>2</b>
Beschreibung	Voltage analog input 2
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	INTEGER16
Defaultwert	-



**5.2.1.41 Objekt 2100h: Write Dummy**

Index	<b>2100h</b>
Name	Write-Dummy variables for mapping
Objektcode	RECORD
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED32
Defaultwert	1

**5.2.1.42 Objekt 2101h: Read Dummy**

Index	<b>2101h</b>
Name	Read-Dummy variables for mapping
Objektcode	RECORD
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	UNSIGNED32
Defaultwert	1

**5.2.1.43 Objekt 60FDh: Digital inputs (DS402)**

Dieser Index definiert einfache digitale Eingänge für Servoverstärker. Die Bits 0 bis 2 können vom Servoverstärker bedient werden, wenn die benötigte Funktion dem digitalen Eingang mit dem ASCII-Kommando INxMODE (x von 1 bis 4) zugeordnet wurde, z.B. IN3MODE = 2, PSTOP-Funktion, siehe ASCII-Dokumentation.

Index	<b>60FDh</b>
Name	digital inputs
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	UNSIGNED32
Defaultwert	0

31	16	15	4	3	2	1	0
herstellerspezifisch		reserviert		enable	Referenzschalter	Endschalter positiv	Endschalter negativ
MSB							LSB

Die Schalter müssen als Schließler ausgelegt sein (active high).

#### 5.2.1.44 Objekt 6502h: Supported drive modes (DS402)

Ein Servoverstärker unterstützen mehr als einen und verschiedene unterschiedliche Betriebsarten. Dieses Objekt gibt einen Überblick über die möglichen Betriebsarten des Servoverstärkers.

Index	<b>6502h</b>									
Name	supported drive modes									
Objektcode	VAR									
Datentyp	UNSIGNED32									
Kategorie	optional									
Zugriff	ro									
PDO Mapping	möglich									
Wertebereich	UNSIGNED32									
Defaultwert	0x6D (ip hm tq pv pp)									
31	16	15	7	6	5	4	3	2	1	0
herstellerspezifisch		reserviert		ip	hm	reserviert	tq	pv	vl	pp
MSB										LSB

### 5.3 PDO Konfiguration

PDOs werden für die Prozess Daten Kommunikation verwendet. Es gibt zwei unterschiedliche Typen von PDOs: Empfangs PDOs (RPDOs) und Sende PDOs (TPDOs).

Der Inhalt der PDOs ist vordefiniert (siehe Beschreibung auf den Seiten (→ # 51) und (→ # 53)). Wenn der Dateninhalt für eine spezielle Anwendung nicht brauchbar ist, können die Datenobjekte in den PDOs frei geändert (remapped) werden.

Dateneintrag in den PDOs

MSB		LSB
Index (16 Bit)	Subindex (8 Bit)	Datenlänge in Bits (8 Bit)

Konfigurationsprozedur für ein freies Mapping (Beispiel für TPDO1)

#### 1. Aktuelles Mapping löschen: Schreiben von 0 in Subindex 0 des Objekts

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
601	2F	00	1A	00h	00 00 00 00	Lösche aktuelles Mapping

#### 2. Mapping aufbauen mit Objekten des Objektverzeichnisses (→ # 103) die gemappt werden dürfen, z.B.

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
601	23	00	1A	01h	10 00 41 60	Erster Eintrag: CANopen Statuswort mit 16 Bits
601	23	00	1A	02h	20 00 02 10	Zweiter Eintrag: Herstellerstatus mit 32 Bits

#### 3. Schreibe die Anzahl der gemappten Objekte in Subindex 0 des Objekts.

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
601	2F	00	1A	00h	02 00 00 00	Prüfe Anzahl Einträge

Das Mapping sollte immer beendet sein, bevor das Network Management auf OPERATIONAL geschaltet wird.

### 5.3.1 Receive PDOs (RXPDO)

Vier Empfangs PDOs können im Servoverstärker konfiguriert werden:

- Konfigurieren der Kommunikation (1400-1403h)
- Konfigurieren des PDO Inhalts (Mapping, Objekte 1600-1603h)

#### 5.3.1.1 Objekte 1400-1403h: 1st-4th RxPDO communication parameter (DS301)

Index	<b>1400h, 1401h, 1402h, 1403h for RXPDO 1 ... 4</b>
Name	receive PDO parameter
Objektcode	RECORD
Datentyp	PDO CommPar
Kategorie	obligatorisch

Definierte Sub-Indizes

Subindex	<b>0</b>
Name	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	2
Defaultwert	2
Subindex	<b>1</b>
Name	COB-ID used by PDO
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	Index 1400h: 200h + Node-ID      Index 1401h: 300h + Node-ID Index 1402h: 400h + Node-ID      Index 1403h: 500h + Node-ID

**Subindex 1** beinhaltet die COB-Id des PDO als Bit kodierte Information:

Bit	Wert	Bedeutung
31	0	PDO existiert / ist gültig
	1	PDO existiert nicht / ist nicht gültig
30	0	RTR erlaubt auf diesem PDO, nicht unterstützt
	1	RTR nicht erlaubt auf diesem PDO, nicht unterstützt
29	0	11 Bit-ID (CAN 2.0A)
	1	29 Bit-ID (CAN 2.0B), nicht unterstützt
28 ... 11	X	Identifizier-Bits mit 29 Bit-ID, nicht relevant
	0	wenn Bit 29=0
10 ... 0	X	Bits 10-0 der COB-ID

**Subindex 2** beinhaltet den Übertragungstyp des PDO. Es gibt zwei Einstellarten:

- Wert FFh oder 255 für Ereignis gesteuertes PDO, das direkt vom Empfänger interpretiert und ausgeführt wird.
- Werte von 0 bis 240 führen zu einer mit dem SYNC Telegramm kontrollierten Interpretation des PDO Inhalts. Werte von 1 bis 240 bedeuten, dass 0 bis 239 SYNC-Telegramme ignoriert werden, bevor eines interpretiert wird. Der Wert 0 bedeutet, dass das nächste SYNC-Telegramm interpretiert wird.

## 5.3.1.2 Objekte 1600-1603h: 1st-4th RxPDO Mapping Parameter (DS301)

Index	<b>1600h, 1601h, 1602h, 1603h for RxPDO 1 to 4.</b>
Name	receive PDO Mapping
Objekt Code	RECORD
Datentyp	PDO Mapping
Kategorie	obligatorisch
Subindex	<b>0</b>
Name	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	0: PDO ist nicht aktiv 1-8: PDO aktiviert, Mappings werden nur Byte-weise übernommen
Defaultwert	PDO1: 1, PDO2: 2, PDO3: 2, PDO4: 2
Subindex	<b>1-8</b>
Name	PDO-mapping for the n-th application object
Kategorie	konditional, hängt von Anzahl und Größe der gemappten Objekte ab.
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	Siehe unten

## 5.3.1.3 Default RXPDO Definition

## RXPDO 1:

Subindex	Wert	Bedeutung
0	1	Ein PDO Mapping Eintrag
1	60 40 00 10	Controlword

## RXPDO 2:

Subindex	Wert	Bedeutung
0	2	Zwei PDO Mapping Einträge
1	60 40 00 10	Controlword
2	60 60 00 08	Betriebsart

## RXPDO 3:

Subindex	Wert	Bedeutung
0	2	Zwei PDO Mapping Einträge
1	60 40 00 10	Controlword
2	60 7A 00 20	Zielposition (Betriebsart PP)

## RXPDO 4:

Subindex	Wert	Bedeutung
0	2	Zwei PDO Mapping Einträge
1	60 40 00 10	Controlword
2	60 FF 00 20	Zielgeschwindigkeit (Betriebsart PV)

### 5.3.2 Transmit PDOs (TxPDO)

Vier Sende PDOs können im Servoverstärker konfiguriert werden:

- Konfigurieren der Kommunikation (Objekte 1800-1803h)
- Konfigurieren des PDO Inhalts (Mapping, Objekte 1A00-1A03h)

#### 5.3.2.1 Objekte 1800-1803h: 1st-4th TxPDO communication parameter (DS301)

Index	<b>1800h, 1801h, 1802h, 1803h for TxPDO 1 to 4.</b>
Name	transmit PDO parameter
Objektcode	RECORD
Datentyp	PDO CommPar
Kategorie	obligatorisch
Subindex	<b>0</b>
Name	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	5
Defaultwert	5
Subindex	<b>1</b>
Name	COB-ID used by PDO
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	Index 1800h: 40000180h + Node-ID Index 1801h: 40000280h + Node-ID Index 1802h: 40000380h + Node-ID Index 1803h: 40000480h + Node-ID
Subindex	<b>2</b>
Name	transmission type
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED8
Defaultwert	FFh
Subindex	<b>3</b>
Name	inhibit time
Kategorie	optional
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED16 (n*1/10ms)
Defaultwert	0h

Subindex	<b>4</b>
Name	reserviert
Kategorie	optional
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	0
Defaultwert	0

Subindex	<b>5</b>
Name	event timer
Kategorie	optional
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED16 (0=nicht benutzt, n*1/10ms)
Defaultwert	0h

**Subindex 1** beinhaltet die COB-Id des PDO als Bit kodierte Information:

Bit	Wert	Bedeutung
31	0	PDO existiert / ist gültig
	1	PDO existiert nicht / ist nicht gültig
30	0	RTR erlaubt auf diesem PDO, unterstützt ab Firmware 3.00
	1	RTR nicht erlaubt auf diesem PDO. Ab FW 3.00 muss dieses Bit gesetzt werden, da generell RTR-Zugriffe auf Tx-PDOs nicht unterstützt werden.
29	0	11 Bit-ID (CAN 2.0A)
	1	29 Bit-ID (CAN 2.0B), nicht unterstützt
28 to 11	X	Identifizier-Bits mit 29 Bit-ID, nicht relevant
	0	Wenn Bit 29=0
10 to 0	X	Bits 10-0 der COB-ID

**Subindex 2** beinhaltet den Übertragungstyp des PDO. Es gibt zwei Einstellarten:

- Wert FFh oder 255 für ein ereignisgesteuertes PDO, das sofort nach einer Änderung in den gemappten Objekten gesendet wird. Die Einstellungen Setting von Subindex 3 und 5 haben einen Einfluss auf die Sendung des PDO.
  - Mit Subindex 3 können Sie konfigurieren, nach welcher Mindestzeit das PDO gesendet wird, wenn sich der PDO-Dateninhalt geändert hat (Reduktion der Busbelastung).
  - Mit Sub\_index 5 wird ein Timer verwendet, der nach jedem Ereignis gesteuerten Senden des PDO neu gestartet wird. Auch wenn der PDO Inhalt sich nicht geändert hat, wird das PDO angestoßen vom Event Timer gesendet.
- Werte von 0 bis 240 führen zu vom SYNC Telegramm kontrolliertem Senden des PDO. Werte von 1 bis 240 definieren, wie oft das SYNC-Telegramm zur Sendung des PDO führt. 0 bedeutet, dass das nächste SYNC-Telegramm zum Senden des PDO führt.

## 5.3.2.2 Objekte 1A00-1A03h: 1st-4th TxPDO mapping parameter (DS301)

Index	1A00h, 1A01h, 1A02h, 1A03h for TxPDO 1 to 4.
Name	transmit PDO Mapping
Objekt Code	RECORD
Datentyp	PDO Mapping
Kategorie	obligatorisch
Subindex	0
Name	number of mapped application objects in PDO
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	0: PDO ist nicht aktiv 1-8: PDO aktiviert, Mappings werden nur Byte-weise übernommen
Defaultwert	PDO1: 1, PDO2: 2, PDO3: 2, PDO4: 2
Subindex	1-8
Name	PDO-mapping for the n-th application object
Kategorie	konditional, hängt von Anzahl und Größe der gemappten Objekte ab.
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	Siehe unten

## 5.3.2.3 Default TXPDO definition

## TXPDO 1:

Subindex	Wert	Bedeutung
0	1	Ein PDO-Mapping Eintrag
1	60 41 00 10	Statusword

## TXPDO 2:

Subindex	Wert	Bedeutung
0	2	Zwei PDO-Mapping Einträge
1	60 41 00 10	Statusword
2	60 61 00 08	Anzeige der Betriebsart

## TXPDO 3:

Subindex	Wert	Bedeutung
0	2	Zwei PDO-Mapping Einträge
1	60 41 00 10	Statusword
2	60 64 00 20	Positions-Istwert

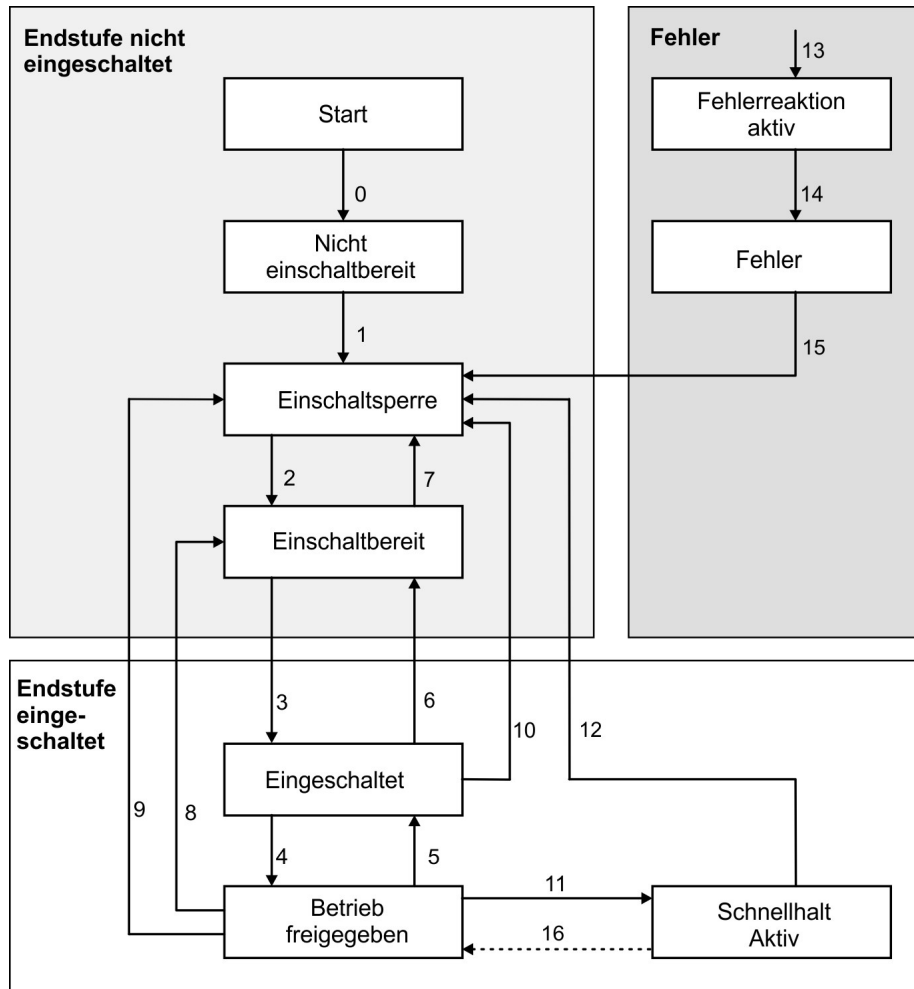
## TXPDO 4:

Subindex	Wert	Bedeutung
0	2	Zwei PDO-Mapping Einträge
1	60 41 00 10	Statusword
2	60 6C 00 20	Geschwindigkeits-Istwert

## 5.4 Device control (dc)

Mit Hilfe der S300/S700 Gerätesteuerung können sämtliche Fahrfunktionen in den entsprechenden Betriebsarten ausgeführt werden. Die Steuerung des S300/S700 ist über eine Betriebsart-abhängige Zustandsmaschine realisiert. Zur Steuerung der Zustandsmaschine dient das controlword (→ # 58). Die Betriebsarteinstellung erfolgt über das Objekt "Modes of Operation" (→ # 103). Die Zustände der Zustandsmaschine können mit dem statusword ermittelt werden (→ # 60).

### 5.4.1 Zustandsmaschine (DS402)



#### 5.4.1.1 Zustände der Zustandsmaschine

Zustand	Beschreibung
Not Ready to Switch On	S300/S700 ist nicht einschaltbereit, es wird keine Betriebsbereitschaft (BTB) vom Reglerprogramm gemeldet.
Switch On Disable	S300/S700 ist einschaltbereit, Parameter können übertragen werden, Zwischenkreisspannung kann eingeschaltet werden, Fahrfunktionen können noch nicht ausgeführt werden
Ready to Switch On	Zwischenkreisspannung kann angeschaltet werden. Parameter können übertragen werden, Fahrfunktionen können noch nicht ausgeführt werden
Switched On	Zwischenkreisspannung muss angeschaltet sein, Parameter können übertragen werden, Fahrfunktionen können noch nicht ausgeführt werden, Endstufe ist freigegeben (Enabled)
Operation Enable	kein Fehler steht an, Endstufe und Fahrfunktionen freigegeben (Enabled)
Quick Stop Active	Antrieb ist mit der Notbremsrampe gestoppt worden, Endstufe ist freigegeben, Fahrfunktionen sind nicht freigegeben.
Fault Reaction Active	Ein Fehler ist aufgetreten und der Antrieb wird mit der Rampe Quickstop gestoppt.
Fault	Ein Fehler liegt an, der Antrieb wurde gestoppt und gesperrt.



### 5.4.1.2 Übergänge der Zustandsmaschine

Die Zustandsübergänge werden durch interne Ereignisse (z.B. Ausschalten der Zwischenkreisspannung) und durch die Flags im controlword (Bits 0,1,2,3,7) beeinflusst.

Übergang	Ereignis	Aktion
0	Reset	Initialisierung
1	Initialisierung erfolgreich abgeschlossen. S300/S700 betriebsbereit	keine
2	Bit 1 (disable Voltage) und Bit 2 (Quick Stop) im controlword gesetzt ('Shutdown' command). Zwischenkreisspannung kann anliegen.	keine
3	Bit 0 wird zusätzlich gesetzt ('Switch On' command)	Endstufe wird freigegeben (enable), sofern Hardware-enable anliegt (UND-Verknüpfung). Antrieb hat Drehmoment.
4	Bit 3 wird zusätzlich gesetzt ('Enable Operation' command)	Fahrfunktion in Abhängigkeit des eingestellten Mode wird freigegeben.
5	Bit 3 wird gelöscht ('Disable Operation' command)	Fahrfunktion wird gesperrt. Antrieb wird mit der relevanten Rampe (modeabhängig) gestoppt. Aktuelle Position wird gehalten.
6	Bit 0 wird gelöscht ('Shutdown' command)	Endstufe wird gesperrt (disable). Motor wird drehmomentfrei.
7	Bit 1/2 werden gelöscht ('Quickstop' / 'Disable Voltage' command)	keine
8	Bit 0 wird gelöscht ('Shutdown' command)	Endstufe wird gesperrt (disable) Motor wird drehmomentfrei.
9	Bit 1 wird gelöscht ('Disable Voltage' command)	Endstufe wird gesperrt (disable) Motor wird drehmomentfrei.
10	Bit 1/2 werden gelöscht ('Quickstop' / 'Disable Voltage' command)	Endstufe wird gesperrt (disable). Motor wird drehmomentfrei.
11	Bit 2 wird gelöscht ('Quickstop' command)	Antrieb wird mit der Notbremsrampe angehalten. Die Endstufe bleibt "enabled". Sollwerte werden gelöscht (Fahrsatznummer, digitaler Sollwert, Geschwindigkeit für Tippbetrieb oder Referenzfahrt). Vor dem erneuten Ausführen einer Fahrfunktion muss Bit 2 wieder gesetzt werden.
12	Bit 1 wird gelöscht ('Disable Voltage' command)	Endstufe wird abgeschaltet (disable) Motor wird drehmomentfrei.
13	Fehler Reaktion aktiv	Fehlerreaktion wird ausgeführt.
14	Fehler Reaktion beendet	Antriebsfunktion ist deaktiviert. Endstufe kann ausgeschaltet werden.
15	"Fault Reset" Kommando vom Host empfangen	Ein Fehler-Reset wird ausgeführt, wenn aktuell kein Fehler im Servoverstärker vorliegt. Wenn der Status "Fault" verlassen wird, muss das Bit7 "Reset Fault" des Kontrollwortes vom Host gelöscht werden.
16	Bit 2 wird gesetzt	Fahrfunktion ist wieder freigegeben.

#### INFO

Wenn der Servoverstärker über das controlword / statusword bedient wird, dürfen keine Steuerkommandos über einen anderen Kommunikationskanal (RS232, CANopen, ASCII-Kanal, Optiskarte) gesendet werden.

## 5.4.2 Objekt Beschreibung

### 5.4.2.1 Objekt 6040h: Control word (DS402)

Aus der logischen Verknüpfung der Bits im controlword und externen Signalen (Freigabe der Endstufe) ergeben sich die Steuerkommandos.

Index	<b>6040h</b>
Name	control word
Objektcode	VAR
Datentyp	UNSIGNED16
Zugriff	rw
PDO Mapping	möglich
Einheit	—
Wertebereich	0 to 65535
EEPROM	nein
Defaultwert	0

#### Bitbelegung im control word

Bit	Name	Bit	Name
0	Switch on	8	Halt (Zwischenstopp)
1	Disable Voltage	9	reserviert
2	Quick Stop	10	reserviert
3	Enable Operation	11	reserviert
4	Betriebsartspezifisch	12	reserviert
5	Betriebsartspezifisch	13	Herstellerspezifisch
6	Betriebsartspezifisch	14	Herstellerspezifisch
7	Fault Reset )*	15	Herstellerspezifisch

)\* Über Bit 7 (Reset Fault) werden auch die Warnungen Schleppfehler (n03) und Ansprechüberwachung (Nodeguarding / Heartbeat-Fehler) gelöscht.

#### Kommandos im control word

Kommando	Bit 7 Fault Reset	Bit 3 Enable Operation	Bit 2 Quick Stop	Bit 1 Disable Voltage	Bit 0 Switch on	Übergänge
Shutdown	X	X	1	1	0	2, 6, 8
Switch on	X	X	1	1	1	3
Disable Voltage	X	X	X	0	X	7, 9, 10, 12
Quick Stop	X	X	0	1	X	7, 10, 11
Disable Operation	X	0	1	1	1	5
Enable Operation	X	1	1	1	1	4, 16
Fault Reset	1	X	X	X	X	15

Mit X gekennzeichnete Bits sind irrelevant.

**Modeabhängige Bits im controlword**

Die folgende Tabelle beschreibt die modeabhängigen Bits im controlword. Es werden z. Zt. nur herstellerspezifische Modi unterstützt. Die einzelnen Modi werden mit dem Objekt 6060h "Modes of operation" eingestellt.

Betriebsart	Nr.	Bit 4	Bit 5	Bit 6
Lage	88h	reserviert	reserviert	reserviert
Drehzahl digital	80h	reserviert	reserviert	reserviert
Strom digital	82h	reserviert	reserviert	reserviert
Drehzahl analog	81h	reserviert	reserviert	reserviert
Strom analog	83h	reserviert	reserviert	reserviert
Profile Position Mode (pp)	01h	new_set_point	change_set_immediately	absolut / relativ
Profile Velocity Mode (pv)	03h	reserviert	reserviert	reserviert
Profile Torque Mode (tq)	04h	reserviert	reserviert	reserviert
Homing Mode (hm)	06h	homing_operation_start	reserviert	reserviert
Interpolated Position Mode (ip)	07h		reserviert	reserviert

**Beschreibung der restlichen Bits im controlword**

Im Folgenden werden die restlichen Bits des controlword beschrieben.

**Bit 8 Zwischenstopp** Ist Bit 8 gesetzt, wird in sämtlichen Modi der Antrieb gestoppt (Zwischenstopp). Die Sollwerte (Geschwindigkeit für Referenzfahrt oder Tippbetrieb, Fahrauftragsnummer, Sollwerte für Digitalmode) der einzelnen Modi bleiben erhalten.

**Bit 9,10** Diese Bits sind durch das Antriebsprofil (DS402) reserviert..

**Bit 13, 14, 15** Diese Bits sind herstellerspezifisch und z.Zt. reserviert.

### 5.4.2.2 Objekt 6041h: Status word (DS402)

Der momentane Zustand der Zustandsmaschine kann mit Hilfe des statusword abgefragt werden.

Index	<b>6041h</b>
Name	Status word
Objektcode	VAR
Datentyp	UNSIGNED16
Zugriff	rw
PDO Mapping	möglich
Einheit	—
Wertebereich	0 to 65535
EEPROM	ja
Defaultwert	0

#### Bitbelegung im Statusword

Bit	Name	Bit	Name
0	Ready to switch on	8	Herstellerspezifisch (reserviert)
1	Switched on	9	Remote (immer 1)
2	Operation enable	10	Target reached
3	Fault	11	Internal limit active
4	Voltage enabled	12	Betriebsartspezifisch (reserviert)
5	Quick stop	13	Betriebsartspezifisch (reserviert)
6	Switch on disabled	14	Herstellerspezifisch (reserviert)
7	Warning	15	Herstellerspezifisch (reserviert)

#### Zustände der Statusmaschine

Zustände	Bit 6 switch on disable	Bit 5 quick stop	Bit 3 fault	Bit 2 operation enable	Bit 1 switched on	Bit 0 ready to switch on
Not ready to switch on	1	X	0	0	0	0
Switch on disabled	1	X	0	0	0	0
Ready to switch on	0	1	0	0	0	1
Switched on	0	1	0	0	1	1
Operation enabled	0	1	0	1	1	1
Fault	0	X	1	0	0	0
Fault reaction active	0	X	1	1	1	1
Quick stop active	0	0	0	1	1	1

Die mit X gekennzeichneten Bits sind irrelevant

#### Beschreibung der restlichen Bits im Statusword

**Bit 4:** voltage\_enabled Die Zwischenkreisspannung liegt an, wenn das Bit gesetzt ist.

**Bit 7:** warning Für das Setzen von Bit 7 kann es mehrere Gründe geben, die zu dieser Warnung geführt haben. Der Grund für diese Warnung kann anhand des Objekts 2000h "herstellerspezifische Warnungen" ermittelt werden.

**Bit 9:** remote ist immer auf 1 gesetzt, d.h. der Antrieb kann immer kommunizieren und über das RS232-Interface beeinflusst werden.

**Bit 10:** target\_reached Wird gesetzt, wenn der Antrieb die Zielposition erreicht hat.

**Bit 11:** internal\_limit\_active Dieses Bit drückt aus, dass eine Bewegung begrenzt wurde oder wird. In verschiedenen Modi führen unterschiedliche Warnungen zum Setzen des Bits. Es existieren folgende Zuordnungen:

Betriebsart	Warnungen, die Bit 11 setzen
alle	n04, n06, n07, n10, n11, n14
0x1 (PP), 0x88	n03, n08, n09, n20

### 5.4.2.3 Objekt 6060h: Modes of Operation (DS402)

Mit diesem Objekt wird die Betriebsart (Mode) eingestellt und mit dem Objekt 6061h kann die eingestellte Betriebsart gelesen werden. Es können zwei Arten von Operationsmodi unterschieden werden:

- **Herstellerspezifische Operationsmodi.**  
Diese Operationsmodi sind auf die Gerätefunktionalitäten optimierte Betriebszustände.
- **Operationsmodi nach CANopen Antriebsprofil DS402**  
Diese Operationsmodi nach CANopen Antriebsprofil DS402 definiert.

Nach einem Betriebsartwechsel muss der entsprechende Sollwert neu gesetzt werden (z.B.: Mode Referenzierung \_ Sollwert Referenzfahrtgeschwindigkeit). Wurde die Betriebsart Lage oder Tipbetrieb gespeichert, so ist nach einem RESET des Servoverstärkers die Betriebsart Referenzierung aktiv.

#### INFO

Ein Operationmode ist erst gültig, wenn er mit Objekt 6061h gelesen werden konnte.



#### **! WARNUNG** Automatischer Wiederanlauf!

Es besteht die Gefahr von tödlichen oder schweren Verletzungen für Personen, die in der Maschine arbeiten. Das Antrieb könnte sich unerwartet bewegen. Das Umschalten der Betriebsart ist bei freigegebenem Verstärker nur bei Drehzahl 0 erlaubt.

- Niemals die Betriebsart bei drehendem Motor umschalten!
- Setzen Sie vor dem Umschalten den Sollwert auf 0.

Index	<b>6060h</b>
Name	mode of operation
Objektcode	VAR
Datentyp	INTEGER8
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	möglich
Wertebereich	80h ... 88h, 1, 3, 4, 6, 7
Defaultwert	—

Unterstützte Modi (negative Werte sind herstellerspezifisch Modi):

Wert (hex)	Betriebsart
80	Digitale Drehzahlregler
81	Analoger Drehzahlregler
82	Digitaler Stromregler
83	Analoger Stromregler
84	Elektronisches Getriebe
85	reserviert
86	reserviert
87	reserviert
88	Fahrsatz
1	Profile position mode
3	Profile velocity mode
4	Profile torque mode
6	Homing mode
7	Interpolated position mode

#### 5.4.2.4 Objekt 6061h: Modes of Operation Display (DS402)

Mit diesem Objekt kann die über das Objekt 6060h eingestellte Betriebsart gelesen werden. Ein Operationmode ist erst gültig, wenn er mit Objekt 6061h gelesen werden konnte (s. auch 6060h).

Index	<b>6061h</b>
Name	mode of operation display
Objektcode	VAR
Datentyp	INTEGER8
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	möglich
Wertebereich	80h ... 88h, 1, 3, 4, 6, 7
Defaultwert	—

### 5.5 Factor Groups (fg) (DS402)

Factor Groups definieren die Dimensionen von Positions-, Geschwindigkeits- und Beschleunigungs-Sollwerten.

#### INFO

Zur Zeit ist die Einheitendefinition im CANopen Profil DS402 noch nicht endgültig festgelegt. Daher sollten die Objekte 6089h bis 609Eh nicht verwendet werden.

Die Antriebsparameter für die Einheitendefinition sollten wie folgt definiert werden:

PUNIT= 0 (counts)  
 VUNIT= 0 (counts / s)  
 ACCUNIT= 3 (counts / s<sup>2</sup>)

#### 5.5.1 Allgemeine Informationen

##### 5.5.1.1 Faktoren

Es gibt eine Umrechnungsmöglichkeit für die physikalischen Dimensionen und Größen auf die antriebsinternen Einheiten (Inkrementen). Dafür können einige Faktoren implementiert werden. Dieses Kapitel beschreibt den Einfluss dieser Faktoren auf das System, die Berechnungsvorschriften für diese Faktoren und die Daten, die dazu erforderlich sind.

##### 5.5.1.2 Beziehung zwischen physikalischen und antriebsinternen Einheiten

Die in den factor groups festgelegten Faktoren definieren die Beziehung zwischen antriebsinternen Einheiten (Inkrementen) und physikalischen Einheiten.

Die Faktoren sind das Ergebnis einer Berechnung von zwei Parametern: des dimension index und des notation index. Der dimension index beschreibt dabei die physikalische Dimension, der notation index die physikalische Einheit und den Dezimalpunkt für Werte. Diese Faktoren werden zur Normierung der physikalischen Werte verwendet.

Der notation index kann auf zwei Arten verwendet werden:

- Für eine Einheit mit dezimaler Skalierung und notation index < 64 definiert der notation index den Dezimalpunkt der Einheit.
- Für eine Einheit mit nicht-dezimaler Skalierung und notation index > 64 definiert der notation index den Subindex der physikalischen Dimension der Einheit.

## 5.5.2 Objekte für Positionsberechnungen

### 5.5.2.1 Objekt 6089h: position notation index (DS402)

Index	<b>6089h</b>
Name	position notation index
Objektcode	VAR
Datentyp	INTEGER8
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	INTEGER8
Defaultwert	0

Der "position notation index" skaliert Positions-Sollwerte, deren Einheiten im "position dimension index" in SI-Einheiten definiert sind, als 10er Potenzen.

Beziehung zwischen den Objektwerten und dem herstellerspezifischen Parameter PUNIT:

Wert von Objekt 6089h	ASCII Parameter PUNIT	Skalierung
FFh	1	10 <sup>-1</sup>
FEh	2	10 <sup>-2</sup>
FDh	3	10 <sup>-3</sup>
FCh	4	10 <sup>-4</sup>
FBh	5	10 <sup>-5</sup>
FAh	6	10 <sup>-6</sup>
F9h	7	10 <sup>-7</sup>
F8h	8	10 <sup>-8</sup>
F7h	9	10 <sup>-9</sup>
0	0	1

### 5.5.2.2 Objekt 608Ah: Position Dimension Index (DS402)

Index	<b>608Ah</b>
Name	position dimension index
Objektcode	VAR
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED8
Defaultwert	0

Der "position dimension index" definiert die SI Einheiten der verwendeten Positions-Sollwerte. Beziehung zwischen den Objektwerten und dem herstellerspezifischen Parameter PUNIT:

Wert von Objekt 608Ah	ASCII Parameter PUNIT	SI Einheit
1	9...1	m
0	0	Hersteller spezifische Inkremente

Der Parameter PUNIT kann im Servoverstärker gespeichert werden. Die Werte von Objekt 6089h und 608Ah werden von diesem Parameter initialisiert.

**5.5.2.3 Objekt 608Fh: Position encoder resolution (DS402)**

Dieser Index definiert das Verhältnis der Encoder Inkremente pro Umdrehung der Motorwelle. Das Objekt wird ebenso für Objekt6090 (velocity encoder resolution) verwendet.

position encoder resolution = encoder increments / motor revolutions

Index	<b>608Fh</b>
Name	Position encoder resolution
Objekt Code	ARRAY
Datentyp	UNSIGNED 32
Kategorie	optional
Subindex	<b>0</b>
Name	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	2
Defaultwert	2
Subindex	<b>1</b>
Name	Encoder increments
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED 32
Defaultwert	2 <sup>20</sup>
Subindex	<b>2</b>
Name	Motor revolutions
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED 32
Defaultwert	1



#### 5.5.2.4 Objekt 6091h: Gear Ratio (DS402)

Der Index definiert das Verhältnis von Umdrehung der Motorwelle (Antriebswelle) zu Umdrehung der Abtriebswelle.

gear ratio = motor shaft revolutions / driving shaft revolutions

Index	<b>6091h</b>
Name	Gear Ratio
Objekt Code	ARRAY
Datentyp	UNSIGNED 32
Kategorie	optional
Subindex	<b>0</b>
Name	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	2
Defaultwert	2
Subindex	<b>1</b>
Name	Motor revolution
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED 32
Defaultwert	1
Subindex	<b>2</b>
Name	Shaft revolutions
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED 32
Defaultwert	1

**5.5.2.5 Objekt 6092h: Feed constant (DS402)**

Hier wird das Verhältnis des Vorschubs (feed, in Positionseinheiten) pro Umdrehung der Abtriebswelle definiert. Dies schließt ein vorhandenes Getriebe mit ein.

$\text{feed constant} = \text{feed} / \text{driving shaft revolutions}$

Index	<b>6092h</b>
Name	Feed constant
Objekt Code	ARRAY
Datentyp	UNSIGNED 32
Kategorie	optional

Subindex	<b>0</b>
Name	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	2
Defaultwert	2

Subindex	<b>1</b>
Name	Feed
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED 32
Defaultwert	1

Subindex	<b>2</b>
Name	Shaft revolutions
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED 32
Defaultwert	1

**5.5.2.6 Objekt 6093h: Position factor (DS402)**

Der Positionsfaktor konvertiert die Sollposition (in Positionseinheiten) in das interne Format (Inkrement). Diese Werte werden über die Objekte 608F und 6091 berechnet.

$\text{position factor} = (\text{position encoder resolution} * \text{gear ratio}) / \text{feed constant}$

Index	<b>6093h</b>
Name	Position factor
Objekt Code	ARRAY
Datentyp	UNSIGNED 32
Kategorie	optional
Subindex	<b>0</b>
Name	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	2
Defaultwert	2
Subindex	<b>1</b>
Name	Zähler ( $\text{position encoder resolution} * \text{gear ratio}$ )
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED 32
Defaultwert	1
Subindex	<b>2</b>
Name	Feed constant
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED 32
Defaultwert	1

**5.5.2.7 Objekt 6094h: Velocity encoder factor (DS402)**

Das Objekt 6094 wird nur aus Kompatibilitätsgründen noch als read-only Objekt unterstützt. Es dient zur Umrechnung der Geschwindigkeit in das interne Datenformat des S300/S700 (Inkrement). Die Skalierung ist dabei entsprechend zum Objekt 6093 Subindex 1 und 2.

Index	<b>6094h</b>
Name	velocity_encoder_factor
Objekt Code	ARRAY
Datentyp	UNSIGNED32
Kategorie	optional
Subindex	<b>0</b>
Name	Anzahl der Einträge
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	2
Defaultwert	2
Subindex	<b>1</b>
Name	Zähler
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	0..(2 <sup>32</sup> -1)
Defaultwert	0
Subindex	<b>2</b>
Name	Nenner
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	0..(2 <sup>32</sup> -1)
Defaultwert	0

### 5.5.3 Objekte für Geschwindigkeitsberechnungen

#### 5.5.3.1 Objekt 608Bh: velocity notation index (DS402)

Index	<b>608Bh</b>
Name	velocity notation index
Objektcode	VAR
Datentyp	INTEGER8
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	INTEGER8
Defaultwert	0

Der "velocity notation index" skaliert Geschwindigkeits-Sollwerte, deren Einheiten im "velocity dimension index" in SI-Einheiten definiert sind, als 10er Potenzen.

Beziehung zwischen den Objektwerten und dem herstellerspezifischen Parameter VUNIT:

Wert von Objekt 608Bh	ASCII Parameter VUNIT	Skalierung
0	0	1
0	1	1
0	5	1
0	6	1
FDh	7	10-3
FDh	8	10-3

#### 5.5.3.2 Objekt 608Ch: velocity dimension index (DS402)

Index	<b>608Ch</b>
Name	velocity dimension index
Objektcode	VAR
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED8
Defaultwert	0

Der "velocity dimension index" definiert die SI Einheiten der verwendeten Geschwindigkeits Sollwerte. Beziehung zwischen den Objektwerten und dem herstellerspezifischen Parameter VUNIT:

Wert von Objekt 608Ch	ASCII Parameter VUNIT	SI Einheit
A6h	0	m/s
A4h	1	turn/min
A6h	5	m/s
A7h	6	m/min
A6h	7	m/s
A7h	8	m/min

Der Parameter VUNIT kann im Servoverstärker gespeichert werden. Die Werte von Objekt 608Bh und 608Ch werden von diesem Parameter initialisiert. Nur die beschriebenen Werte für VUNIT sind mit dem Profil DS402 möglich.

## 5.5.4 Objekte für Beschleunigungsberechnungen

### 5.5.4.1 Objekt 608Dh: acceleration notation index (DS402)

Index	<b>608Dh</b>
Name	acceleration notation index
Objektcode	VAR
Datentyp	INTEGER8
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	INTEGER8
Defaultwert	0

Der "acceleration notation index" skaliert Beschleunigungs-Sollwerte, deren Einheiten im "acceleration dimension index" in SI-Einheiten definiert sind, als 10er Potenzen.

Beziehung zwischen den Objektwerten und dem herstellerspezifischen Parameter ACCUNIT:

Wert von Objekt 608Dh	ASCII Parameter ACCUNIT	Skalierung
0	1,5	1
FAh	3	$10^{-6}$
FDh	4	$10^{-3}$

### 5.5.4.2 Objekt 608Eh: acceleration dimension index (DS402)

Index	<b>608Eh</b>
Name	acceleration dimension index
Objektcode	VAR
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED8
Defaultwert	AEh

Der "acceleration dimension index" definiert die SI Einheiten der verwendeten Beschleunigungs Sollwerte. Beziehung zwischen den Objektwerten und dem herstellerspezifischen Parameter ACCUNIT:

Wert von Objekt 608Eh	ASCII Parameter ACCUNIT	SI Einheit
AEh	1	rad/s <sup>2</sup>
55h	3, 4, 5	m/s

Der Parameter ACCUNIT kann im Servoverstärker gespeichert werden. Die Werte von Objekt 608Dh und 608Eh werden von diesem Parameter initialisiert. Nur die beschriebenen Werte für ACCUNIT sind mit dem Profil DS402 möglich.

### 5.5.4.3 Objekt 6097h: Acceleration factor (DS402)

Der Beschleunigungsfaktor konvertiert die Beschleunigung (Beschleunigungseinheit / s) in das interne Format (Inkr./s). Dieser Faktor wird z.Zt. von Objekt 6093 berechnet und ist nur lesbar.

$\text{acceleration factor} = (\text{velocity unit} * \text{velocity encoder factor}) / (\text{acceleration unit} * \text{second})$

Index	<b>6097h</b>
Name	Acceleration factor
Objekt Code	ARRAY
Datentyp	UNSIGNED 32
Kategorie	optional
Subindex	<b>0</b>
Name	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	2
Defaultwert	2
Subindex	<b>1</b>
Name	Zähler (velocity unit * velocity encoder factor)
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	möglich
Wertebereich	UNSIGNED 32
Defaultwert	1
Subindex	<b>2</b>
Name	Nenner (acceleration unit * second)
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	möglich
Wertebereich	UNSIGNED 32
Defaultwert	1

## 5.6 Profile Velocity Mode (pv) (DS402)

### 5.6.1 Allgemeine Informationen

Der Profile Velocity Mode ermöglicht die Verarbeitung von Geschwindigkeitssollwerten und den zugehörigen Beschleunigungen.

### 5.6.2 Objekte, die in diesem Kapitel definiert werden

Index	Objekt	Name	Typ	Zugriff
606Ch	VAR	velocity actual value	INTEGER32	r
60FFh	VAR	target velocity	INTEGER32	rw

### 5.6.3 Objekte, die in anderen Kapiteln definiert werden

Index	Objekt	Name	Typ	Section
6040h	VAR	control word	INTEGER16	dc (→ # 58)
6041h	VAR	status word	UNSIGNED16	dc (→ # 60)
6063h	VAR	position actual value*	INTEGER32	pc (→ # 74)
6083h	VAR	profile acceleration	UNSIGNED32	pp (→ # 88)
6084h	VAR	profile deceleration	UNSIGNED32	pp (→ # 88)
6086h	VAR	motion profile type	INTEGER16	pp (→ # 89)
6094h	ARRAY	velocity encoder factor	UNSIGNED32	fg (→ # 68)

### 5.6.4 Objektbeschreibung

#### 5.6.4.1 Objekt 606Ch: Velocity actual value (DS402)

Das Objekt "velocity actual value" repräsentiert die aktuelle Drehzahl. Die Skalierung des Wertes hängt von dem Faktor "velocity encoder resolution" (Objekt 6094h) ab.

Index	<b>606Ch</b>
Name	velocity actual value
Objektcode	VAR
Datentyp	INTEGER32
Mode	pv
Zugriff	ro
PDO Mapping	möglich
Einheit	Geschwindigkeits Einheit
Wertebereich	$(-2^{31})$ bis $(2^{31}-1)$
Defaultwert	—
EEPROM	nein

#### 5.6.4.2 Objekt 60FFh: Target velocity (DS402)

Die Soll Drehzahl (target velocity) repräsentiert den Sollwert für den Rampengenerator. Die Skalierung des Wertes hängt von dem Faktor "velocity encoder resolution" (Objekt 6094h) ab.

Index	<b>60FFh</b>
Name	target velocity
Objektcode	VAR
Datentyp	INTEGER32
Mode	pv
Zugriff	rw
PDO Mapping	möglich
Einheit	increments
Wertebereich	$(-2^{31})$ bis $(2^{31}-1)$
Defaultwert	—
EEPROM	nein



## 5.7 Profile Torque Mode (tq) (DS402)

### 5.7.1 Allgemeine Informationen

Der profile torque Modus ermöglicht die Verarbeitung von Drehmomentsollwerten und den zugehörigen Strömen.

### 5.7.2 Objekte, die in diesem Kapitel definiert werden

Index	Objekt	Name	Typ	Zugriff
6071h	VAR	Target torque	INTEGER16	rw
6073h	VAR	Max current	UNSIGNED16	rw
6077h	VAR	Torque actual value	INTEGER16	ro

### 5.7.3 Objektbeschreibung

#### 5.7.3.1 Objekt 6071h: Target torque (DS402)

Dieser Parameter ist der Eingangswert für den Drehmomentregler im profile torque Modus. Die Skalierung ist 1/1000 des Nenndrehmomentes.

Index	<b>6071h</b>
Name	Target torque
Objektcode	VAR
Datentyp	INTEGER16
Kategorie	konditional; obligatorisch, wenn tq unterstützt wird
Zugriff	rw
PDO Mapping	möglich
Wertebereich	INTEGER16
Defaultwert	0

#### 5.7.3.2 Objekt 6073h: Max current (DS402)

Dieser Parameter repräsentiert den max. erlaubten, Drehmoment erzeugenden Strom im Motor. Die Skalierung ist 1/1000 des Nennstroms.

Index	<b>6073h</b>
Name	Max current
Objektcode	VAR
Datentyp	UNSIGNED16
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED16
Defaultwert	0

#### 5.7.3.3 Objekt 6077h: Torque actual value (DS402)

Das aktuelle Drehmoment bezieht sich auf das augenblickliche Drehmoment im Motor. Die Skalierung ist 1/1000 des Nenndrehmomentes.

Index	<b>6077h</b>
Name	Torque actual value
Objektcode	VAR
Datentyp	INTEGER16
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	INTEGER16
Defaultwert	0

## 5.8 Position Control Function (pc) (DS402)

### 5.8.1 Allgemeine Informationen

In diesem Kapitel werden die Positionswerte beschrieben, die im Zusammenhang mit dem Lageregler des Antriebs stehen. Sie finden Verwendung im Profile Position Mode.

### 5.8.2 Objekte, die in diesem Kapitel definiert werden

Index	Objekt	Name	Typ	Zugriff
6063h	VAR	position actual value*	INTEGER32	r
6064h	VAR	position actual value	INTEGER32	r
6065h	VAR	following error window	UNSIGNED32	rw
6067h	VAR	position window	UNSIGNED32	rw
6068h	VAR	position window time	UNSIGNED16	rw

### 5.8.3 Objekte, die in anderen Kapiteln definiert werden

Index	Objekt	Name	Typ	Kapitel
607Ah	VAR	target position	INTEGER32	pp (→ # 86)
607Ch	VAR	home-offset	INTEGER32	hm (→ # 82)
607Dh	ARRAY	software position limit	INTEGER32	pp (→ # 86)
607Fh	VAR	max. profile velocity	UNSIGNED32	pp (→ # 87)
6093h	VAR	position factor	UNSIGNED32	fg (→ # 67)
6094h	ARRAY	velocity encoder factor	UNSIGNED32	fg (→ # 68)
6040h	VAR	control word	INTEGER16	dc (→ # 58)
6041h	VAR	status word	UNSIGNED16	dc (→ # 60)

### 5.8.4 Objektbeschreibung

#### 5.8.4.1 Objekt 6063h: position actual value\* (DS402)

Das Objekt Aktueller Positionswert\* liefert die aktuelle Istposition in Inkrementen. Die Auflösung pro Umdrehung wird über Objekt 608F in Zweierpotenzen vorgegeben (s. ASCII-Kommando PRBASE).

Index	<b>6063h</b>
Name	position actual value
Objektcode	VAR
Datentyp	INTEGER32
Mode	pc, pp
Zugriff	rw
PDO Mapping	möglich
Einheit	Inkmente (1 Umdr. = $2^{PRBASE}$ )
Wertebereich	$(-2^{31})$ bis $(2^{31}-1)$
Defaultwert	$2^{20}$
EEPROM	nein

**5.8.4.2 Objekt 6064h: position actual value (DS402)**

Das Objekt Aktueller Positionswert liefert die aktuelle Istposition. Die Auflösung kann mit den Getriebefaktoren des Lagereglers geändert werden (Objekt 6092).

Index	<b>6064h</b>
Name	position actual value
Objektcode	VAR
Datentyp	INTEGER32
Mode	pc, pp
Zugriff	rw
PDO Mapping	möglich
Einheit	Positionseinheiten
Wertebereich	$(-2^{31})$ bis $(2^{31}-1)$
Defaultwert	—
EEPROM	nein

**5.8.4.3 Objekt 6065h: Following error window**

Das Schleppfehlerfenster definiert eine tolerierte Positionswerte symmetrisch zum Positionswert. Ein Schleppfehler kann bei blockiertem Antrieb auftreten, bei unerreichbarer Drehzahlvorgabe oder bei fehlerhaften Reglerparametern. Wenn der eingestellte Wert des Schleppfehlerfensters 0 ist, ist die Überwachung abgeschaltet.

Index	<b>6065h</b>
Name	Following error window
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED32
Defaultwert	1/4 einer Motorumdrehung

**5.8.4.4 Objekt 6067h: Position window (DS402)**

Das Positionsfenster definiert einen symmetrischen Bereich tolerierter Positionen relativ zur Zielposition. Wenn der aktuelle Wert des Positionsgebers im Positionsfenster liegt, wird diese Zielposition als erreicht betrachtet. Das Statuswort Bit "Target reached" wechselt auf 1.

Index	<b>6067h</b>
Name	Position window
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED32
Defaultwert	4000 Positionseinheiten

**5.8.4.5 Objekt 6068h: Position window time (DS402)**

Wenn die aktuelle Position während der definierten Zeit (Vielfaches von 1ms) im Positionsfenster liegt, wird das zugehörige Bit 10 "Target reached" im Statuswort auf 1 gesetzt.

Index	<b>6068h</b>
Name	Position window time
Objektcode	VAR
Datentyp	UNSIGNED16
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED16
Defaultwert	nein

**5.8.4.6 Objekt 60F4h: Following error actual value (DS402)**

Dieses Objekt liefert den aktuellen Wert des Schleppfehlers in benutzerdefinierten Einheiten.

Index	<b>60F4h</b>
Name	Following error actual value
Objektcode	VAR
Datentyp	Integer32
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	INTEGER32
Defaultwert	0

## 5.9 Interpolated Position Mode (ip) (DS402)

### 5.9.1 Allgemeine Informationen

Der interpolierte Positionsmodus ist auf eine einfache, geradlinige Weise realisiert. Einzelne Positionssollwerte müssen im Interpolationszeitraum übertragen werden und werden bei jedem definierten, gesendeten SYNC Telegramm übernommen. Eine lineare Interpolation wird zwischen den Sollwerten verwendet.

### 5.9.2 Objekte, die in diesem Kapitel definiert werden

Index	Objekt	Name	Typ	Zugriff
60C0h	VAR	Interpolation sub mode select	INTEG16ER	rw
60C1h	ARRAY	Interpolation data record	INTEGER32	rw
60C2h	RECORD	Interpolation time period	Interpolation time period	rw
60C3h	ARRAY	Interpolation sync definition	UNSIGNED8	rw
60C4h	RECORD	Interpolation data configuration record	Interpolation data configuration record	rw

### 5.9.3 Objektbeschreibung

#### 5.9.3.1 Objekt 60C0h: Interpolation sub mode select

Im den S300/S700 wird die lineare Interpolation zwischen Positionssollwerten unterstützt. Der einzige erlaubte Wert ist 0.

Index	<b>60C0h</b>
Name	Interpolation sub mode select
Objektcode	VAR
Datentyp	INTEGER16
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	0
Defaultwert	0

### 5.9.3.2 Objekt 60C1h: Interpolation data record

Im S300/S700 wird nur ein einzelner Sollwert für den interpolierten Positionsmodus unterstützt. Beim linearen Interpolationsmodus kann jeder Interpolationsdatensatz (Record) als neuer Positionssollwert betrachtet werden.

Nachdem das letzte Element des Records in den Geräteeingangspuffer geschrieben wurde, wird der Zeiger des Speichers automatisch auf die nächste Speicherposition gesetzt.

Index	<b>60C1h</b>
Name	Interpolation data record
Objektcode	ARRAY
Datentyp	INTEGER32
Kategorie	optional

Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	1
Defaultwert	nein

Subindex	<b>1</b>
Beschreibung	x1, der erste Parameter der ip Funktion $fip(x1, \dots, xN)$
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	möglich
Wertebereich	INTEGER32
Defaultwert	nein

### 5.9.3.3 Objekt 60C2h: Interpolation time period

Der Interpolationszeitraum wird für den PLL (phase locked loop) synchronisierten Positionierbetrieb verwendet. Die Einheit (Subindex 1) der Zeit ist mit  $10^{\text{interpolation time index}s}$  gegeben, nur Vielfache von 1ms sind erlaubt.

Die zwei Werte definieren den internen ASCII-Parameter PTBASE (Vielfache von 250  $\mu$ s). Es müssen immer beide Werte geschrieben werden um einen neuen Interpolationszeitraum festzulegen. PTBASE wird erst dann geändert.

Index	<b>60C2h</b>
Name	Interpolation time period
Objektcode	RECORD
Datentyp	Interpolation time period record (0080h)
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	2
Defaultwert	2
Subindex	<b>1</b>
Beschreibung	Interpolation time units
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED8
Defaultwert	1
Subindex	<b>2</b>
Beschreibung	Interpolation time index
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	möglich
Wertebereich	-123 ... 63
Defaultwert	-3

**5.9.3.4 Objekt 60C3h: Interpolation sync definition**

Im S300/S700 wird das allgemein verwendete SYNC-Objekt zur Synchronisation verwendet. Daher wird nur ein fester Wert für Subindex 1 akzeptiert.

Index	<b>60C3h</b>
Name	Interpolation sync definition
Objektcode	ARRAY
Datentyp	UNSIGNED8
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	2
Defaultwert	1
Subindex	<b>1</b>
Beschreibung	Synchronize on group
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED8
Defaultwert	0

**5.9.3.5 Objekt 60C4h: Interpolation data configuration**

Nur ein einzelner Positionssollwert wird im S300/S700 unterstützt. Daher ist nur der Wert 1 in Subindex 5 möglich. Alle anderen Subindizes werden auf 0 gesetzt.

Index	<b>60C4h</b>
Name	Interpolation data configuration
Objektcode	RECORD
Datentyp	Interpolation data configuration record (0081h)
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	6
Defaultwert	6
Subindex	<b>1</b>
Beschreibung	Maximum buffer size
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	0



Subindex	<b>2</b>
Beschreibung	Actual buffer size
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED32
Defaultwert	0
Subindex	<b>3</b>
Beschreibung	Buffer organization
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED8
Defaultwert	0
Subindex	<b>4</b>
Beschreibung	Buffer position
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED16
Defaultwert	0
Subindex	<b>5</b>
Beschreibung	Size of data record
Kategorie	obligatorisch
Zugriff	w
PDO Mapping	möglich
Wertebereich	1 to 254
Defaultwert	1
Subindex	<b>6</b>
Beschreibung	Buffer clear
Kategorie	obligatorisch
Zugriff	w
PDO Mapping	möglich
Wertebereich	UNSIGNED8
Defaultwert	0

## 5.10 Homing Mode (hm) (DS402)

### 5.10.1 Allgemeine Informationen

Dieses Kapitel beschreibt die verschiedenen Parameter, die zur Definition einer Referenzierung benötigt werden.

### 5.10.2 Objekte, die in diesem Kapitel definiert werden

Index	Objekt	Name	Typ	Zugriff
607Ch	VAR	home offset	INTEGER32	rw
6098h	VAR	homing method	INTEGER8	rw
6099h	ARRAY	homing speeds	UNSIGNED32	rw
609Ah	VAR	homing acceleration	UNSIGNED32	rw

### 5.10.3 Objekte, die in anderen Kapiteln definiert werden

Index	Objekt	Name	Typ	Section
6040h	VAR	control word	INTEGER16	dc (→ # 58)
6041h	VAR	status word	UNSIGNED16	dc (→ # 60)

### 5.10.4 Objektbeschreibung

#### 5.10.4.1 Objekt 607Ch: Homing offset (DS402)

Der Referenzoffset ist die Differenz zwischen der Nullposition der Anwendung und des Maschinennullpunktes. Alle nachfolgenden absoluten Fahraufträge berücksichtigen den Referenzoffset.

Index	<b>607Ch</b>
Name	home offset
Objektcode	VAR
Datentyp	INTEGER32
Mode	hm
Zugriff	rw
PDO Mapping	möglich
Einheit	benutzerdefiniert
Wertebereich	$(-2^{31})$ bis $(2^{31}-1)$
Defaultwert	0

## 5.10.4.2 Objekt 6098h: Homing method (DS402)

Index	<b>6098h</b>
Name	homing method
Objektcode	VAR
Datentyp	INTEGER8
Mode	hm
Zugriff	rw
PDO Mapping	möglich
Einheit	Positionseinheiten
Wertebereich	-128 bis 127
Defaultwert	0
EEPROM	ja

Die folgenden Referenzfahrtarten (Methoden nach DS402) werden unterstützt:

DS402	Kurzbeschreibung Referenzfahrt	ASCII command
-128..-9	reserviert	—
-8	Fahren auf die absolute SSI-Position	—
-7	Fahren auf mechanischen Anschlag, Schleppfehler setzt Referenz	NREF=9, DREF=0
-6	Fahren auf mechanischen Anschlag, Schleppfehler setzt Referenz	NREF=9, DREF=1
-5, -4	reserviert	—
-3	Fahren auf mechanischen Anschlag, mit Nullpunktsuche	NREF=7
-2	Setzen des Referenzpunktes auf die aktuelle Position unter Berücksichtigung des Schleppabstandes	NREF=6
-1	Referenzfahrt innerhalb einer Umdrehung (Drehrichtung entfernungsabhängig)	NREF=5, DREF= 2
0	reserviert	—
1	Referenzfahrt auf negativen Endschalter, mit Nullpunktsuche, Fahrtrichtung negativ	NREF=2, DREF= 0
2	Referenzfahrt auf positiven Endschalter, mit Nullpunktsuche, Fahrtrichtung positiv	NREF=2, DREF= 1
3..7	werden nicht unterstützt	—
8	Referenzfahrt mit Referenzschalter, mit Nullpunktsuche, Fahrtrichtung positiv	NREF=1, DREF= 1
9..11	werden nicht unterstützt	—
12	Referenzfahrt mit Referenzschalter, mit Nullpunktsuche, Fahrtrichtung negativ	NREF=1, DREF= 0
13..14	werden nicht unterstützt	—
15..16	reserviert	—
17	Referenzfahrt auf negativen Endschalter, ohne Nullpunktsuche, Fahrtrichtung negativ	NREF=4, DREF= 0
18	Referenzfahrt auf negativen Endschalter, ohne Nullpunktsuche, Fahrtrichtung positiv	NREF=4, DREF= 1
19..23	werden nicht unterstützt	—
24	Referenzfahrt auf Referenzschalter, ohne Nullpunktsuche, Fahrtrichtung positiv	NREF=3, DREF= 1
25..27	werden nicht unterstützt	—
28	Referenzfahrt auf Referenzschalter, ohne Nullpunktsuche, Fahrtrichtung negativ	NREF=3, DREF= 0
29..30	werden nicht unterstützt	—
31..32	reserviert	—
33	Referenzfahrt innerhalb einer Umdrehung, Fahrtrichtung negativ	NREF=5, DREF= 0
34	Referenzfahrt innerhalb einer Umdrehung, Fahrtrichtung positiv	NREF=5, DREF= 1
35	Setzen des Referenzpunktes an die aktuelle Position	NREF=0
36..127	reserviert	—

**Beschreibung der Referenziermethoden**

Durch die Auswahl einer Referenzfahrtart durch Beschreiben des Parameters homing method (Objekt 6098h) wird folgendes bestimmt:

- das Referenzsignal (PStop, NStop, Referenzschalter)
- die Richtung der Referenzfahrt

und, wenn vorhanden

- die Position des Nullimpulses.

Die Referenzposition wird durch den Referenzoffset (Objekt 607Ch) festgelegt. Zur Anpassung der Grundeinstellung der Motorlage bei Referenzierung auf den Nullimpuls kann der herstellerspezifische Parameter ENCZERO (Objekt 3537h, Subindex 01h) verwendet werden.

Eine ausführliche Beschreibung der Referenzfahrtarten finden Sie in der Beschreibung der Inbetriebnahmesoftware DRIVEGUI.EXE.

**5.10.4.3 Objekt 6099h: Homing speeds (DS402)**

Index	<b>6099h</b>
Name	homing speeds
Objektcode	ARRAY
Datentyp	UNSIGNED32
Subindex	<b>1</b>
Beschreibung	speed during search for switch
Mode	hm
Zugriff	rw
PDO Mapping	möglich
Einheit	Geschwindigkeitseinheit
Wertebereich	0 bis $(2^{32}-1)$
Defaultwert	entspricht 60 U/min
EEPROM	ja
Subindex	<b>2</b>
Beschreibung	speed during search for zero
Mode	hm
Zugriff	rw
PDO Mapping	nicht möglich
Einheit	Geschwindigkeitseinheit
Wertebereich	0 bis $(2^{32}-1)$
Defaultwert	$1/8 * \text{Objekt } 6099 \text{ sub } 1$
EEPROM	ja

**5.10.4.4 Objekt 609Ah: Homing acceleration (DS402)**

Index	<b>609Ah</b>
Name	homing acceleration
Objektcode	VAR
Datentyp	UNSIGNED32
Mode	hm
Zugriff	rw
PDO Mapping	möglich
Einheit	Beschleunigungseinheit
Wertebereich	0 bis $(2^{32}-1)$
Defaultwert	0
EEPROM	ja

### 5.10.5 Homing Mode Sequence

Mit Setzen des Bit 4 (positive Flanke) wird die Referenzfahrt gestartet. Der erfolgreiche Abschluss wird mit Bit 12 im Zustandswort angezeigt (s. Objekt 6041h). Bit 13 zeigt einen Fehler an, der sich während der Referenzfahrt ereignet hat. Hier ist der Fehlercode auszuwerten:  
Error register (Objekte 1001h, 1003h), manufacturer status (Objekt1002h)

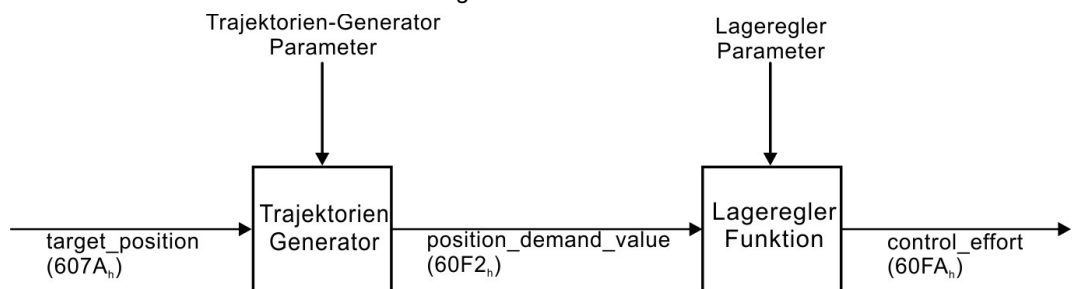
Bit 4	Bedeutung
0	Referenzfahrt inaktiv
0 → 1	Referenzfahrt starten
1	Referenzfahrt aktiv
1 → 0	Unterbrechung der Referenzfahrt

Bit 13	Bit 12	Bedeutung
0	0	Referenzpunkt nicht gesetzt bzw. Referenzfahrt noch nicht abgeschlossen
0	1	Referenzpunkt gesetzt bzw. Referenzfahrt erfolgreich abgeschlossen
1	0	Referenzfahrt konnte nicht erfolgreich abgeschlossen werden. (Schleppfehler)
1	1	kein erlaubter Zustand

## 5.11 Profile Position Mode (pp)

### 5.11.1 Allgemeine Informationen

Die Struktur dieser Betriebsart ist hier dargestellt:



Die spezielle Handshake-Verarbeitung von Controlword und Statusword wird auf Seite (→ # 90).

### 5.11.2 Objekte, die in diesem Kapitel definiert werden

Index	Objekt	Name	Typ	Zugriff
607Ah	VAR	target position	INTEGER32	rw
607Dh	ARRAY	software position limit	INTEGER32	rw
607Fh	VAR	max. profile velocity	UNSIGNED32	rw
6080h	VAR	max. motor speed	UNSIGNED32	rw
6081h	VAR	profile velocity	UNSIGNED32	rw
6083h	VAR	profile acceleration	UNSIGNED32	rw
6084h	VAR	profile deceleration	UNSIGNED32	rw
6085h	VAR	quick stop deceleration	UNSIGNED32	rw
6086h	VAR	motion profile type	INTEGER16	rw

### 5.11.3 Objekte, die in anderen Kapiteln definiert werden

Index	Objekt	Name	Typ	Section
6040h	VAR	control word	INTEGER16	dc (→ # 58)
6041h	VAR	status word	UNSIGNED16	dc (→ # 60)
6093h	ARRAY	position factor	UNSIGNED32	fg (→ # 67)
6094h	ARRAY	velocity encoder factor	UNSIGNED32	fg (→ # 68)
6097h	ARRAY	acceleration factor	UNSIGNED32	fg (→ # 71)

## 5.11.4 Objektbeschreibung

### 5.11.4.1 Objekt 607Ah: Target position (DS402)

Das Objekt definiert die Zielposition des Antriebes. Abhängig vom Bit 6 im controlword wird die Zielposition als relativer Weg oder als absolute Position interpretiert. Dabei kann die Art der Relativfahrt durch den Parameter 35B9h Subindex 1 weiter aufgeschlüsselt werden. Die mechanische Auflösung wird über die Getriebefaktoren Objekt 6093h Subindex 01h und 02h eingestellt.

Index	<b>607Ah</b>
Name	target position
Objektcode	VAR
Datentyp	INTEGER32
Mode	pp
Zugriff	rw
PDO Mapping	möglich
Einheit	benutzerdefiniert
Wertebereich	$-(2^{31}-1)$ bis $(2^{31}-1)$
Defaultwert	nein
EEPROM	nein

### 5.11.4.2 Objekt 607Dh: Software position limit (DS402)

Das Objekt beinhaltet die Subindizes min position limit und max position limit. Neue Zielpositionen werden bezogen auf diese Grenzen geprüft. Die Grenzen sind relativ zum Maschinennullpunkt, der während der Referenzfahrt einschließlich des Referenz-Offsets definiert wurde (607C). Im Auslieferungszustand sind die Software-Endschalter ausgeschaltet. Wenn Sie die Werte ändern, wird im Servoverstärker eine spezielle Konfiguration durchgeführt. Daher müssen Sie die neuen Werte speichern und den Verstärker neu starten, um die SW-Endschalter zu aktivieren.

Index	<b>607Dh</b>
Name	Software position limit
Objektcode	ARRAY
Datentyp	INTEGER32
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Kategorie	obligatorisch
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	2
Defaultwert	2
Subindex	<b>1</b>
Beschreibung	min position limit
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	möglich
Wertebereich	INTEGER32
Defaultwert	0 (abgeschaltet)
Subindex	<b>2</b>
Beschreibung	Min Position Limit
Kategorie	obligatorisch
Zugriff	rw
PDO Mapping	möglich
Wertebereich	INTEGER32
Defaultwert	0 (abgeschaltet)

**5.11.4.3 Objekt 607Fh: Max profile velocity (DS402)**

Die maximale Verfahrgeschwindigkeit ist die maximal zulässige Geschwindigkeit in jeder Richtung einer definierten Bewegung. Einheit analog zur Verfahrgeschwindigkeit.

Index	<b>607Fh</b>
Name	Max profile velocity
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED32
Defaultwert	entspricht der maximalen Motordrehzahl (Objekt 6080)

**5.11.4.4 Objekt 6080h: Max motor speed (DS402)**

Die maximale Motordrehzahl ist die maximal zulässige Drehzahl in jede Richtung und wird in U/min angegeben. Damit wird der Motor geschützt, der korrekte Wert ist bei den Motordaten angegeben.

Index	<b>6081h</b>
Name	Max motor speed
Objektcode	VAR
Datentyp	UNSIGNED32
Mode	optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED32
Defaultwert	abhängig vom Motor

**5.11.4.5 Objekt 6081h: Profile velocity (DS402)**

Die Profilvergeschwindigkeit ist die Endgeschwindigkeit, die nach der Beschleunigungsphase eines Fahrauftrages erreicht werden soll. Die verwendete Skalierung ist abhängig von dem eingestellten "velocity encoder factor" (Objekt 6094h). Der Sollwert wird in Abhängigkeit des eingestellten Operationsmode (pp, pv) benutzt.

Index	<b>6081h</b>
Name	profile velocity
Objektcode	VAR
Datentyp	UNSIGNED32
Mode	pp, pv
Zugriff	rw
PDO Mapping	möglich
Einheit	Geschwindigkeitseinheit
Wertebereich	0 bis $(2^{32}-1)$
Defaultwert	10
EEPROM	nein

**5.11.4.6 Objekt 6083h: Profile acceleration (DS402)**

Die Beschleunigungsrampe (profile acceleration) wird in Einheiten, die der Anwender definiert hat, vorgegeben (Positionswerte / s<sup>2</sup>). Sie können mit dem Beschleunigungsfaktor (acceleration factor, definiert durch Objekt 6097 sub 1 und 2) umgewandelt werden.

Die Art der Beschleunigungsrampe kann als lineare Rampe oder als Sin<sup>2</sup> Rampe ausgewählt werden (siehe Objekt 6086h).

Index	<b>6083h</b>
Name	profile acceleration
Objektcode	VAR
Datentyp	UNSIGNED32
Mode	pp
Zugriff	rw
PDO Mapping	möglich
Einheit	Beschleunigungseinheit
Wertebereich	0 bis (2 <sup>32</sup> -1)
Defaultwert	0

**5.11.4.7 Objekt 6084h: Profile deceleration (DS402)**

Die Bremsrampe wird analog zur Beschleunigungsrampe behandelt. (s. Objekt 6083h)

Index	<b>6084h</b>
Name	profile deceleration
Objektcode	VAR
Datentyp	UNSIGNED32
Mode	pp
Zugriff	rw
PDO Mapping	möglich
Einheit	Beschleunigungseinheit
Wertebereich	0 bis (2 <sup>32</sup> -1)
Defaultwert	0

**5.11.4.8 Objekt 6085h: Quick stop deceleration**

Mit der Schnellstopp-Bremsrampe wird der Motor gebremst, wenn das 'Quick Stop' Kommando ausgelöst wurde.

Index	<b>6085h</b>
Name	Quick stop deceleration
Objektcode	VAR
Datentyp	UNSIGNED32
Mode	pp
Zugriff	rw
PDO Mapping	möglich
Einheit	Beschleunigungseinheit
Wertebereich	0 bis (2 <sup>32</sup> -1)
Defaultwert	-



**5.11.4.9 Objekt 6086h: Motion Profile Type (DS402)**

Die Art der Beschleunigungsrampe kann mit diesem Objekt als lineare Rampe oder als Sin<sup>2</sup> Rampe ausgewählt werden.

Index	<b>6086h</b>
Name	Motion Profile Type
Objektcode	VAR
Datentyp	INTEGER16
Mode	pp
Zugriff	rw
PDO Mapping	möglich
Einheit	keine
Wertebereich	-2 <sup>15</sup> bis (2 <sup>15</sup> -1)
Defaultwert	-
EEPROM	ja

profile code	profile type
-32768...-1	herstellerspez. (wird nicht unterstützt)
0	linear (trapez)
1	sin <sup>2</sup>
2...32767	profilspez. Erweiterungen (wird nicht unterstützt)

**5.11.4.10 Objekt 60C5h: Max acceleration**

Die maximale Beschleunigung wird verwendet um Motor und Applikation zu schützen. Die Einheit ist benutzerspezifisch definiert (608Dh, 608Eh). Die Einheit wird mit dem Objekt Acceleration Factor (6097h) in Positions-Inkrementen pro s<sup>2</sup> konvertiert.

Index	<b>60C5h</b>
Name	Max acceleration
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	Optional
Zugriff	rw
PDO Mapping	möglich
Wertebereich	UNSIGNED32
Defaultwert	-

### 5.11.5 Funktionelle Beschreibung

In diesem Profil werden zwei Wege der Positionssollwertübergabe an den Antrieb unterstützt.

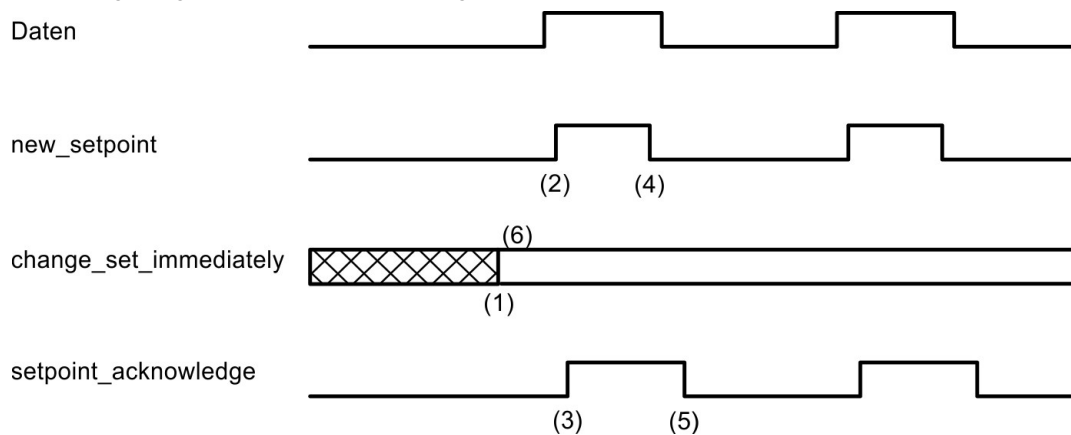
#### Eine Folge von Sollwerten:

Nach Erreichen der Zielposition berechnet der Antrieb sofort die Bewegung zur vorher übergebenen neuen Zielposition. Dies führt zu einer kontinuierlichen Bewegung, ohne dass der Antrieb zwischendurch auf Geschwindigkeit 0 abbremst. Beim S300/S700 ist dies nur bei Verwendung von Trapezrampen möglich.

#### Einzelne Sollwerte:

Nach Erreichen der Zielposition signalisiert der Antrieb an den Master, dass er das Ziel erreicht hat und erhält dann einen neuen Sollwert. Nach Erreichen der Zielposition ist die Geschwindigkeit normalerweise 0 bevor die Achse zu einer neuen Zielposition fährt.

Die zwei Arten werden über das Timing der Bits "new\_setpoint", "change\_set\_immediately" des Steuerworts und das Bit "setpoint\_acknowledge" des Zustandswort gesteuert. Diese Bits ermöglichen einen Handshake-Mechanismus, der es ermöglicht einen neuen Sollwert vorzubereiten während ein alter Fahrauftrag ausgeführt wird. Das verringert Reaktionszeiten innerhalb eines Steuerungsprogramms in einer Steuerung.

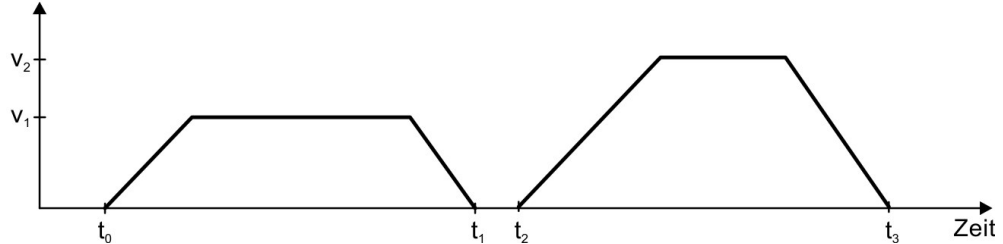


Die Bilder zeigen den Unterschied zwischen einer "Folge von Sollwerten" und einzelnen Sollwerten. Der Startwert des Bits "change\_set\_immediatly" im Steuerwort entscheidet über die verwendete Art. Die betrachteten Beispiele gelten nur für Trapezrampen.

Wenn das Bit **change\_set\_immediately = 0** ist, wird ein einzelner Sollwert vom Antrieb erwartet (1). Nachdem Daten an den Antrieb übergeben wurden, signalisiert der Master durch den Wechsel am Bit "new\_setpoint" im Steuerwort auf "1", dass die Daten gültig sind (2). Der Antrieb antwortet mit dem "setpoint\_acknowledge" Bit = 1 im Zustandswort nachdem er den Wert erkannt und gespeichert hat (3). Nun kann der Master das Bit "new\_setpoint" auf 0 setzen (4) woraufhin der Antrieb durch Rücksetzen des Bits "setpoint\_acknowledge" signalisiert, dass er wieder neue Sollwerte entgegennehmen kann (5).

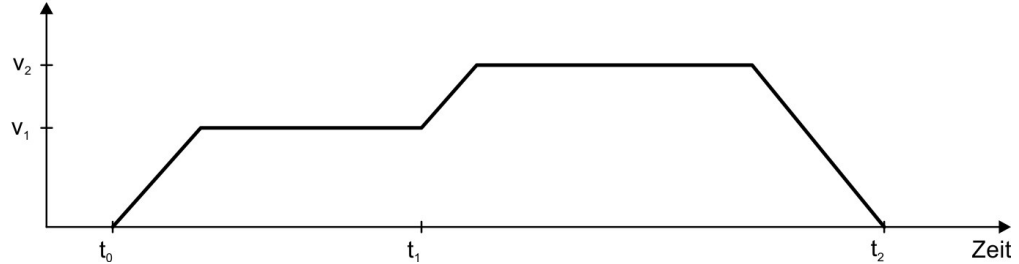
Im Bild unten führt dies zu einer Geschwindigkeit von 0, nachdem eine Rampe gefahren wurde, um die Zielposition X1 zur Zeit t1 zu erreichen. Nach dem Signal an den Master, dass das Ziel erreicht wurde, wird die neue Zielposition zum Zeitpunkt t2 verarbeitet und zum Zeitpunkt t3 erreicht.

Drehzahl



Mit dem Bit **change\_set\_immediatly = 1** weist der Master den Antrieb an, direkt nach dem Erreichen des letzten Sollwerts einen neuen Sollwert zu verarbeiten. Das Timing der Signale bleibt gleich. Diese Vorgehensweise bewirkt beim Antrieb, dass er schon den nächsten Sollwert X2 verarbeitet und Geschwindigkeit behält, wenn er die Zielposition X1 zum Zeitpunkt t1 erreicht. Danach fährt der Antrieb zur schon berechneten Zielposition X2.

Drehzahl



Bits im control word:		Bits im status word:	
Bit 4	new_set_point (positive edge!)	Bit 12	setpoint acknowledge
Bit 5	change_set_immediatly	Bit 13	lag/following error
Bit 6	absolute/relative		

**Hinweis zur Fahrauftragsart "relativ":**

Wird das Bit 6 gesetzt, ist die Fahrauftragsart "relativ in Abhängigkeit zur letzten Zielposition oder Istposition" aktiviert. Sollten andere Relativfahrtarten gewünscht sein, müssen diese im Vorfeld mit dem ASCII Objekt O\_C (Objekt 35B9 sub 1) aktiviert werden.

**Hinweis zum Mode Profile Position Mode:**

Funktionale Beschreibung für den Mode: Profile Position Mode

Das Antriebsprofil DS402 unterscheidet zwei Methoden Zielpositionen anzufahren. Diese beiden Methoden werden über die Bits "new\_setpoint" und "change\_set\_immediatly" im controlword und "setpoint\_acknowledge" im statusword gesteuert. Mit Hilfe dieser Bits kann ein Fahrauftrag aufgebaut werden, während ein anderer gerade ausgeführt wird (Handshake).

- Anfahren von mehreren Zielpositionen ohne Zwischenstopp  
Nachdem die Zielposition erreicht wurde, wird sofort die nächste Zielposition angefahren. Voraussetzung ist, dass dem Antrieb neue Sollwerte signalisiert werden. Dies wird mit einer positiven Flanke (Bit "new\_setpoint") realisiert. Das Bit "setpoint\_acknowledge" im Statusword darf hierbei nicht mehr aktiv (= 1) sein (siehe auch Handshake DS402). Die Geschwindigkeit wird, nach Erreichen des ersten Zielpunktes, nicht auf Null reduziert.
- Anfahren von einer einzelnen Zielpositionen  
Der Antrieb fährt in die Zielposition, wobei die Geschwindigkeit auf Null reduziert wird. Das Erreichen der Zielposition wird mit dem Bit "target\_reached" im statusword signalisiert.

## 6 Anhang

### 6.1 Der Objektkanal

#### 6.1.1 Objekte >3500h Herstellerspezifischer Objektkanal

Das Objektverzeichnis ist ab Index 3500h (reservierter Objektbereich 3500h– 3900h) um sämtliche Geräteobjekte erweitert worden, die mit bis zu 4 Byte Nutzdaten darstellbar sind. Dieser Bereich ist dynamisch erweiterbar d.h., neue Geräteparameter, die das o. g. Datenformat erfüllen, werden bei Erweiterungen in der "Kernfirmware" automatisch der Tabelle angehängt. Mit 3500h (Sub 1, read) kann die Gesamtanzahl der Objekte im Objektkanal bestimmt werden (→ # 103).

**INFO**

Die Objekte aus dem Objektkanal (ASCII-Parameter) können nicht in ein PDO gemappt werden!

Index	<b>&gt; 3500h</b>
Name	Objektabhängig
Objektcode	VAR
Datentyp	RECORD
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Einheit	—
Zugriff	—
PDO Mapping	nicht möglich
Datentyp	UNSIGNED8
Wertebereich	0 ... 28-1
EEPROM	—
Defaultwert	—
Subindex	<b>1</b>
Beschreibung	read/write a parameter
Einheit	siehe jeweiliges ASCII-Kommando
Zugriff	siehe jeweiliges ASCII-Kommando
PDO Mapping	nicht möglich
Datentyp	siehe jeweiliges ASCII-Kommando, Übertragung immer als INTEGER32
Wertebereich	siehe jeweiliges ASCII-Kommando
EEPROM	siehe Subindex 4
Defaultwert	siehe jeweiliges ASCII-Kommando
Subindex	<b>2</b>
Beschreibung	read lower limit value
Einheit	siehe jeweiliges ASCII-Kommando
Zugriff	ro
PDO Mapping	nicht möglich
Datentyp	siehe jeweiliges ASCII-Kommando
Wertebereich	siehe jeweiliges ASCII-Kommando
EEPROM	—
Defaultwert	—
Subindex	<b>3</b>
Beschreibung	read upper limit value
Einheit	siehe jeweiliges ASCII-Kommando
Zugriff	ro
PDO Mapping	nicht möglich
Datentyp	siehe jeweiliges ASCII-Kommando
Wertebereich	siehe jeweiliges ASCII-Kommando
EEPROM	—
Defaultwert	—

Subindex	<b>4</b>
Beschreibung	read the default value
Einheit	siehe jeweiliges ASCII-Kommando
Zugriff	ro
PDO Mapping	nicht möglich
Datentyp	siehe jeweiliges ASCII-Kommando
Wertebereich	siehe jeweiliges ASCII-Kommando
EEPROM	—
Defaultwert	—

Subindex	<b>5</b>
Beschreibung	read the parameter format
Einheit	—
Zugriff	ro
PDO Mapping	nicht möglich
Datentyp	siehe jeweiliges ASCII-Kommando
Wertebereich	siehe jeweiliges ASCII-Kommando
EEPROM	—
Defaultwert	—

**Mögliche Parameterformate:**

<b>0</b>	Funktion (kein Parameter, nur lesen)	<b>7</b>	INTEGER32
<b>1</b>	Funktion (INTEGER32 Parameter)	<b>8</b>	UNSIGNED32
<b>2</b>	Funktion (INTEGER32 Parameter mit Wichtung 3)	<b>9</b>	INTEGER32 (Wichtung 3)
<b>3</b>	INTEGER8	<b>10</b>	INTEGER32 (Wichtung 3)
<b>4</b>	UNSIGNED8	<b>11</b>	
<b>5</b>	INTEGER16	<b>12</b>	UNSIGNED32
<b>6</b>	UNSIGNED16	<b>13</b>	UNSIGNED16

Subindex	<b>6</b>
Beschreibung	read the parameter check data
Einheit	—
Zugriff	ro
PDO Mapping	nicht möglich
Datentyp	UNSIGNED32
Wertebereich	0 ... $2^{32}-1$
EEPROM	—
Defaultwert	—

**Beschreibung:**

0x00010000 Nach Änderung muss die Variable abgespeichert und der Regler resettiert werden.

0x00020000 Variable wird im seriellen EEPROM abgespeichert.

0x00200000 Variable ist read-only, darf nicht über den Bus geschrieben werden.

Subindex	<b>7 / 8</b>
Beschreibung	reserviert
Einheit	—
Zugriff	ro
PDO Mapping	nicht möglich
Datentyp	UNSIGNED32
Wertebereich	0 ... $2^{32}-1$
EEPROM	—
Defaultwert	—

## 6.1.2 ASCII Kommando Referenz

MLC=mehrzeiliges Kommando, ro=nur lesen, rw=lesen/schreiben

CAN Objekt Nummer	ASCII Kommando	ASCII Typ	Beschreibung
3501 (hex)	ACC	Variable rw	Beschleunigungsrampe Drehzahlregelung
3502 (hex)	ACCR	Variable rw	Beschleunigungsrampe Referenzfahrt/Tippbetrieb
3503 (hex)	ACTFAULT	Variable rw	Fehler Stop Modus
3504 (hex)	ACTIVE	Variable ro	Endstufe freigegeben/gesperrt
3505 (hex)	ADDR	Variable ro	Stationsadresse
3506 (hex)	AENA	Variable rw	Initialisierungszustand der Software-Freigabe
3507 (hex)	ANCNFG	Variable rw	Konfiguration der analogen Eingänge
3508 (hex)	ANDB	Variable rw	Totband für den analogen Drehzahlsollwert
3509 (hex)	ANIN1	Variable ro	Spannung am Analog-Eingang SW1
350A (hex)	ANIN2	Variable ro	Spannung am Analog-Eingang SW2
350B (hex)	ANOFF1	Variable rw	Analogoffset für den Analogeingang analog input 1
350C (hex)	ANOFF2	Variable rw	Analogoffset für den Analogeingang SW2
350F (hex)	ANZERO1	Command	Offsetabgleich für den Analogeingang 1
3510 (hex)	ANZERO2	Command	Offsetabgleich für den Analogeingang SW2
3511 (hex)	AVZ1	Variable rw	Filter-Zeitkonstante für den Analogeingang 1
3512 (hex)	CALCHP	Command	Ermittlung der Hiperface-Parameter
3513 (hex)	CALCRK	Command	Ermittlung der Resolverparameter
3514 (hex)	CALCRP	Command	Ermittlung der Resolverphase
3515 (hex)	CBAUD	Variable rw	Übertragungsrate CAN-Bus
3518 (hex)	CLRFAULT	Command	Löschen des Verstärker-Fehlers
3519 (hex)	CLRHR	Command	Löschen des Bit 5 im Statusregister STAT
351A (hex)	CLRORDER	Command	Löschen eines Fahrsatzes
351B (hex)	CLRWARN	Variable rw	Behandlung der Verstärker-Warnungen
351D (hex)	CONTINUE	Command	Fortsetzen des letzten Fahrauftrages
351E (hex)	CTUNE	Command	Optimierung der Stromreglerparameter
351F (hex)	CUPDATE	Command	Programm-Update über CAN-Bus
3522 (hex)	DEC	Variable rw	Bremsrampe für den Drehzahlsollwert
3523 (hex)	DECDIS	Variable rw	Drehzahl-Bremsrampe beim Sperren der Endstufe
3524 (hex)	DECR	Variable rw	Bremsrampe für Referenzfahrt/Tippbetrieb
3525 (hex)	DECSTOP	Variable rw	Bremsrampe in Nothaltsituation
3527 (hex)	DICONT	Variable ro	Geräte-Nennstrom
3528 (hex)	DIFVAR	MLC	Parameterunterschiede zu den Default-Einstellungen
3529 (hex)	DIPEAK	Variable ro	Geräte-Spitzenstrom
352B (hex)	DIS	Command	Software Disable der Endstufe
352C (hex)	DREF	Variable rw	Fahrtrichtung für die Referenzfahrt
352D (hex)	DRVSTAT	Variable ro	Geräte-Statusinformation
352E (hex)	DR_TYPE	Variable ro	Liefert die Endstufenkennung
352F (hex)	DUMP	MLC	Auflistung aller EEPROM-Variablen
3530 (hex)	EN	Command	Setzen des Software-Enable
3533 (hex)	ENCLINES	Variable rw	Auflösung eines Sincos-Gebers
3534 (hex)	ENCMODE	Variable rw	Auswahl der Encoder-Emulation
3535 (hex)	ENCOUT	Variable rw	Auflösung für die Encoder-Emulation (ROD)
3537 (hex)	ENCZERO	Variable rw	Nullimpulsoffset (ROD-Ausgabe)
3538 (hex)	EXTMUL	Variable rw	Multiplikationsfaktor für den externen Encoder
3539 (hex)	EXTPOS	Variable rw	Positionsquelle der Lageregelung
353A (hex)	EXTWD	Variable rw	Überwachungszeit für Feldbus-Befehle
353B (hex)	FBTYPE	Variable rw	Vorwahl der Kommutierungs-Rückführeinheit
353C (hex)	FILTMODE	Variable rw	Smith Predictor
353E (hex)	GEARI	Variable rw	Eingangsfaktor "elektronisches Getriebe"
353F (hex)	GEARMODE	Variable rw	Masterschnittstelle für "elektronisches Getriebe"
3540 (hex)	GEARO	Variable rw	Ausgangsfaktor "elektronisches Getriebe"

CAN Objekt Nummer	ASCII Kommando	ASCII Typ	Beschreibung
3541 (hex)	GET	Command	Ausgabe der aufgezeichneten SCOPE-Daten
3542 (hex)	GP	Variable rw	Lageregler: Proportionalverstärkung
3545 (hex)	GPFFV	Variable rw	Lageregler: Vorsteuerung Drehzahl
3548 (hex)	GV	Variable rw	Drehzahlregler: Proportionalverstärkung
354B (hex)	GVFR	Variable rw	PI-PLUS Drehzahlwertvorsteuerung
354D (hex)	GVTN	Variable rw	Drehzahlregler: Nachstellzeit
3551 (hex)	HDUMP	MLC	Ausgabe aller SinCos Variablen
3552 (hex)	HICOFFS	Variable rw	Hiperface: Cosinus-Offset (Inkrementalspur)
3553 (hex)	HIFACT1	Variable rw	Hiperface: Sinus/Cosinus-Faktor (Inkrementalspur)
3554 (hex)	HISOFFS	Variable rw	Hiperface: Sinus-Offset (Inkrementalspur)
3556 (hex)	HSAVE	Command	Hiperface: Speichern der Parameter im Geber
3557 (hex)	HVER	Variable ro	Ausgabe der Hardware-Version
3558 (hex)	I	Variable ro	Stromistwert
355A (hex)	I2TLIM	Variable rw	I2t-Meldeschwelle
355B (hex)	ICMD	Variable ro	Stromsollwert
355D (hex)	ID	Variable ro	D-Anteil vom Strom-Istwert
355E (hex)	IDUMP	MLC	Ausgabe der Strom-Grenzwerte
3560 (hex)	IN	MLC	Anzeige der A/D-Spannungen
3561 (hex)	IN1	Variable ro	Zustand des digitalen Eingangs INPUT1
3561 (hex)	IN5_20	Variable ro	Zustand der digitalen Eingänge 5...20
3562 (hex)	IN1MODE	Variable rw	Funktion des digitalen Eingangs INPUT1
3562 (hex)	IN5_20MODE	Variable rw	Funktion der digitalen Eingänge 5...20
3563 (hex)	IN1TRIG	Variable rw	Hilfsvariable für IN1MODE
3563 (hex)	IN5_20TRIG	Variable rw	Hilfsvariable der digitalen Eingänge 5...20
3564 (hex)	IN2	Variable ro	Zustand des digitalen Eingangs INPUT2
3565 (hex)	IN2MODE	Variable rw	Funktion des digitalen Eingangs INPUT2
3566 (hex)	IN2TRIG	Variable rw	Hilfsvariable für IN2MODE
3567 (hex)	IN3	Variable ro	Zustand des digitalen Eingangs INPUT3
3568 (hex)	IN3MODE	Variable rw	Funktion des digitalen Eingangs INPUT3
3569 (hex)	IN3TRIG	Variable rw	Hilfsvariable für IN3MODE
356A (hex)	IN4	Variable ro	Zustand des digitalen Eingangs INPUT4
356B (hex)	IN4MODE	Variable rw	Funktion des digitalen Eingangs INPUT4
356C (hex)	IN4TRIG	Variable rw	Hilfsvariable für IN3MODE
356D (hex)	INPOS	Variable ro	In-Position-Meldung
356E (hex)	IPEAK	Variable rw	Spitzenstrom
3570 (hex)	IQ	Variable ro	Q-Anteil des Stromistwertes
3571 (hex)	ISCALE1	Variable rw	Skalierungsfaktor für den analogen Stromsollwert 1
3572 (hex)	ISCALE2	Variable rw	Skalierungsfaktor für den analogen Stromsollwert 2
3573 (hex)	K	Command	Software-Sperre der Endstufe
3574 (hex)	KC	Variable rw	Stromistwert-Vorsteuerung Stromregler
3575 (hex)	KEYLOCK	Variable rw	Sperre für die Tastenbedienung
3577 (hex)	ML	Variable rw	Statorinduktivität des Motors
3578 (hex)	LATCH2P16	Variable rw	Gelatchte 16 Bit-Position (positive Flanke)
3579 (hex)	LATCH2N16	Variable rw	Gelatchte 16 Bit-Position (negative Flanke)
357A (hex)	LATCH2P32	Variable rw	Gelatchte 32 Bit-Position (positive Flanke)
357B (hex)	LATCH2N32	Variable rw	Gelatchte 32 Bit-Position (negative Flanke)
357C (hex)	LATCH1P32	Variable rw	Gelatchte 32 Bit-Position (positive Flanke)
357D (hex)	LATCH1N32	Variable rw	Gelatchte 32 Bit-Position (negative Flanke)
357E (hex)	LED1	Variable rw	Anzeigestatus des LED1-Segmentes
357F (hex)	LED2	Variable rw	Anzeigestatus des LED2-Segmentes
3580 (hex)	LED3	Variable rw	Anzeigestatus des LED3-Segmentes
3581 (hex)	LEDSTAT	Variable rw	Seitennummer für das LED-Display
3582 (hex)	LIST	MLC	Auflistung aller ASCII-Kommandos
3583 (hex)	LOAD	Command	Laden der Parameter aus dem seriellen EEPROM

CAN Objekt Nummer	ASCII Kommando	ASCII Typ	Beschreibung
3584 (hex)	MAXTEMPE	Variable rw	Abschaltwert der Umgebungstemperatur
3585 (hex)	MAXTEMPH	Variable rw	Abschaltwert der Kühlkörpertemperatur
3586 (hex)	MAXTEMPM	Variable rw	Abschaltwert der Motortemperatur (Widerstand)
3587 (hex)	MBRAKE	Variable rw	Vorwahl für Motorbremse
3588 (hex)	MDBCNT	Variable ro	Anzahl der Motorsätze
3589 (hex)	MDBGET	Command	Info-Zeile für einen Motordatensatz
358A (hex)	MDBSET	Command	Vorwahl eines Motordatensatzes
358C (hex)	VLIM	Variable rw	Max. Drehzahl
358D (hex)	MH	Command	Start der Referenzfahrt
358E (hex)	MICONT	Variable rw	Motor Nennstrom
358F (hex)	MIPEAK	Variable rw	Motor Spitzenstrom
3591 (hex)	MJOG	Command	Start des Tippbetriebes
3592 (hex)	MVANGLP	Variable rw	Drehzahlabhängige Voreilung
3593 (hex)	MKT	Variable rw	Motor KT
3595 (hex)	MLGC	Variable rw	Relative Stromreglerverstärkung bei Dauerstrom
3596 (hex)	MLGD	Variable rw	Relative Stromreglerverstärkung des D-Stromreglers
3597 (hex)	MLGP	Variable rw	Relative Stromreglerverstärkung bei Spitzenstrom
3598 (hex)	MLGQ	Variable rw	Absolute Verstärkung des Stromreglers
3599 (hex)	MNUMBER	Variable rw	Laden eines Motor-Datensatzes
359C (hex)	MPHASE	Variable rw	Phasenlage des Feedback-Systems zum Motor
359D (hex)	MPOLES	Variable rw	Anzahl der Motorpole
35A0 (hex)	MRESBW	Variable rw	Resolver-Bandbreite
35A1 (hex)	MRESPOLES	Variable rw	Anzahl der Resolverpole
35A2 (hex)	MSG	Variable rw	RS232-Ausgabe der Warnungen/Fehlermeldungen
35A3 (hex)	MSPEED	Variable rw	Motor Maximaldrehzahl
35A5 (hex)	MTANGLP	Variable rw	Stromvoreilung
35A6 (hex)	MTYPE	Variable rw	Art des Motors
35A7 (hex)	MVANGLB	Variable rw	Drehzahlabhängige Voreilung (Einsatz Phi)
35A8 (hex)	MVANGLF	Variable rw	Drehzahlabhängige Voreilung (Endwert Phi)
35A9 (hex)	M_RESET	Command	Neuübersetzung der Macro-Programme
35AA (hex)	NONBTB	Variable rw	Netz-BTB-Überprüfung ein/aus
35AD (hex)	NREF	Variable rw	Referenzfahrtart
35AE (hex)	O1	Variable rw	Zustand des digitalen Ausgangs 1
35AE (hex)	O3_18	Variable rw	Zustand des digitalen Ausgangs 1
35AF (hex)	O1MODE	Variable rw	Funktion des digitalen Ausgangs 1
35AF (hex)	O3_18MODE	Variable rw	Funktion des digitalen Ausgangs 1
35B0 (hex)	O1TRIG	Variable rw	Hilfsvariable für O1MODE
35B0 (hex)	O3_18TRIG	Variable rw	Hilfsvariable für O1MODE
35B1 (hex)	O2	Variable rw	Zustand des digitalen Ausgangs 1
35B2 (hex)	O2MODE	Variable rw	Funktion des digitalen Ausgangs 2
35B3 (hex)	O2TRIG	Variable rw	Hilfsvariable für O2MODE
35B4 (hex)	OPMODE	Variable rw	Betriebsart des Verstärkers
35B5 (hex)	OPTION	Variable ro	Slotkarten-Kennung
35B6 (hex)	OVERRIDE	Variable rw	Override-Funktion
35B7 (hex)	O_ACC	Variable rw	Beschleunigungszeit für den Fahrsatz 0
35B9 (hex)	O_C	Variable rw	Steuervariable für den Fahrsatz 0
35BA (hex)	O_DEC	Variable rw	Bremsszeit für den Fahrsatz 0
35BC (hex)	O_FN	Variable rw	Folgefahrsatz-Nummer für den Fahrsatz 0
35BD (hex)	O_FT	Variable rw	Verzögerungszeit für den Folge-Fahrsatz
35BE (hex)	O_P	Variable rw	Zielposition/Verfahrstrecke für den Fahrsatz 0
35BF (hex)	O_V	Variable rw	Zielgeschwindigkeit für den Fahrsatz 0
35C0 (hex)	PBAL	Variable ro	Istwert der Ballastleistung
35C1 (hex)	PBALMAX	Variable rw	Maximale Ballastleistung
35C2 (hex)	PBALRES	Variable rw	Vorwahl des Ballastwiderstandes



CAN Objekt Nummer	ASCII Kommando	ASCII Typ	Beschreibung
35C3 (hex)	PBAUD	Variable ro	Profibus-Baudrate
35C5 (hex)	PE	Variable ro	Istwert des Schleppfehlers
35C6 (hex)	PEINPOS	Variable rw	In-Position-Fenster
35C7 (hex)	PEMAX	Variable rw	Max. Schleppfehler
35C8 (hex)	PFB	Variable ro	aktuelle Lagereglerposition
35C9 (hex)	PFB0	Variable ro	Lagereglerposition über den externen Encoder
35CA (hex)	PGEARI	Variable rw	Lageregler-Auflösung (Zähler)
35CB (hex)	PGEARO	Variable rw	Lageregler-Auflösung (Nenner)
35CC (hex)	PIOBUF	Variable rw	Profibus-Daten
35CD (hex)	PMODE	Variable rw	Netz-Phase Modus
35CE (hex)	PNOID	Variable ro	Profibus-Kennung (ID)
35CF (hex)	POSCNFG	Variable rw	Achsentyp
35D0 (hex)	PPOTYP	Variable rw	Profibus PPO-Typ
35D1 (hex)	PRBASE	Variable rw	interne Lage-Auflösung
35D2 (hex)	PRD	Variable ro	20 Bit Feedback-Position
35D3 (hex)	PROMPT	Variable rw	Vorwahl des RS232-Protokolls
35D4 (hex)	PSTATE	Variable ro	Profibus-Status
35D6 (hex)	PTMIN	Variable rw	Min. Beschleunigungsrampe
35D7 (hex)	PV	Variable ro	Ist-Geschwindigkeit (Lageregler)
35D8 (hex)	PVMAX	Variable rw	max. Geschwindigkeit für den Lageregler
35D9 (hex)	PVMAXN	Variable rw	max. Geschwindigkeit für den Lageregler (negativ)
35DB (hex)	PVMAXP	Variable rw	max. Geschwindigkeit für den Lageregler
35DD (hex)	READY	Variable ro	Zustand von Software-Enable
35DE (hex)	RECDONE	Variable ro	Scope: Aufnahme beendet
35DF (hex)	RECING	Variable ro	Scope: Aufzeichnung läuft
35E0 (hex)	RECOFF	Command	Scope: Abbruch einer Scope-Aufzeichnung
35E1 (hex)	RECRDY	Variable ro	Scope: Zustand der RECORD-Funktion
35E2 (hex)	REFIP	Variable rw	Spitzenstrom für die Referenzfahrt 7
35E4 (hex)	REMOTE	Variable ro	Zustand des Hardware-Enable
35E5 (hex)	RESPHASE	Variable rw	Resolverphase
35E6 (hex)	RK	Variable rw	Verstärkungsfaktor Resolver-Sinussignal
35E7 (hex)	ROFFS	Variable rw	Referenzoffset
35E8 (hex)	RS232T	Variable rw	Watch-Dog Zeit (RS232)
35E9 (hex)	RSTVAR	Command	Setzen aller Parameter auf Default-Werte
35EA (hex)	S	Command	Stop und Disable
35EB (hex)	SAVE	Command	Speichern der Daten im EEPROM
35EC (hex)	SBAUD	Variable rw	Sercos: Baudrate
35ED (hex)	SCAN	Command	Erkennung der CAN-Stationen
35EF (hex)	SERIALNO	Variable ro	Seriennummer des Verstärkers
35F0 (hex)	SETREF	Command	Setzen des Referenzpunktes
35F2 (hex)	SLEN	Variable rw	Optische Ausgangsleistung bei Sercos
35F3 (hex)	SLOTIO	Variable rw	I/O-Erweiterungskarte: IN/OUT-Zustand
35F4 (hex)	SPHAS	Variable rw	Sercos-Phase
35FA (hex)	SSTAT	Variable ro	
35FA (hex)	DUMPSLNO	Variable rw	Auflistung der numerischen EEPROM-Variablen
35FB (hex)	STAT	Variable ro	Verstärker-Statuswort
35FC (hex)	STATIO	Variable ro	Status der Ein/Ausgänge
35FD (hex)	STATUS	Variable ro	detaillierte Verstärker-Statusinformation
35FE (hex)	STOP	Command	Setzen des Sollwertes auf 0
35FF (hex)	STOPMODE	Variable rw	Bremsverhalten bei Disable
3600 (hex)	SWCNFG	Variable rw	Konfiguration der Software-Endschalter
3604 (hex)	SWE1	Variable rw	Softwareendschalter (kleinste Position)
3606 (hex)	SWE2	Variable rw	Softwareendschalter (größte Position)

CAN Objekt Nummer	ASCII Kommando	ASCII Typ	Beschreibung
360E (hex)	T	Command	digitaler Stromsollwert
360F (hex)	TASK	Variable ro	Task-Auslastung
3610 (hex)	TEMPE	Variable ro	Istwert der Umgebungstemperatur
3611 (hex)	TEMPH	Variable ro	Istwert der Kühlkörpertemperatur
3612 (hex)	TEMPM	Variable ro	Istwert der Motortemperatur
3613 (hex)	TRJSTAT	Variable ro	Status2-Information
3614 (hex)	TRUN	Variable ro	Betriebsstundenzähler
3617 (hex)	UVLTMODE	Variable rw	Unterspannungsmodus
3618 (hex)	V	Variable ro	Aktuelle Drehzahl
361A (hex)	VBUS	Variable ro	Zwischenkreisspannung
361B (hex)	VBUSBAL	Variable rw	Maximale Netzspannung
361C (hex)	VBUSMAX	Variable ro	Maximale Zwischenkreisspannung
361D (hex)	VBUSMIN	Variable rw	Minimale Zwischenkreisspannung
361E (hex)	VCMD	Variable ro	interner Drehzahlsollwert in UPM
3620 (hex)	VEL0	Variable rw	Stillstandsschwelle
3621 (hex)	VJOG	Variable rw	Tippbetrieb-Geschwindigkeit
3622 (hex)	VLIMP	Variable rw	Max. Drehzahl
3623 (hex)	VLIMN	Variable rw	Max. negative Drehzahl
3626 (hex)	VMUL	Variable rw	Geschwindigkeitsmultiplikator (Feldbus)
3627 (hex)	VOSPD	Variable rw	Überdrehzahl
3628 (hex)	VREF	Variable rw	Referenzfahrt-Geschwindigkeit
3629 (hex)	VSCALE1	Variable rw	SW1-Drehzahlskalierungsfaktor
362A (hex)	VSCALE2	Variable rw	SW2-Drehzahlskalierungsfaktor
362B (hex)	\	Command	Anwahl der Remote Adresse
362C (hex)	DILIM	Variable rw	DPR Strombegrenzung aktivieren
362D (hex)	DENA	Variable rw	DPR Software Disable Reset Modus
362F (hex)	KTN	Variable rw	Integralanteil des Stromreglers
3630 (hex)	INPT0	Variable rw	In-Position-Verzögerung
3632 (hex)	COLDSTART	Command	Hardware-Reset des Verstärkers
3636 (hex)	WPOS	Variable ro	Freigabe der schnellen Positionsregister
3637 (hex)	SRND	Variable rw	Startposition Modulo-Achse
3638 (hex)	ERND	Variable rw	Endposition Modulo-Achse
363A (hex)	BCC	Variable ro	EEPROM-Checksumme
363C (hex)	REFMODE	Variable rw	Quelle des Nullimpulse bei Referenzfahrt
363D (hex)	VLO	Variable rw	Software Resolver/Digital Wandler Vorsteuerung
363E (hex)	WMASK	Variable rw	Warnung/Fehler-Maske
363F (hex)	WPOSE	Variable ro	Freigabe der schnellen Positionsregister 1...16
3640 (hex)	WPOSP	Variable rw	Polarität der schnellen Positionsregister 1...16
3641 (hex)	WPOSX	Variable rw	Modus der schnellen Positionsregister 1...16
3642 (hex)	MOVE	Command	Starten eines Fahrsatzes
3643 (hex)	POSRSTAT	Variable rw	Status der schnellen Positionsregister 1...16
3644 (hex)	P1	Variable rw	schnelle Positionsschwelle
3645 (hex)	P2	Variable rw	schnelle Positionsschwelle
3646 (hex)	P3	Variable rw	schnelle Positionsschwelle
3647 (hex)	P4	Variable rw	schnelle Positionsschwelle
3648 (hex)	P5	Variable rw	schnelle Positionsschwelle
3649 (hex)	P6	Variable rw	schnelle Positionsschwelle
364A (hex)	P7	Variable rw	schnelle Positionsschwelle
364B (hex)	P8	Variable rw	schnelle Positionsschwelle
364C (hex)	P9	Variable rw	schnelle Positionsschwelle
364D (hex)	P10	Variable rw	schnelle Positionsschwelle
364E (hex)	P11	Variable rw	schnelle Positionsschwelle
364F (hex)	P12	Variable rw	schnelle Positionsschwelle
3650 (hex)	P13	Variable rw	schnelle Positionsschwelle

CAN Objekt Nummer	ASCII Kommando	ASCII Typ	Beschreibung
3651 (hex)	P14	Variable rw	schnelle Positionsschwelle
3652 (hex)	P15	Variable rw	schnelle Positionsschwelle
3653 (hex)	P16	Variable rw	schnelle Positionsschwelle
3654 (hex)	PTARGET	Variable rw	letzte Fahrsatzzielposition
3655 (hex)	ACTRS232	Variable rw	Freigabe des RS232-Watchdogs
3656 (hex)	ROFFSABS	Variable rw	Referenzoffset
3657 (hex)	FW	Variable ro	Liefert die Versionsnummer der Firmware
3658 (hex)	DPRILIMIT	Variable rw	Digitale Begrenzung des Spitzenstroms über DPR
3659 (hex)	ACCUNIT	Variable rw	Art der Beschleunigungsvorgabe im System
365A (hex)	VCOMM	Variable rw	Drehzahlschwelle für Kommutierungsüberwachung
365B (hex)	MTMUX	Variable rw	Voreinstellung für zu bearbeitenden Fahrsatz
365D (hex)	REFLS	Variable rw	
365F (hex)	VUNIT	Variable rw	Systemweite Definition der Drehzahl / Geschw.
3660 (hex)	PUNIT	Variable rw	Vorgabe der Positionsauflösung
366E (hex)	TBRAKE	Variable rw	Disableverzögerungszeit bei Bremsenbetrieb
366F (hex)	TBRAKE0	Variable rw	Bremsen Lüftzeit
3670 (hex)	CMDDLY	Variable rw	Kommandoverzögerungszeit der RS232
3671 (hex)	MSLBRAKE	Variable rw	Bremsrampe bei sensorlosem Nothalt
3672 (hex)	DRVCNFG	Variable rw	Konfigurationsvariable für CAN-Bus Kompatibilität
3673 (hex)	DISDPR	Variable rw	Disablen des DPR-Zugriffs für Schreibbefehle
3675 (hex)	ESPEED	Variable ro	Maximale Drehzahl in Abhängigkeit vom Gebertyp
367F (hex)	LATCH1P16	Variable rw	Gelatchte 16 Bit-Position (positive Flanke)
3680 (hex)	LATCH1N16	Variable rw	Gelachte 16 Bit-Position (negative Flanke)
3681 (hex)	EXTLATCH	Variable rw	Einstellung der Quellen für die Latcheingänge
3682 (hex)	STAGECODE	Variable ro	Endstufenkennung
3683 (hex)	SYNCSRC	Variable rw	
3686 (hex)	MRS	Variable rw	Wicklungswiderstand des Stators Phase-Phase
3691 (hex)	SERCSET	Variable rw	Setze Sercos Einstellungen
3695 (hex)	SMNUMBER	Variable ro	Gespeicherte Motornummer im Geber
3698 (hex)	VREF0	Variable rw	Reduzierfaktor Referenzfahrgeschwindigkeit
3699 (hex)	AN11NR	Variable rw	Nr. der INxTRIG Variable, bei analoger Vorgabe
369A (hex)	AN11RANGE	Variable rw	Bereich für die analoge Änderung von INxTRIG
36A3 (hex)	MSERIALNO	Variable rw	Motorseriennummer bei Encoder mit Parameterkanal
36A5 (hex)	VSTFR	Variable rw	Drehzahleckpunkt bei Reibungskompensation
36B6 (hex)	DOVRIDE	Variable rw	Vorgabe eines digitalen Override-Faktors
36BE (hex)	INS0	Variable ro	Status von Eingang A0 der I/O Erweiterungskarte
36BF (hex)	INS1	Variable ro	Status von Eingang A1 der I/O Erweiterungskarte
36C0 (hex)	INS2	Variable ro	Status von Eingang A2 der I/O Erweiterungskarte
36C1 (hex)	INS3	Variable ro	Status von Eingang A3 der I/O Erweiterungskarte
36C2 (hex)	INS4	Variable ro	Status von Eingang A4 der I/O Erweiterungskarte
36C3 (hex)	INS5	Variable ro	Status von Eingang A5 der I/O Erweiterungskarte
36C4 (hex)	INS6	Variable ro	Status von Eingang A6 der I/O Erweiterungskarte
36C5 (hex)	INS7	Variable ro	Status von Eingang A7 der I/O Erweiterungskarte
36C6 (hex)	INS8	Variable ro	Status von FSTART_IO der I/O Erweiterungskarte
36C7 (hex)	OS1	Variable rw	Setzen von "Posreg1" der I/O Erweiterungskarte
36C8 (hex)	OS2	Variable rw	Setzen von "Posreg2" der I/O Erweiterungskarte
36C9 (hex)	OS3	Variable rw	Setzen von "Posreg3" der I/O Erweiterungskarte
36CA (hex)	OS4	Variable rw	Setzen von "Posreg4" der I/O Erweiterungskarte
36CB (hex)	OS5	Variable rw	Setzen von "Posreg5" der I/O Erweiterungskarte
36CE (hex)	LASTWMASK	Variable ro	Fehlerspeicher von WMASK
36D0 (hex)	WSTIME	Variable rw	Ausführungszeit der W&S-Funktion
36D1 (hex)	WSAMPL	Variable rw	Minimale Bewegung der W&S Funktion
36D2 (hex)	NREFMT	Variable rw	Referenzfahrt mit automatischem Folgefahrauftrag

CAN Objekt Nummer	ASCII Kommando	ASCII Typ	Beschreibung
36D7 (hex)	AUTOHOME	Variable rw	
36D8 (hex)	PASSCNFG	Variable rw	Passwort Funktion
36E4 (hex)	DRVCNFG2	Variable rw	Regler Zusatzfunktionen
36E5 (hex)	BUSP1	Variable rw	Zustand des Modbus+ Netzwerks
36E6 (hex)	BUSP2	Variable rw	Anzahl der Datenworte (Sollwert) bei MODBUS+
36E7 (hex)	BUSP3	Variable rw	Vorgaberichtung der Adresse bei Modbus+
36E8 (hex)	BUSP4	Variable rw	Anzahl der Datenworte (Sollwert) bei MODBUS+
36E9 (hex)	BUSP5	Variable rw	Anzahl der Datenworte (Sollwert) bei MODBUS+
36EA (hex)	BUSP6	Variable rw	Anzahl der Istwert-Datenworte über Modbus
3890 (hex)	CSSTAT	Variable ro	Status (siehe SDO 2401h (→ # 103))
3891 (hex)	CSIOSTAT	Variable ro	Status (siehe SDO 2402h (→ # 103))
3892 (hex)	CSERR	Variable ro	Status (siehe SDO 2403h (→ # 103))
3893 (hex)	FBGEARI	Variable rw	Kompensationsfaktor Motor/Getriebe
3894 (hex)	FBGEARO	Variable rw	Kompensationsfaktor Motor/Getriebe
3895 (hex)	BCCOT	Variable ro	EEPROM Checksumme ohne Betriebsstundenzähler
3896 (hex)	HLIMIT1	Variable rw	Software Endschalter 1 für Referenzfahrt
3897 (hex)	HLIMIT2	Variable rw	Software Endschalter 2 für Referenzfahrt
3898 (hex)	HLIMIT3	Variable rw	Software Endschalter 3 für Referenzfahrt
3899 (hex)	PASSPLC	Variable rw	Passwort für SPS Kommandos
389A (hex)	PASSXPLC	Variable rw	Passwort Konfiguration für SPS Kommandos
389B (hex)	PTARGOFFS	Variable rw	Fahraufträge Zielposition Offset
389C (hex)	RSVERROFF	Variable rw	Resolver Fehlerüberwachung Aus
389D (hex)	DIRIN	Variable rw	Polarität der digitalen Eingänge
389E (hex)	RELTIME	Variable rw	Thyristor Spannungsregelung einschalten
38A0 (hex)	ANIN3	Variable ro	Spannung am Analogeingang 3 (POS I/O)
38A1 (hex)	ANIN4	Variable ro	Spannung am Analogeingang 4 (POS I/O)
38A2 (hex)	ANOFF3	Variable rw	Analog Offset für Analogeingang 3 (POS I/O)
38A3 (hex)	ANOFF4	Variable rw	Analog Offset für Analogeingang 4 (POS I/O)
38A4 (hex)	ANZERO3	Command	Analogeingang 3 nullen (POS I/O)
38A5 (hex)	ANZERO4	Command	Analogeingang 4 nullen (POS I/O)
38A6 (hex)	CALCPOSIO	Command	Initialisierung der POS I/O Karte
38A7 (hex)	GPTNTH1	Variable rw	Integralanteil der Schwelle des Geschwindigkeit Sollwerts
38A7 (hex)	GPTNTH2	Variable rw	Integralanteil der Schwelle des Geschwindigkeit Istwert
36CC (hex)	GPTNTH3	Variable rw	Integralanteil der Schwelle des Schleppfehlers
38AF (hex)	GPDELAY	Variable rw	Verzögerung für externen Position Sollwert
38B1 (hex)	GPTN	Variable rw	Verstärkung Integralanteil des PI Lagereglers
38B2 (hex)	GPTN_X	Variable rw	Verstärkung Integralanteil des PI Lagereglers
38B3 (hex)	MAXISETP	Variable rw	Begrenzung des Integralanteils des Lagereglers
38B4 (hex)	MAXVSETP	Variable rw	Begrenzung des Proportionalanteils des Lagereglers
38B7 (hex)	BISSCNFG	Variable rw	Konfigurationsparameter für BiSS-C
38B9 (hex)	S1DLY	Variable rw	Einschaltverzögerung
38BB (hex)	TEMPMFILT	Variable rw	Filterzeitkonstante für Filter der Motortemperatur Sensoren
38BE (hex)	EWH	Variable rw	Verbrauchte Elektrische Energie in Watt-Stunden
38BF (hex)	VBUSBOFFS	Variable rw	Absenkung der Ballasteinschaltsschwelle
38C0 (hex)	VLO_X	Variable rw	Vorsteuerung des Software Resolver/Digital Konverters
38C1 (hex)	VBUSRAMP	Variable rw	Einschaltsschwelle für Bremsrampensteuerung
38C2 (hex)	VBUSFACT	Variable rw	Verstärkungsfaktor für Steuerung der Verzögerungsrampe
38C3 (hex)	GP2	Variable rw	Proportionalverstärkung für Geschwindigkeit größer GPS2
38C4 (hex)	GPS1	Variable rw	Untere Geschwindigkeitsschwelle (Positionsregler)
38C5 (hex)	GPS2	Variable rw	Obere Geschwindigkeitsschwelle (Positionsregler)
38C6 (hex)	GV2	Variable rw	Proportionalverstärkung für Geschwindigkeit größer GVS2
38C7 (hex)	GVS1	Variable rw	Untere Geschwindigkeitsschwelle (Geschwindigkeitsregler)
38C8 (hex)	GVS2	Variable rw	Obere Geschwindigkeitsschwelle (Geschwindigkeitsregler)

CAN Objekt Nummer	ASCII Kommando	ASCII Typ	Beschreibung
38CA (hex)	CSCNFG	Variable rw	Konfigurationsvariable für Safety Karte
38CB (hex)	CSENID	Variable rw	Blindstromeinprägung
38CC (hex)	CSIDMAX	Variable rw	Maximalwert für Blindstromsollwert
38CE (hex)	DRVCNFG4	Variable rw	Konfigurationsparameter für zusätzliche Verstärker Funktionalität
38CF (hex)	MVANGLMODE	Variable rw	S600 kompatibler Modus für geschwindigkeitsabhängige Phasenvoreilung
38D0 (hex)	VBUSOFFS	Variable rw	Korrekturwert für Schaltschwelle Bremsschaltung S748/772
38D1 (hex)	PMECH	Variable ro	Geschätzte vom Motor erzeugte mechanische Leistung
38D2 (hex)	P	Variable ro	Berechnete aktive Leistung (elektrisch)
38D3 (hex)	ENCFIX	Variable rw	Dezimalstellen für ENCLINES
38D5 (hex)	LINRESOL	Variable rw	Umrechnungsfaktor für lineare Messsysteme
38D6 (hex)	ENCDIV	Variable rw	Dezimalstellen Teiler von ENCLINES
38D7 (hex)	QCNFG	Variable rw	Konfiguration der Servopumpe
38D8 (hex)	QFACTOR	Variable rw	Schluckvolumen der Servopumpe
38D9 (hex)	QFCFACT	Variable ro	Schluckvolumen der Servopumpe
38DA (hex)	QMACT	Variable ro	Aktueller Momentenwert für Servopumpe
38DB (hex)	QTVEFF	Variable ro	Aktueller D-Anteil für Servopumpen Regelung
38DD (hex)	QFSET	Variable ro	Durchflusssollwert für Servopumpe (gefiltert)
38DE (hex)	QFACT	Variable ro	Durchflussswert für Servopumpe
38DF (hex)	QPRSET	Variable rw	Drucksollwert für Servopumpe
38E0 (hex)	QPRACT	Variable ro	Druckistwert für Servopumpe
38E1 (hex)	QMCMD	Variable ro	Aktueller Momentensollwert für Servopumpe
38E2 (hex)	QFACTCOMP	Variable ro	Aktueller Durchflussswert für Servopumpe (kompensiert)
38E3 (hex)	QFACTFILT	Variable ro	Aktueller Durchflussswert für Servopumpe (gefiltert)
38E4 (hex)	QTIMEKD	Variable rw	Zeitkonstante des D-Anteils für Servopumpen Regelung
38E5 (hex)	QVT1	Variable rw	Zeitkonstante des Durchflussfilters für Servopumpe
38E6 (hex)	QMCMDLIM	Variable ro	Aktueller Momentensollwert für Servopumpe (nach Begrenzung)
38E7 (hex)	QMLIMP	Variable rw	Momentbegrenzung für Servopumpe (Druckaufbau)
38E9 (hex)	QFMAX	Variable rw	Max. Durchflusssollwert für Servopumpen Regelung
38EA (hex)	QPRMAX	Variable rw	Max. Drucksollwert für Servopumpen Regelung
38EB (hex)	QTAB1KP	Variable rw	Proportionalverstärkung 1 für Servopumpen Regelung
38EC (hex)	QTAB2KP	Variable rw	Proportionalverstärkung 2 für Servopumpen Regelung
38ED (hex)	QTAB3KP	Variable rw	Proportionalverstärkung 3 für Servopumpen Regelung
38EE (hex)	QTAB4KP	Variable rw	Proportionalverstärkung 4 für Servopumpen Regelung
38EF (hex)	QTAB5KP	Variable rw	Proportionalverstärkung 5 für Servopumpen Regelung
38F0 (hex)	QTAB1V	Variable rw	Geschwindigkeitsschwellwert 1 für Servopumpen Regelung
38F1 (hex)	QTAB2V	Variable rw	Geschwindigkeitsschwellwert 2 für Servopumpen Regelung
38F2 (hex)	QTAB3V	Variable rw	Geschwindigkeitsschwelle 3 für Servopumpen Regelung
38F3 (hex)	QTAB4V	Variable rw	Geschwindigkeitsschwelle 4 für Servopumpen Regelung
38F4 (hex)	QTAB5V	Variable rw	Geschwindigkeitsschwelle 5 für Servopumpen Regelung
38F5 (hex)	QTAB1TN	Variable rw	Verstärkung Integralanteil 1 für Servopumpen Regelung
38F6 (hex)	QTAB2TN	Variable rw	Verstärkung Integralanteil 2 für Servopumpen Regelung
38F7 (hex)	QTAB3TN	Variable rw	Verstärkung Integralanteil 3 für Servopumpen Regelung
38F8 (hex)	QTAB4TN	Variable rw	Verstärkung Integralanteil 4 für Servopumpen Regelung
38F9 (hex)	QTAB5TN	Variable rw	Verstärkung Integralanteil 5 für Servopumpen Regelung
38FA (hex)	QPE	Variable ro	Druck Schleppfehler für Servopumpen Regelung
38FB (hex)	QKPEFF	Variable ro	Aktuelle Proportionalverstärkung für Servopumpen Regelung
38FC (hex)	QTNEFF	Variable ro	Aktuelle Integralverstärkung für Servopumpen Regelung
38FD (hex)	GVTV	Variable rw	Vorhaltezeit für Geschwindigkeitsregler
38FE (hex)	GVTV_X	Variable rw	Vorhaltezeit für Geschwindigkeitsregler (2. Parametersatz)
3900 (hex)	QTABID	Variable ro	Aktuelle Verstärkungen für Servopumpen Regelung
3901 (hex)	QTAB1TV	Variable rw	Vorhaltezeit 1 für Servopumpen Regelung
3902 (hex)	QTAB2TV	Variable rw	Vorhaltezeit 2 für Servopumpen Regelung

CAN Objekt Nummer	ASCII Kommando	ASCII Typ	Beschreibung
3903 (hex)	QTAB3TV	Variable rw	Vorhaltezeit 3 für Servopumpen Regelung
3904 (hex)	QTAB4TV	Variable rw	Vorhaltezeit 4 für Servopumpen Regelung
3905 (hex)	QTAB5TV	Variable rw	Vorhaltezeit 5 für Servopumpen Regelung
3907 (hex)	GFTN	Variable rw	Zeitkonstante Flusskontrolle (Asynchronmotor)
3908 (hex)	MCFW	Variable rw	Korrekturfaktor für Feldschwächung
3909 (hex)	ENCDIVD	Variable rw	Teiler für Dezimalstellen von ENCIN
390A (hex)	ENCFIXD	Variable rw	Dezimalstellen für ENCIN Parameter
390B (hex)	SSITIME	Variable rw	SSI Erholungszeit
390C (hex)	SSIMASK	Variable rw	Filtermaske für empfangene SSI Daten
390D (hex)	EN22A1CNT	Variable ro	ENDAT2.2 Status zusätzliche Informationen 1
390E (hex)	EN22A2CNT	Variable ro	ENDAT2.2 Status zusätzliche Informationen 2
390F (hex)	EN22CNFG	Variable rw	ENDAT2.2 Konfigurationsparameter für Diagnose Information
3910 (hex)	EN22DCNT	Variable ro	ENDAT2.2 Status Diagnose Informationen
3911 (hex)	GV2_X	Variable rw	Proportionalverstärkung für Geschwindigkeit größer GVS2
3912 (hex)	ZEROTIME	Variable rw	Default value for the ZERO function
3913 (hex)	QENA	Variable rw	Enable/Disable Funktion Servopumpe
3914 (hex)	QFCMD	Variable rw	Durchfluss Sollwert für Servopumpe (ungefiltert)
3916 (hex)	QT1	Variable rw	Durchfluss Filterzeit für Servopumpe
3917 (hex)	QMLIMN	Variable rw	Momentbegrenzung für Servopumpe (Druckminderung)
3918 (hex)	HDSLTEMP	Variable ro	Motor/Encoder Temperatur gelesen vom HIPERFACE DSL
3919 (hex)	HDSLLED	Variable ro	LED Strom des HIPERFACE DSL
391A (hex)	HDSL VOLT	Variable ro	Spannungsversorgung des HIPERFACE DSL
391B (hex)	HDSLVEL	Variable ro	Geschwindigkeits Istwert des HIPERFACE DSL
391C (hex)	HDSLTIME	Variable ro	Betriebstundenzähler des HIPERFACE DSL
391D (hex)	HDSLROTAT	Variable ro	Umdrehungszähler des HIPERFACE DSL

## 6.1.3 Objektverzeichnis

Index	Sub-Index	Datentyp	Zugriff	PDO mapp.	Beschreibung	ASCII Objekt
1000h	0	UNSIGNED32	ro	—	Gerätetyp	—
1001h	0	UNSIGNED8	ro	—	Fehlerregister	—
1002h	0	UNSIGNED32	ro	ja	Herstellerspezifisches Statusregister	—
1003h		ARRAY			Vordefiniertes Fehlerfeld	—
	0	UNSIGNED8	rw	—	Anzahl der Fehler	—
	1...8	UNSIGNED32	ro	—	Standard Fehlerfeld	—
1005h	0	UNSIGNED32	rw	—	COB-ID SYNC message	—
1006h	0	UNSIGNED32	rw	nein	Communication cycle period	—
1008h	0	Visible String	const	—	Hersteller Gerätename	—
1009h	0	Visible String	const	—	Hersteller Hardware Version	—
100Ah	0	Visible String	const	—	Hersteller Software Version	—
100Ch	0	UNSIGNED16	rw	—	Überwachungszeit	—
100Dh	0	UNSIGNED8	rw	—	Lifetime Faktor	—
1010h	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	—	Speichern aller Parameter	SAVE
1011h	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	—	Auf Defaultwerte zuruecksetzen	RSTVAR
1014h	0	UNSIGNED32	rw	—	COB-ID für das Emergency Object	—
1016h		RECORD			Consumer heartbeat time	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	—	Consumer heartbeat time	—
1017h	0	UNSIGNED16	rw	—	Producer heartbeat time	—
1018h		RECORD			Identitäts Objekt	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	ro	—	Hersteller ID	—
	2	UNSIGNED32	ro	—	Produkt Code	—
	3	UNSIGNED32	ro	—	Revisions Nummer	—
	4	UNSIGNED32	ro	—	Seriennummer	SERIALNO
1026h		ARRAY			OS Prompt	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED8	w	—	StdIn	—
	2	UNSIGNED8	ro	—	StdOut	—
1400h		RECORD			RXPDO1 Kommunikations Parameter	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	—	RXPDO1 COB-ID	—
	2	UNSIGNED8	rw	—	Übertragungstyp RXPDO1	—
1401h		RECORD			RXPDO2 Kommunikations Parameter	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	—	RXPDO2 COB-ID	—
	2	UNSIGNED8	rw	—	Übertragungstyp RXPDO2	—
1402h		RECORD			RXPDO3 Kommunikations Parameter	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	—	RXPDO3 COB-ID	—
	2	UNSIGNED8	rw	—	Übertragungstyp RXPDO3	—
1403h		RECORD			RXPDO4 Kommunikations Parameter	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	—	RXPDO4 COB-ID	—
	2	UNSIGNED8	rw	—	Übertragungstyp RXPDO4	—
1600h		RECORD			RXPDO1 Mapping Parameter	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1...8	UNSIGNED32	rw	—	Mapping für n-tes Applikation Objekt	—
1601h		RECORD			RXPDO2 Mapping Parameter	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1...8	UNSIGNED32	rw	—	Mapping für n-tes Applikation Objekt	—

Index	Sub-Index	Datentyp	Zugriff	PDO mapp.	Beschreibung	ASCII Objekt
1602h		RECORD			RXPDO3 Mapping Parameter	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1...8	UNSIGNED32	rw	—	Mapping für n-tes Applikation Objekt	—
1603h		RECORD			RXPDO4 Mapping Parameter	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1...8	UNSIGNED32	rw	—	Mapping für n-tes Applikation Objekt	—
1800h		RECORD			TXPDO1 Kommunikations Parameter	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	—	TXPDO1 COB-ID	—
	2	UNSIGNED8	rw	—	Übertragungstyp TXPDO1	—
	3	UNSIGNED16	rw	—	Verzögerungszeit	—
	4	UNSIGNED8	const	—	reserviert	—
1801h		RECORD			TXPDO2 Kommunikations Parameter	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	—	TXPDO2 COB-ID	—
	2	UNSIGNED8	rw	—	Übertragungstyp TXPDO2	—
	3	UNSIGNED16	rw	—	Verzögerungszeit	—
	4	UNSIGNED8	const	—	reserviert	—
1802h		RECORD			TXPDO3 Kommunikations Parameter	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	—	TXPDO3 COB-ID	—
	2	UNSIGNED8	rw	—	Übertragungstyp TXPDO3	—
	3	UNSIGNED16	rw	—	Verzögerungszeit	—
	4	UNSIGNED8	const	—	reserviert	—
1803h		RECORD			TXPDO4 Kommunikations Parameter	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	—	TXPDO4 COB-ID	—
	2	UNSIGNED8	rw	—	Übertragungstyp TXPDO4	—
	3	UNSIGNED16	rw	—	Verzögerungszeit	—
	4	UNSIGNED8	const	—	reserviert	—
1A00h		RECORD			Mapping Parameter TXPDO1	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1...8	UNSIGNED32	rw	—	Mapping für n-tes Applikation Objekt	—
1A01h		RECORD			Mapping Parameter TXPDO2	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1...8	UNSIGNED32	rw	—	Mapping für n-tes Applikation Objekt	—
1A02h		RECORD			Mapping Parameter TXPDO3	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1...8	UNSIGNED32	rw	—	Mapping für n-tes Applikation Objekt	—
1A03h		RECORD			Mapping Parameter TXPDO4	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1...8	UNSIGNED32	rw	—	Mapping für n-tes Applikation Objekt	—
2000h	0	UNSIGNED32	ro	ja	Herstellerspezifische Warnungen	STATCODE
2014h	0	ARRAY	ro	—	Maske TxPDO Kanal 1	—
	1	UNSIGNED32	rw	—	Maske (Byte 0..3)	—
	2	UNSIGNED32	rw	—	Maske (Byte 4..7)	—
2015h	0	ARRAY	ro	—	Maske TxPDO Kanal 2	—
	1	UNSIGNED32	rw	—	Maske (Byte 0..3)	—
	2	UNSIGNED32	rw	—	Maske (Byte 4..7)	—



Index	Sub-Index	Datentyp	Zugriff	PDO mapp.	Beschreibung	ASCII Objekt
2016h	0	ARRAY	ro	—	Maske TxPDO Kanal 3	—
	1	UNSIGNED32	rw	—	Maske (Byte 0..3)	—
	2	UNSIGNED32	rw	—	Maske (Byte 4..7)	—
2017h	0	ARRAY	ro	—	Maske TxPDO Kanal 4	—
	1	UNSIGNED32	rw	—	Maske (Byte 0..3)	—
	2	UNSIGNED32	rw	—	Maske (Byte 4..7)	—
2030h		ARRAY	wo		DP-Ram Variablen (PDO)	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	INTEGER32	rw	ja	DP-Ram Variable 9	DPRVAR9
	2	INTEGER32	rw	ja	DP-Ram Variable 10	DPRVAR10
	3	INTEGER32	rw	ja	DP-Ram Variable 11	DPRVAR11
	4	INTEGER32	rw	ja	DP-Ram Variable 12	DPRVAR12
	5	INTEGER32	rw	ja	DP-Ram Variable 13	DPRVAR13
	6	INTEGER32	rw	ja	DP-Ram Variable 14	DPRVAR14
	7	INTEGER32	rw	ja	DP-Ram Variable 15	DPRVAR15
8	INTEGER32	rw	ja	DP-Ram Variable 16	DPRVAR16	
2040h		RECORD			Übersetzungsverhältnis elektr. Getriebe	—
	0	UNSIGNED8		—	Anzahl der Einträge	—
	1	INTEGER32	rw	ja	Eingangsfaktor für elektron. Getriebe	GEARI
	2	UNSIGNED32	rw	ja	Ausgangsfaktor für elektron. Getriebe	GEARO
2041h		RECORD			Istwerte elektronisches Getriebe	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	INTEGER32	ro	ja	Slave-Geschwindigkeitssollwert	—
	2	INTEGER32	ro	ja	Mastergeschwindigkeit	—
	3	UNSIGNED8	ro	ja	Internes Kontrollwort für elektrisches Getriebe	—
2051h		ARRAY			Konfiguration der Positionsregister	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	ja	Freigabe der Positionsregister	WPOSE
	2	UNSIGNED32	rw	ja	Modus der Positionsregister	WPOSX
	3	UNSIGNED32	rw	ja	Polarität der Positionsregister	WPOSP
2052h		ARRAY			Positionsregister, absolut	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	INTEGER32	rw	ja	Positionsregister 1, absolut	P1
	2	INTEGER32	rw	ja	Positionsregister 2, absolut	P2
	3	INTEGER32	rw	ja	Positionsregister 3, absolut	P3
	4	INTEGER32	rw	ja	Positionsregister 4, absolut	P4
	5	INTEGER32	rw	ja	Positionsregister 5, absolut	P5
	6	INTEGER32	rw	ja	Positionsregister 6, absolut	P6
	7	INTEGER32	rw	ja	Positionsregister 7, absolut	P7
	8	INTEGER32	rw	ja	Positionsregister 8, absolut	P8
	9	INTEGER32	rw	ja	Positionsregister 9, absolut	P9
	10	INTEGER32	rw	ja	Positionsregister 10, absolut	P10
	11	INTEGER32	rw	ja	Positionsregister 11, absolut	P11
	12	INTEGER32	rw	ja	Positionsregister 12, absolut	P12
	13	INTEGER32	rw	ja	Positionsregister 13, absolut	P13
	14	INTEGER32	rw	ja	Positionsregister 14, absolut	P14
	15	INTEGER32	rw	ja	Positionsregister 15, absolut	P15
16	INTEGER32	rw	ja	Positionsregister 16, absolut	P16	

Index	Sub-Index	Datentyp	Zugriff	PDO mapp.	Beschreibung	ASCII Objekt
2053h		ARRAY			Positionsregister, relativ	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	INTEGER32	rw	ja	Positionsregister 1, relativ	P1
	2	INTEGER32	rw	ja	Positionsregister 2, relativ	P2
	3	INTEGER32	rw	ja	Positionsregister 3, relativ	P3
	4	INTEGER32	rw	ja	Positionsregister 4, relativ	P4
	5	INTEGER32	rw	ja	Positionsregister 5, relativ	P5
	6	INTEGER32	rw	ja	Positionsregister 6, relativ	P6
	7	INTEGER32	rw	ja	Positionsregister 7, relativ	P7
	8	INTEGER32	rw	ja	Positionsregister 8, relativ	P8
	9	INTEGER32	rw	ja	Positionsregister 9, relativ	P9
	10	INTEGER32	rw	ja	Positionsregister 10, relativ	P10
	11	INTEGER32	rw	ja	Positionsregister 11, relativ	P11
	12	INTEGER32	rw	ja	Positionsregister 12, relativ	P12
	13	INTEGER32	rw	ja	Positionsregister 13, relativ	P13
	14	INTEGER32	rw	ja	Positionsregister 14, relativ	P14
	15	INTEGER32	rw	ja	Positionsregister 15, relativ	P15
16	INTEGER32	rw	ja	Positionsregister 16, relativ	P16	
2061h	0	UNSIGNED16	rw	ja	Strombegrenzung im Drehzahlmode	DPRLIMIT
2080h	0	UNSIGNED16	rw	ja	Fahrauftrag für Profile Position Mode	—
2081h	0	UNSIGNED16	rw	ja	Anzeige des aktiven Fahrauftrages	MOVE
2082h	0	UNSIGNED32	wo	—	Kopieren von Fahrsätzen	OCOPY
2083h	0	UNSIGNED32	wo	—	Löschen der Flash-Fahrsätze	—
2090h		ARRAY			DP-Ram Variablen, nur lesbar	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	INTEGER32	ro	ja	DP-Ram Variable 1	DPRVAR1
	2	INTEGER32	ro	ja	DP-Ram Variable 2	DPRVAR2
	3	INTEGER32	ro	ja	DP-Ram Variable 3	DPRVAR3
	4	INTEGER32	ro	ja	DP-Ram Variable 4	DPRVAR4
	5	INTEGER32	ro	ja	DP-Ram Variable 5	DPRVAR5
	6	INTEGER32	ro	ja	DP-Ram Variable 6	DPRVAR6
	7	INTEGER32	ro	ja	DP-Ram Variable 7	DPRVAR7
8	INTEGER32	ro	ja	DP-Ram Variable 8	DPRVAR8	
20A0h	0	INTEGER32	ro	ja	Latchposition 1, positive Flanke	LATCH1P32
20A1h	0	INTEGER32	ro	ja	Latchposition 1, negative Flanke	LATCH1N32
20A2h	0	INTEGER32	ro	ja	Latchposition 2, positive Flanke	LATCH2P32
20A3h	0	INTEGER32	ro	ja	Latchposition 2, negative Flanke	LATCH2N32
20A4h	0	UNSIGNED8	rw	ja	Latch—Kontrollregister	—
20B0h	0	INTEGER32	rw	ja	Triggervariable Digitaleingang 20	IN20TRIG
20B1h	0	UNSIGNED32	rw	ja	Steuerwort Digitaleingänge 5 .. 20	IN5 .. IN20
20B2h		ARRAY			Analoge Eingänge	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	INTEGER16	ro	ja	Spannung Analogeingang 1	ANIN1
	2	INTEGER16	ro	ja	Spannung Analogeingang 2	ANIN2
2100h	0	UNSIGNED32	rw	ja	Schreib-Dummy	—
2101h	0	UNSIGNED32	ro	ja	Lese-Dummy	—
2400h		RECORD			Seriennummer der Sicherheitskarte	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1—6	UNSIGNED32	ro	—	Seriennummer Sicherheitskarte Teil 1-6	—
2401h	0	UNSIGNED32	ro	ja	Status der Sicherheitskarte	CSSTAT
2402h	0	UNSIGNED32	ro	ja	I/O Status der Sicherheitskarte	CSIOSTAT
2403h	0	UNSIGNED32	ro	ja	Errorregister der Sicherheitskarte	CSERR

Index	Sub-Index	Datentyp	Zugriff	PDO mapp.	Beschreibung	ASCII Objekt
2404h		RECORD			Fehlerstack – Error Number	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1–128	UNSIGNED32	ro	—	Fehlerstack – Error Number 1...128	—
2405h		RECORD			Fehlerstack – Error Time	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1–128	UNSIGNED32	ro	—	Fehlerstack – Error Time 1...128	—
2406h		RECORD			Fehlerstack – Error Index	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1–128	UNSIGNED32	ro	—	Fehlerstack – Error Index 1...128	—
2407h		RECORD			Fehlerstack – Error Info	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1–128	UNSIGNED32	ro	—	Fehlerstack – Error Info 1...128	—
2408h		RECORD			Fehlerstack – Error Parameter 1	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1–128	UNSIGNED32	ro	—	Fehlerstack – Parameter 1 Error 1...128	—
2409h		RECORD			Fehlerstack – Error Parameter 2	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1–128	UNSIGNED32	ro	—	Fehlerstack – Parameter 2 Error 1...128	—
240Ah		RECORD			Fehlerstack – Error Parameter 3	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1–128	UNSIGNED32	ro	—	Fehlerstack – Parameter 3 Error 1...128	—
240Bh		RECORD			Fehlerstack – Error Parameter 4	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1–128	UNSIGNED32	ro	—	Fehlerstack – Parameter 4 Error 1...128	—
240Ch	0	UNSIGNED32	ro	ja	Sicherheitskarte: Aktuelle Geschwindigkeit	—
3500h		RECORD			ASCII Kommando MAXCMD	MAXCMD
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	ro	—	Wert	—
	2	UNSIGNED32	ro	—	Unterer Grenzwert	—
	3	UNSIGNED32	ro	—	Oberer Grenzwert	—
	4	UNSIGNED32	ro	—	Defaultwert	—
	5	UNSIGNED32	ro	—	Parameter Format	—
	6	UNSIGNED32	ro	—	Parameter Steuerdaten	—
	7	UNSIGNED32	ro	—	reserviert	—
8	UNSIGNED32	ro	—	reserviert	—	
3500h+MAXCMD		RECORD			Letzter Eintrag des ASCII Objektkanals	—
6040h	0	UNSIGNED16	w	ja	Control Word	—
6041h	0	UNSIGNED16	ro	ja	Status Word	—
6060h	0	INTEGER8	rw	ja	Betriebsarten	—
6061h	0	INTEGER8	ro	ja	Anzeige Betriebsarten	—
6063h	0	INTEGER32	ro	ja	Position-Istwert (Inkremente)	—
6064h	0	INTEGER32	ro	ja	Position-Istwert (Positions Einheiten)	PFB
6065h	0	UNSIGNED32	rw	—	Schleppfehlerfenster	PEMAX
6067h	0	UNSIGNED32	rw	—	"In Position" – Fenster	PEINPOS
6068h	0	UNSIGNED16	rw	—	"In Position" – Zeitfenster	INPT1
606Ch	0	INTEGER32	ro	ja	Geschwindigkeits—Istwert	—
6071h	0	INTEGER16	rw	ja	Drehmoment—Sollwert	—
6073h	0	UNSIGNED16	rw	—	Normierter Maximalstrom	MIPEAK
6077h	0	INTEGER16	ro	ja	Drehmoment—Istwert	—
607Ah	0	INTEGER32	rw	ja	Positions—Sollwert	O_P
607Ch	0	INTEGER32	rw	—	Referenz Offset	ROFFS

Index	Sub-Index	Datentyp	Zugriff	PDO mapp.	Beschreibung	ASCII Objekt
607Dh		ARRAY			Softwareendschalter	
	0	UNSIGNED8	ro	—	Anzahl der Einträge	
	1	INTEGER32	rw	—	Softwareendschalter 1	SWE1
	2	INTEGER32	rw	—	Softwareendschalter 2	SWE2
607Fh	0	UNSIGNED32	rw	—	Maximale Geschwindigkeit im PP-Mode	PVMAX
6080h	0	UNSIGNED32	rw	—	Maximale Motordrehzahl	VLIM
6081h	0	UNSIGNED32	rw	ja	Profil Geschwindigkeit	O_V
6083h	0	UNSIGNED32	rw	ja	Profil Beschleunigung	O_ACC (pp)/ ACC (pv)
6084h	0	UNSIGNED32	rw	ja	Profil Bremsen	O_DEC (pp)/ DEC (pv)
6085h	0	UNSIGNED32	rw	—	Schnellhaltrampe	DECSTOP
6086h	0	INTEGER16	rw	ja	Rampenprofil (Trapez / sin <sup>2</sup> )	O_C
6089h	0	INTEGER8	rw	—	Position Notation Index	—
608Ah	0	UNSIGNED8	rw	—	Position Dimension Index	—
608Bh	0	INTEGER8	rw	—	Geschwindigkeit Notation Index	—
608Ch	0	UNSIGNED8	rw	—	Geschwindigkeit Dimension Index	—
608Dh	0	INTEGER8	rw	—	Beschleunigung Notation Index	—
608Eh	0	UNSIGNED8	rw	—	Beschleunigung Dimension Index	—
608Fh		ARRAY			Positions—Encoder Auflösung	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	—	Encoder Inkremente	PGEARO
	2	UNSIGNED32	rw	—	Motorumdrehungen	BUSP7
6090h		ARRAY			Geschwindigkeits-Encoder Auflösung	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	—	Encoder Incremente/s	—
	2	UNSIGNED32	rw	—	Motorumdrehungen/s	—
6091h		ARRAY			Übersetzung	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	—	Motorumdrehungen	—
	2	UNSIGNED32	rw	—	Wellenumdrehungen	—
6092h		ARRAY			Vorschubkonstante	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	—	Vorschub	PGEARI
	2	UNSIGNED32	rw	—	Wellenumdrehungen	—
6093h		ARRAY			Position Faktor	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	—	Zähler	—
	2	UNSIGNED32	rw	—	Vorschubkonstante	—
6094h		ARRAY			Geschwindigkeits-Encoder Faktor	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	ro	—	Zähler	—
	2	UNSIGNED32	ro	—	Nenner	—
6097h		ARRAY			Beschleunigung Faktor	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	—	Zähler	—
	2	UNSIGNED32	rw	—	Nenner	—
6098h	0	INTEGER8	rw	—	Referenzfahrtart	NREF, DREF
6099h		ARRAY			Geschwindigkeit Referenzfahrt	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	rw	—	Geschwindigkeit während der Suche nach Endschaltern	VREF
	2	UNSIGNED32	rw	—	Geschwindigkeit während der Suche nach Nullmarken	VREF0

Index	Sub-Index	Datentyp	Zugriff	PDO mapp.	Beschreibung	ASCII Objekt
609Ah	0	UNSIGNED32	rw	—	Beschleunigung Referenzfahrt	ACCR, DECR
60C0h	0	INTEGER8	rw	—	Interpolation Sub-Mode Auswahl	—
60C1h		ARRAY			Interpolation Data Record	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	INTEGER32	rw	ja	x1, erster Parameter der ip Funktion	—
60C2h		RECORD			Interpolation Zeitspanne	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	PTBASE
	1	UNSIGNED8	rw	—	Interpolation Zeiteinheiten	—
	2	INTEGER16	rw	—	Interpolation Zeitindex	—
60C3h		ARRAY			Interpolation Sync Definition	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED8	rw	—	Synchronisiere auf Gruppe	—
	2	UNSIGNED8	rw	—	Ip Sync alle n Ereignisse	—
60C4h		RECORD			Interpolation Daten Konfiguration	—
	0	UNSIGNED8	ro	—	Anzahl der Einträge	—
	1	UNSIGNED32	ro	—	Maximale Speichergröße	—
	2	UNSIGNED32	rw	—	Aktuelle Speichergröße	—
	3	UNSIGNED8	rw	—	Speicher Organisation	—
	4	UNSIGNED16	rw	—	Speicher Position	—
	5	UNSIGNED8	w	—	Größe eines Datensatzes	—
60C5h	0	UNSIGNED32	rw	—	Maximale Systembeschleunigung	—
60C4h	6	UNSIGNED8	w	—	Speicher leeren	—
60F4h	0	INTEGER32	ro	ja	Ist-Schleppfehler	PE
60FDh	0	UNSIGNED32	ro	ja	Digitale Eingänge	IN1 .. IN4
60FFh	0	INTEGER32	rw	ja	Geschwindigkeits-Sollwert	J
6502h	0	UNSIGNED32	ro	—	Unterstützte CANopen – Betriebsarten	—

## 6.2 CANopen SDOs für Safety Erweiterungskarten S1/S2/S1-2/S2-2

Die Safety Erweiterungskarten S1, S2, S3, S4 können nur mit S700 Servoverstärkern ab Hardware Revision 2.10 verwendet werden.

### 6.2.1 Objekt 2400h: Safety card serial number

Die Seriennummer muss im Konfigurator angegeben werden, sie ist auf der Frontplatte der Safetykarte abgedruckt. Die Seriennummer ist in sechs 32-Bit Blöcke unterteilt, jedes Byte steht für ein ASCII Zeichen. Das Stringende wird durch 0h abgeschlossen.

Index	<b>2400h</b>
Name	Safety card serial number
Objektcode	Record
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	6
Defaultwert	6
Subindex	<b>1 bis 6</b>
Beschreibung	Safety card serial number part 1 to 6
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	—

### 6.2.2 Objekt 2401h: Safety card status

Index	<b>2401h</b>
Name	Safety card status
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	UNSIGNED32
Defaultwert	—

Bit	Bedeutung
0	Status LED 'POWER' auf Frontblende der Sicherheitskarte
1	Status LED 'RUN' auf Frontblende der Sicherheitskarte
2	Status LED 'CONFIG' auf Frontblende der Sicherheitskarte
3	Status LED 'FAULT' auf Frontblende der Sicherheitskarte
4	Zustand interner 'STO' (0 = STO inaktiv, 1 = STO aktiv)
5	OK-Meldung nach erfolgreichem Download der Konfigurationsdaten
8	Bit gesetzt, wenn Sicherheitskarte im Zustand 'STARTUP'
9	Bit gesetzt, wenn Sicherheitskarte im Zustand 'RUN'
10	Bit gesetzt, wenn Sicherheitskarte im Zustand 'STOP'
11	Bit gesetzt, wenn Sicherheitskarte im Zustand 'CONFIG'

### 6.2.3 Objekt 2402h: Safety card I/O status

Index	<b>2402h</b>
Name	Safety card I/O status
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	UNSIGNED32
Defaultwert	—

Bit	Bedeutung
0	Zustand Eingang 'SS1 Activate'
1	Zustand Eingang 'SS2 Activate'
2	Zustand Eingang 'SOS Activate'
3	Zustand Eingang 'SLS Activate'
4	Zustand Eingang 'SSR Activate'
5	Zustand Eingang 'SDI Left Activate'
6	Zustand Eingang 'SDI Right Activate'
7	Zustand Eingang 'SBT Activate'
8	Zustand Ausgang 'Ready'
9	Zustand Ausgang 'STO Acknowledge'
10	Zustand Ausgang 'SOS Acknowledge'
11	Zustand Ausgang 'SDI Acknowledge'
12	Zustand Ausgang 'Safe Range Acknowledge'
13	Zustand Ausgang 'SBT Acknowledge'
14	Zustand zweipoliger Ausgang 'Safe Brake Control' (ext. Bremse)
15	Zustand Ausgang 'STO SIL3' (zweiter Abschaltpfad)
16	Zustand Eingang 'SS1 SIL3/Reset'
29	Bremsrampensteuerung SS2 (0=antriebsgeführt, 1=steuerungsgeführt)
30	Bremsrampensteuerung SS1 (0=antriebsgeführt, 1=steuerungsgeführt)
31	NOTSTOP-Bremsrampe bei SS1 (0=nicht aktiviert, 1=aktiviert)

**6.2.4 Objekt 2403h: Safety card error register**

Index	<b>2403h</b>
Name	Safety card error register
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	UNSIGNED32
Defaultwert	0

Bit	Bedeutung
0	Fehler bei Sicherheitsfunktion 'Safe Stop 1' SS1
1	Fehler bei Sicherheitsfunktion 'Safe Stop 2' SS2
2	Fehler bei Sicherheitsfunktion 'Safe Operating Stop' SOS
3	Fehler bei Sicherheitsfunktion 'Safely Limited Speed' SLS
4	Fehler bei Sicherheitsfunktion 'Safe Speed Range' SSR
5	Fehler bei Sicherheitsfunktion 'Safe Direction' SDI
6	Fehler bei Sicherheitsfunktion 'Safe Brake Test' SBT
11	Fehler Eingangsbeschaltung digitale Eingänge
12	Fehler Beschaltung/Verdrahtung/Rücklesen digitale Ausgänge
13	Fehler Test Spannungsüberwachung
14	Fehler beim Watchdog-Test (Auslösen, Entladen)
15	Fehler beim Speichertest (Flash/RAM)
16	Timing Fehler (Überwachung Interrupt)
17	Überwachung Position durch Stromsignal
18	Überwachung Position durch Gebersignal (intern, extern, Sollwert)
19	Fehlerhafter Parameter (Servoverstärker)
20	Eingangszustände/Konfigurationsdaten passen nicht zueinander
21	Fehler Datenabgleich Kanal A-B
24	Fehler beim Schreiben/Löschen im Flash-Speicher
25	Zeitliche Synchronisation zwischen Kanal A und B
26	Watchdog hat ausgelöst
27	Überprüfung Daten/Parameter (Datenübertragung, Konfiguration)
28	Timeout beim Übertragen von Daten/Parameter
29	Fehler CRC-Berechnung
30	Fehler Überprüfung Gerätedaten (Seriennummer, Version)
31	Fehler System-Software (fatal)



### 6.2.5 Objekt 2404h: Safety card error stack error number

Fehlernummer der Einträge.

Der Fehlerstack kann über die SDOs 2404h bis 240Bh abgerufen werden. Dabei gehören immer die gleichen Subindizes zusammen. Ein kompletter Eintrag wäre also z.B. 2404h bis 240Bh jeweils der Subindex 5.

Index	<b>2404h</b>
Name	Safety card error stack error number
Objektcode	Record
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	0...128
Defaultwert	0
Subindex	<b>1 bis 128</b>
Beschreibung	Safety card error stack error number X
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	0

### 6.2.6 Objekt 2405h: Safety card error stack error time

Zeitindex bei der der Fehler aufgetreten ist.

Index	<b>2405h</b>
Name	Safety card error stack error time
Objektcode	Record
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	0...128
Defaultwert	0
Subindex	<b>1 bis 128</b>
Beschreibung	Safety card error stack error time X
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	0

**6.2.7 Objekt 2406h: Safety card error stack error index**

Detailfehlernummer bzw. Subfehlernummer.

Index	<b>2406h</b>
Name	Safety card error stack error index
Objektcode	Record
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	0...128
Defaultwert	0
Subindex	<b>1 bis 128</b>
Beschreibung	Safety card error stack error index X
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	0

**6.2.8 Objekt 2407h: Safety card error stack error info**

Zeigt an bei welchem Kanal der Fehler aufgetreten ist und ob er aktuell noch an steht.

Bit 0 = Kanal A

Bit 1 = Kanal B

Bit 2 = Fehler steht aktuell an

Index	<b>2407h</b>
Name	Safety card error stack error info
Objektcode	Record
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	0...128
Defaultwert	0
Subindex	<b>1 bis 128</b>
Beschreibung	Safety card error stack error info X
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	0

**6.2.9 Objekt 2408h: Safety card error stack error parameter 1**

Spezifische Zusatzinformationen für die Fehlernummer der Safety Karte.

Index	<b>2408h</b>
Name	Safety card error stack error parameter 1
Objektcode	Record
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	0...128
Defaultwert	0
Subindex	<b>1 bis 128</b>
Beschreibung	Safety card error stack error parameter 1 of error X
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	0

**6.2.10 Objekt 2409h: Safety card error stack error parameter 2**

Spezifische Zusatzinformationen für die Fehlernummer der Safety Karte.

Index	<b>2409h</b>
Name	Safety card error stack error parameter 2
Objektcode	Record
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	0...128
Defaultwert	0
Subindex	<b>1 bis 128</b>
Beschreibung	Safety card error stack error parameter 2 of error X
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	0

**6.2.11 Objekt 240Ah: Safety card error stack error parameter 3**

Fehlernummerspezifische Zusatzinformationen.

Index	<b>240Ah</b>
Name	Safety card error stack error parameter 3
Objektcode	Record
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	0...128
Defaultwert	0
Subindex	<b>1 bis 128</b>
Beschreibung	Safety card error stack error parameter 3 of error X
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	0

**6.2.12 Objekt 240Bh: Safety card error stack error parameter 4**

Fehlernummerspezifische Zusatzinformationen.

Index	<b>240Bh</b>
Name	Safety card error stack error parameter 4
Objektcode	Record
Kategorie	optional
Subindex	<b>0</b>
Beschreibung	Anzahl Einträge
Datentyp	UNSIGNED8
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	0...128
Defaultwert	0
Subindex	<b>1 bis 128</b>
Beschreibung	Safety card error stack error parameter 4 of error X
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	ro
PDO Mapping	nicht möglich
Wertebereich	UNSIGNED32
Defaultwert	0

### 6.2.13 Objekt 240Ch: Actual speed

Aktuelle Geschwindigkeit in 4096/Sekunde.

Index	<b>240Ch</b>
Name	Safety card actual speed
Objektcode	VAR
Datentyp	UNSIGNED32
Kategorie	optional
Zugriff	ro
PDO Mapping	möglich
Wertebereich	UNSIGNED32
Defaultwert	—

## 6.3 Beispiele

Alle Beispiele gelten für den Servoverstärker S300/S700. Alle Werte sind hexadezimal.

### 6.3.1 Prinzipielle Prüfung des CAN Verbindungsaufbaus

Beim Einschalten des S300/S700 wird eine Boot-up Message auf den Bus gesendet. Wenn sich im Bussystem kein geeigneter Empfänger findet, wird dieses Telegramm fortlaufend weiter gesendet. Kann ein vorhandener CAN-Master diese Nachricht nicht erkennen, überprüfen Sie die Kommunikation:

- Überprüfung des Buskabels: richtiger Wellenwiderstand, korrekte Abschlusswiderstände an beiden Enden?
- Multimeterüberprüfung des Ruhepegels der Busleitungen CAN-H und CAN-L gegenüber der CAN-GND-Leitung (ca. 2.5 V).
- Oszilloskop-Überprüfung der Ausgangssignale an CAN-H und CAN-L am S300/S700. Werden Signale auf den Bus gesendet? Spannungsdifferenz zwischen CAN-H und CAN-L bei logischer "0" ca. 2-3 V.
- Werden die Signale bei angeschlossenem Master nicht mehr weitergesendet?
- Überprüfen der Master-Hardware!
- Überprüfung der Master-Software!

### 6.3.2 Beispiel: Bedienung der Zustandsmaschine

**INFO**

Die Zustandsmaschine muss beim Hochfahren sequentiell bedient werden. Ein Überspringen eines Zustandes ist nicht möglich.

Nach dem Einschalten des S300/S700 und dem Erkennen der Boot-Up-Message kann die Kommunikation über SDOs aufgenommen werden, z.B. können so Parameter abgefragt oder geschrieben werden oder die Zustandsmaschine des Antriebs gesteuert werden.

Der Zustand der Zustandsmaschine kann über die Abfrage des Objekts 6041 Sub 0 erkannt werden. Direkt nach dem Einschalten erhält man dann z.B. als Antwortwert ein 0240h. Dies entspricht dem Zustand „Switch on disabled“.

Auf dem CAN-Bus würde man als Daten folgendes sehen:

COB-ID	Control Byte	Index		Subindex	Daten	Kommentar
		Low Byte	High Byte			
603	40	41	60	00h	40 00 00 00	
583	4B	41	60	00h	40 02 00 00	
	2 Byte Daten				status	

Ist die Leistungsspannung vorhanden und das Hardware-Enable auf High-Signal (24 V gegen DGND) kann durch Schreiben des Controlwords (Objekt6040 Sub 0) versucht werden, den Antrieb in den Zustand „Switched on“ zu schalten. Im Erfolgsfall wird dies in der SDO-Antwort positiv quittiert (Controlbyte 0 im Datenfeld = 60h).

#### Switch on

Die Nachrichten sehen dann wie folgt aus:

COB-ID	Control Byte	Index		Subindex	Daten	Kommentar
		Low Byte	High Byte			
603	2B	40	60	00h	06 00 00 00	Shut down
583	60	40	60	00h	00 00 00 00	
603	2B	40	60	00h	07 00 00 00	Switch on
583	60	40	60	00h	00 00 00 00	

control word = 0x0007:

Bit 0, Bit 1, Bit 2 set => Switch On, Disable Voltage off, Quick Stop off

#### Status Abfrage 2

Der neue Zustand kann dann wieder abgefragt werden und liefert das folgende Ergebnis:

COB-ID	Control Byte	Index		Subindex	Daten	Kommentar
		Low Byte	High Byte			
603	40	41	60	00h	—	query status
583	4B	41	60	00h	33 02 00 00	

Status = 0x0233:

Bit 0, Bit 1, Bit 5 set => ready to Switch On, Bit 9 set => Bedienung über RS232 auch möglich

### 6.3.3 Beispiel: Tipbetrieb überSDO

Der Motor soll mit einer konstanten Drehzahl arbeiten.

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
603	2F	60	60	00h	03 00 00 00	Betriebsart "Profile Velocity"
583	60	60	60	00h	00 00 00 00	
603	23	FF	60	00h	00 00 00 00	Sollwert=0
583	60	FF	60	00h	00 00 00 00	
603	2B	40	60	00h	06 00 00 00	shutdown
583	60	40	60	00h	00 00 00 00	
603	2B	40	60	00h	07 00 00 00	switch on
583	60	40	60	00h	00 00 00 00	
603	2B	40	60	00h	0F 00 00 00	enable operation
583	60	40	60	00h	00 00 00 00	
603	23	FF	60	00h	00 41 00 00	Sollwert=16640dec / PGEARI=U/s für PGEARI=10000, Sollwert=1.664 s <sup>-1</sup> 99.84 U/min
583	60	FF	60	00h	00 00 00 00	
603	2B	40	60	00h	0F 01 00 00	Zwischenstopp
583	60	40	60	00h	00 00 00 00	

### 6.3.4 Beispiel: Drehmoment-Modus über SDO

Der Motor soll ein konstantes Drehmoment abgeben. In diesen Fall ist es sinnvoll die maximal zu erreichende Drehzahl des Motors über den Parameter ICMDVLIM zu begrenzen. Der Parameter kann über die Bediensoftware eingestellt werden.

Beispiel:

ICMDVLIM 300 (= Begrenzung der maximalen Drehzahl auf 300 U/min)

SAVE

COLDSTART

CAN Daten:

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
603	2F	60	60	00h	04 00 00 00	Betriebsart "Torque"
583	60	60	60	00h	00 00 00 00	
603	2B	71	60	00h	00 00 00 00	Sollwert=0
583	60	71	60	00h	00 00 00 00	
603	2B	40	60	00h	06 00 00 00	shutdown
583	60	40	60	00h	00 00 00 00	
603	2B	40	60	00h	07 00 00 00	switch on
583	60	40	60	00h	00 00 00 00	
603	2B	40	60	00h	0F 00 00 00	enable operation
583	60	40	60	00h	00 00 00 00	
603	2B	71	60	00h	90 01 00 00	Sollwert 400 mA
583	60	71	60	00h	00 00 00 00	
603	2B	40	60	00h	0F 01 00 00	Zwischenstopp
583	60	40	60	00h	00 00 00 00	

### 6.3.5 Beispiel: Tippbetrieb über PDO

Generell ist es sinnvoll, nicht benutzte PDOs abzuschalten. Im Mode Drehzahl digital wird ein digitaler Drehzahlsollwert über RXPDO übertragen. Ist-Position und Ist-Drehzahl werden über ein SYNC-getriggertes TXPDO gelesen.

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low	High			
603	2F	60	60	00h	03 00 00 00	Betriebsart "Profile Velocity"
583	60	60	60	00h	00 00 00 00	
603	2F	00	16	00h	00 00 00 00	Löschen der Einträge für das erste RXPDO
583	60	00	16	00h	00 00 00 00	
603	23	00	16	01h	20 00 FF 60	Mapping RXPDO1, Objekt60FF, Sub-Index 0 Drehzahlsollwert, Datenlänge 32bit
583	60	00	16	01h	00 00 00 00	
603	2F	00	16	00h	01 00 00 00	Zahl der gemappten Objekte bestätigen
583	60	00	16	00h	00 00 00 00	
603	2F	00	1A	00h	00 00 00 00	Löschen der Einträge für das erste TXPDO
583	60	00	1A	00h	00 00 00 00	
603	23	00	1A	01h	20 00 64 60	mapping TXPDO1/1, Objekt6064, Sub-Index 0 current Positionswert in SI Einheiten, Datenlänge 32bit
583	60	00	1A	01h	00 00 00 00	
603	23	00	1A	02h	20 00 6C 60	mapping TXPDO1/2, Objekt606C, Sub-Index 0 current Drehzahlwert, Datenlänge 32Bit
583	60	00	1A	02h	00 00 00 00	
603	2F	00	1A	00h	02 00 00 00	Zahl der gemappten Objekte prüfen
583	60	00	1A	00h	00 00 00 00	
603	2F	00	18	02h	01 00 00 00	Einstellen TXPDO1 auf synchron, Übertragung mit jedem SYNC
583	60	00	18	02h	00 00 00 00	
603	23	01	18	01h	83 02 00 C0	disable TPDO2, setze Bit 31 (80h)
583	60	01	18	01h	00 00 00 00	
603	23	02	18	01h	83 03 00 C0	disable TPDO3
583	60	02	18	01h	00 00 00 00	
603	23	03	18	01h	83 04 00 C0	disabled TPDO4
583	60	03	18	01h	00 00 00 00	
603	23	01	14	01h	03 03 00 80	disabled RPDO2
583	60	01	14	01h	00 00 00 00	
603	23	02	14	01h	03 04 00 80	disabled RPDO3
583	60	02	14	01h	00 00 00 00	
603	23	03	14	01h	03 05 00 80	disabled RPDO4
583	60	03	14	01h	00 00 00 00	
000					01 03	Freigabe NMT
603	2B	40	60	00h	06 00 00 00	shutdown
583	60	40	60	00h	00 00 00 00	
603	2B	40	60	00h	07 00 00 00	switch on
583	60	40	60	00h	00 00 00 00	
603	2B	40	60	00h	0F 00 00 00	enable operation
583	60	40	60	00h	00 00 00 00	
203					00 40	Sollwert V= 98,3 U/min; Berechnung: 4000h=16384dec=N <sub>soll</sub> ;10000=PGEARI; (N <sub>soll</sub> /60)xPGEARI=V (16384/10000)x60=98,3 U/min
080						sende SYNC
183					FE 45 01 00 A6 AB 1A 00	Antwort, Position und Nist: Pos.= 00 01 45 FE = 83454 [Si Dim]; Nist = (00 1A AB A6) / 17894,4dec = 97,7 U/min 17894,4 ist der Umrechnungsfaktor.
603	2B	40	60	00h	0F 01 00 00	Zwischenstopp
583	60	40	60	00h	00 00 00 00	



### 6.3.6 Beispiel: Drehmoment-Modus über PDO

Generell ist es sinnvoll nicht benutzte PDOs abzuschalten. Weiterhin soll das erste TXPDO den Stromistwert mit jedem SYNC Telegramm übertragen.

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
603	2F	60	60	00h	04 00 00 00	Betriebsart "Torque"
583	60	60	60	00h	00 00 00 00	
603	2F	00	16	00h	00 00 00 00	Löschen der Einträge für das erste RXPDO
583	60	00	16	00h	00 00 00 00	
603	23	00	16	01h	10 00 71 60	Mapping RXPDO1, Objekt6071, Sub-Index 0 Stromsollwert, Datenlänge 16 Bit
583	60	00	16	01h	00 00 00 00	
603	2F	00	16	00h	01 00 00 00	Zahl der gemappten Objekte prüfen
583	60	00	16	00h	00 00 00 00	
603	2F	00	1A	00h	00 00 00 00	Löschen der Einträge für das erste TXPDO
583	60	00	1A	00h	00 00 00 00	
603	23	00	1A	01h	10 00 77 60	Mapping TXPDO1, Objekt6077, Sub-Index 0 Stromistwert, Datenlänge 16bit
583	60	00	1A	01h	00 00 00 00	
603	2F	00	1A	00h	01 00 00 00	Zahl der gemappten Objekte prüfen
583	60	00	1A	00h	00 00 00 00	
603	2F	00	18	02h	01 00 00 00	Einstellen TXPDO1 auf synchron, Übertragung mit jedem SYNC
583	60	00	18	02h	00 00 00 00	
603	23	01	18	01h	83 02 00 C0	disable TPDO2, Bit 31 (80h)
583	60	01	18	01h	00 00 00 00	
603	23	02	18	01h	83 03 00 C0	disable TPDO3
583	60	02	18	01h	00 00 00 00	
603	23	03	18	01h	83 04 00 C0	disabled TPDO4
583	60	03	18	01h	00 00 00 00	
603	23	01	14	01h	03 03 00 80	disabled RPDO2
583	60	01	14	01h	00 00 00 00	
603	23	02	14	01h	03 04 00 80	disabled RPDO3
583	60	02	14	01h	00 00 00 00	
603	23	03	14	01h	03 05 00 80	disabled RPDO4
583	60	03	14	01h	00 00 00 00	
000					01 03	enable NMT
603	2B	40	60	00h	06 00 00 00	shutdown
583	60	40	60	00h	00 00 00 00	
603	2B	40	60	00h	07 00 00 00	switch on
583	60	40	60	00h	00 00 00 00	
603	2B	40	60	00h	0F 00 00 00	enable operation
583	60	40	60	00h	00 00 00 00	
203					12 02	Sollwert 530 mA
080						sende SYNC
183					19 02	Istwert 537 mA
603	2B	40	60	00h	0F 01 00 00	Zwischenstopp
583	60	40	60	00h	00 00 00 00	

### 6.3.7 Beispiel: Referenzfahrt überSDO

Beim Betrieb des S300/S700 als Linearachse, muss, bevor Positionierungen durchgeführt werden können, ein Referenzpunkt festgelegt werden. Dies erfolgt durch das Ausführen einer Referenzfahrt im Homing Mode (0x6).

Hier wird beispielhaft das Vorgehen im Modus Homing aufgezeigt. Einige die Referenzfahrt betreffenden Parameter werden über den Bus eingestellt. Wenn man sicher sein kann, dass niemand die Parameter im Gerät geändert hat, kann dieser Teil entfallen, da die Geräte die Daten nullspannungssicher speichern können. Die Eingänge müssen als Endschalter konfiguriert sein. Da im DS402 die Einheiten-Parameter noch nicht abschließend definiert sind, müssen Sie die folgende Einstellung wählen.

PUNIT = 0 (counts)

VUNIT = 0 (counts/s)

ACCUNIT = 3 (counts/s<sup>2</sup>)

Die Reglergrundeinstellung haben Sie mit der Bediensoftware bereits vorgenommen. Die Auflösung ist in unserem Beispiel auf 10000 µm/Umdrehung eingestellt.

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
703	00					Boot-up Meldung
603	40	41	60	00h	00 00 00 00	Lese Profilizustand
583	4B	41	60	00h	40 02 00 00	
603	23	99	60	01h	10 27 00 00	vref=10000 counts/s bis zum Erreichen des Endschalters
583	60	99	60	01h	00 00 00 00	
603	23	99	60	02h	88 13 00 00	vref=5000 counts/s vom Endschalter zum Nullpunkt
583	60	99	60	02h	00 00 00 00	
603	23	9A	60	00h	10 27 00 00	Bremsund Beschl.-Rampen 1000-counts/s <sup>2</sup>
583	60	9A	60	00h	00 00 00 00	
603	23	7C	60	00h	A8 61 00 00	Referenz-Offset 25000counts
583	60	7C	60	00h	00 00 00 00	

Art der Referenzfahrt (6098)

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
603	2F	60	60	00h	06 00 00 00	Betriebsart = homing
583	60	60	60	00h	00 00 00 00	
603	40	41	60	00h	00 00 00 00	Lese Profilizustand, Antwort: 0250h Voltage Enabled
583	4B	41	60	00h	40 02 00 00	
603	2B	40	60	00h	06 00 00 00	Controlword Übergang_2, "ready to switch on". Shutdown
583	60	40	60	00h	00 00 00 00	
603	2B	40	60	00h	07 00 00 00	Übergang_3, "switch on". switch on
583	60	40	60	00h	00 00 00 00	
603	2B	40	60	00h	0F 00 00 00	Übergang_4, "operation enable"
583	60	40	60	00h	00 00 00 00	
603	40	41	60	00h	00 00 00 00	Lese Profilizustand
583	4B	41	60	00h	37 02 00 00	
603	2B	40	60	00h	1F 00 00 00	Homing_operation_start
583	60	40	60	00h	00 00 00 00	
603	40	41	60	00h	00 00 00 00	Lese Profilizustand, Antwort: Referenzierung nicht abgeschlossen
583	4B	41	60	00h	37 02 00 00	
603	40	41	60	00h	00 00 00 00	Lese Profilizustand, Antwort: Referenzierung abgeschlossen
583	4B	41	60	00h	37 16 00 00	

Das Bit 12 im SDO 6041 gibt an ob die Referenzierung abgeschlossen wurde. Das Lesen des Profilizustandes ist für den Funktionsablauf nicht zwingend notwendig, .

### 6.3.8 Beispiel: Starte Fahrauftrag über SDO aus dem internen Speicher des S300/S700

Dieses Beispiel setzt die Definition eines Fahrauftrages z.B. mit der Inbetriebnahmesoftware voraus. Vor dem Start von absoluten Fahraufträgen muss eine Referenzfahrt durchgeführt werden.

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
603	2F	60	60	00h	01 00 00 00	Betriebsart = Lage
583	60	60	60	00h	00 00 00 00	
603	2B	40	60	00h	06 00 00 00	Shutdown
583	60	40	60	00h	00 00 00 00	
603	2B	40	60	00h	07 00 00 00	Switch On
583	60	40	60	00h	00 00 00 00	
603	2B	40	60	00h	0F 00 00 00	Enable Operation
583	60	40	60	00h	00 00 00 00	
603	2B	80	20	00h	03 00 00 00	Anwahl Fahrsatz 3
583	60	80	20	00h	00 00 00 00	
603	2B	40	60	00h	3F 00 00 00	Start mit new SETPOINT und CHANGE_SET_IMMEDIATELY
583	60	40	60	00h	00 00 00 00	
603	40	81	20	00h	00 00 00 00	Lese gestarteten Fahrsatz. Antwort: Fahrsatz 3 wird abgearbeitet
583	4B	81	20	00h	03 00 00 00	

### 6.3.9 Beispiel: Bedienung des Profile Position Mode

Dieses Beispiel zeigt die Bedienung des Profile position mode. Dazu werden die PDOs wie folgt eingestellt:

#### Erstes RPDO

Für das erste RPDO ist kein Mapping erforderlich, da das Default Mapping das Controlword in das RXPDO1 einträgt

#### Zweites RPDO

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
603	2F	01	16	00h	00 00 00 00	RPDO2: Mapping löschen
583	60	01	16	00h	00 00 00 00	
603	23	01	16	01h	20 00 7A 60	RPDO2, Eintrag1: target_position
583	60	01	16	01h	00 00 00 00	
603	23	01	16	02h	20 00 81 60	RPDO2, Eintrag2: profile_velocity
583	60	01	16	02h	00 00 00 00	
603	2F	01	16	00h	02 00 00 00	Zahl der gemappten Objekte
583	60	01	16	00h	00 00 00 00	

#### Erstes TPDO

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
603	2F	00	1A	00h	00 00 00 00	TPDO1: Mapping löschen
583	60	00	1A	00h	00 00 00 00	
603	23	00	1A	01h	10 00 41 60	TPDO1, Eintrag 1: profile statusword
583	60	00	1A	01h	00 00 00 00	
603	2F	00	1A	00h	01 00 00 00	Zahl der gemappten Objekte
583	60	00	1A	00h	00 00 00 00	

#### Zweites TPDO

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
603	2F	01	1A	00h	00 00 00 00	TPDO2: Mapping löschen
583	60	01	1A	00h	00 00 00 00	
603	23	01	1A	01h	20 00 64 60	TPDO2, Eintrag 1: position_actual_value
583	60	01	1A	01h	00 00 00 00	
603	23	01	1A	02h	20 00 6C 60	TPDO2, Eintrag 2: velocity_actual_value
583	60	01	1A	02h	00 00 00 00	
603	2F	01	1A	00h	02 00 00 00	Zahl der gemappten Objekte
583	60	01	1A	00h	00 00 00 00	

Das zweite TPDO soll mit jedem SYNC vom Antrieb gesendet werden.

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
603	2F	01	18	02h	01 00 00 00	TPDO2 mit jedem SYNC
583	60	01	18	02h	00 00 00 00	

## Sperren der nichtbenötigten TPDOs

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
603	23	02	18	01h	83 03 00 C0	disable TPDO3
583	60	02	18	01h	00 00 00 00	
603	23	03	18	01h	83 04 00 C0	disable TPDO4
583	60	03	18	01h	00 00 00 00	

## Sperren der nichtbenötigten RPDOs

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
603	23	02	14	01h	03 04 00 80	disable RPDO3
583	60	02	14	01h	00 00 00 00	
603	23	03	14	01h	03 05 00 80	disable RPDO4
583	60	03	14	01h	00 00 00 00	

Vorgabe der mechanischen Auflösung über Objekt 6092h, Subindex 01h und 02h. Defaultwerte sind die antriebsspezifischen Faktoren PGEARI und PGEARO:

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
603	23	93	60	01h	00 00 10 00	2E20 Inkremente
583	60	93	60	01h	00 00 00 00	
603	23	93	60	02h	A0 8C 00 00	3600 Nutzereinheiten
583	60	93	60	02h	00 00 00 00	

Nach der Festlegung der PDOs können diese mit dem NMT freigegeben werden:

COB-ID	Daten	Kommentar
000	01 03	enable NMT
183	40 02	profile status

Nach diesen Einstellungen kann eine Referenzfahrt eingestellt und angestoßen werden.

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
603	2F	60	60	00h	06 00 00 00	Betriebsart = homing
583	60	60	60	00h	00 00 00 00	
603	2F	98	60	00h	0C 00 00 00	Referenzfahrt 12, negative Fahrtrichtung (DS402)
583	60	98	60	00h	00 00 00 00	
603	23	99	60	01h	40 19 01 00	Referenzfahrtgeschw. 72000 Dim/s=2s-1
583	80	99	60	01h	31 00 09 06	
603	2B	40	60	00h	06 00 00 00	Übergang_2,"ready to switch on". Shutdown
583	60	40	60	00h	00 00 00 00	
183					21 02	
603	2B	40	60	00h	07 00 00 00	Übergang_3,"switch on". Switch on
583	60	40	60	00h	00 00 00 00	
183					33 02	
603	2B	40	60	00h	0F 00 00 00	Controlword: Operation Enable
583	60	40	60	00h	00 00 00 00	
183					37 02	
603	2B	40	60	00h	1F 00 00 00	Referenzfahrt starten
583	60	40	60	00h	00 00 00 00	Antworttelegramm
183					37 06	Antwort: target reached
183					37 16	Antwort: homing attained

Beenden der Referenzfahrt über das Steuerwort 1\_RPDO

COB-ID	Daten	Kommentar
203	0F 00	

Umschalten in den Profile Position Mode und Rampen für Positionierungen vorgegeben

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
603	2F	60	60	00h	01 00 00 00	Betriebsart = Profile Positioning
583	60	60	60	00h	00 00 00 00	
603	23	83	60	00h	32 00 00 00	50ms Beschleunigungszeit
583	60	83	60	00h	00 00 00 00	
603	23	84	60	00h	32 00 00 00	50ms Bremszeit
583	60	84	60	00h	00 00 00 00	

Sollwert

COB-ID	Daten	Kommentar
303	20 4E 00 00	Pos 8CA0 =36000µm ; V= 20000 µm/s
080		sende SYNC
283	BB F8 FF FF	

Controlword mit „new setpoint“-Bit (bit 4) gesetzt

COB-ID	Daten	Kommentar
203	1F 00	

Warten

COB-ID	Daten	Kommentar
183	37 12	setpoint acknowledge

Controlword mit „new setpoint“-Bit (bit 4) zurückgesetzt

COB-ID	Daten	Kommentar
203	0F 00	
183	37 02	reset Setpoint acknowledge

Warten

COB-ID	Daten	Kommentar
183	37 06	Antwort: target reached
080		SYNC
283	92 FC FF FF	Antwort: 92 FC Position FF FF Geschwindigkeit

### 6.3.10 Beispiel: ASCII Kommunikation über SDO

In diesem Beispiel wird die P-Verstärkung des Drehzahlreglers auf 6 eingestellt. Der ASCII Befehl dazu lautet: „GV 6“.

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
603	2F	26	10	01h	47 00 00 00	sende ASCII Zeichen "G"
583	60	26	10	01h	00 00 00 00	
603	2F	26	10	01h	56 00 00 00	sende ASCII Zeichen "V"
583	60	26	10	01h	00 00 00 00	
603	2F	26	10	01h	20 00 00 00	sende ASCII Zeichen "SP" (Leerzeichen)
583	60	26	10	01h	00 00 00 00	
603	2F	26	10	01h	36 00 00 00	sende ASCII Zeichen "6"
583	60	26	10	01h	00 00 00 00	
603	2F	26	10	01h	0D 00 00 00	sende ASCII Zeichen "CR"
583	60	26	10	01h	00 00 00 00	
603	2F	26	10	01h	0A 00 00 00	sende ASCII Zeichen "LF"
583	60	26	10	01h	00 00 00 00	

### 6.3.11 Test für SYNC Telegramme

#### Konfiguration

Ziele:

- Zielposition und Verfahrensgeschwindigkeit einem PDO (2. RPDO) zuweisen.
- Aktuelle Position einem PDO (1. TPDO) zuweisen, generiert mit jedem zweiten SYNC.
- Statuswort und Herstellerstatus einem PDO (2. TPDO) zuweisen, generiert mit jedem dritten SYNC.

Telegramm mit den zugehörigen Antworten:

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low	High			
603	2F	01	16	00h	00 00 00 00	RPDO2: lösche Mapping
583	60	01	16	00h	00 00 00 00	
603	23	01	16	01h	20 00 7A 60	RPDO2, Eintrag 1: Zielposition
583	60	01	16	01h	00 00 00 00	
603	23	01	16	02h	20 00 81 60	RPDO2, Eintrag 2: Geschwindigkeit
583	60	01	16	02h	00 00 00 00	
603	2F	01	16	00h	02 00 00 00	RPDO2: Anzahl der gemappten Objekte
583	60	01	16	00h	00 00 00 00	
603	2F	00	1A	00h	00 00 00 00	TPDO1: lösche Mapping
583	60	00	1A	00h	00 00 00 00	
603	23	00	1A	01h	20 00 64 60	TPDO1: Eintrag 1: Ist-Position
583	60	00	1A	01h	00 00 00 00	
603	2F	00	1A	00h	01 00 00 00	TPDO1: Anzahl der gemappten Objekte
583	60	00	1A	00h	00 00 00 00	
603	2F	00	18	02h	02 00 00 00	TPDO1: sende mit jedem 2. SYNC
583	60	00	18	02h	00 00 00 00	
603	2F	01	1A	00h	00 00 00 00	TPDO2: lösche Mapping
583	60	01	1A	00h	00 00 00 00	
603	23	01	1A	01h	10 00 41 60	TPDO2: Eintrag 1: Statuswort
583	60	01	1A	01h	00 00 00 00	
603	23	01	1A	02h	20 00 02 10	TPDO2: Eintrag 2: Herstellerstatus
583	60	01	1A	02h	00 00 00 00	
603	2F	01	16	00h	02 00 00 00	TPDO2: Anzahl der gemappten Objekte
583	60	01	16	00h	00 00 00 00	
603	2F	01	18	02h	03 00 00 00	TPDO2: sende mit jedem 3. SYNC
583	60	01	18	02h	00 00 00 00	

#### SYNC-Objekt

COB-ID	Kommentar
080	Objekt 181 (TPDO 1) erscheint bei jedem 2. SYNC, Objekt 281 (TPDO 2) erscheint bei jedem 3. SYNC.

#### Emergency-Objekt

Wenn z.B. der Resolver Stecker nicht gesteckt ist, wird im Verstärker ein schwerwiegender Fehler ausgelöst. Ein Emergency Telegramm wird ausgelöst:

COB-ID	Emergency error		Error register	Daten	Kommentar
	Low	High			
081	10	43	08	00 00 00 00	Motor Temperatur, Temperatur, herstellenspezifisch
081	00	00	88	00 00 00 00	



### 6.3.12 Anwendung: Elektrisches Getriebe

Für die Verwendung des Modes elektrisches Getriebe über die CANopen-Schnittstelle sind die folgenden Parameter für die Einstellung des Slaves des elektrischen Getriebes relevant:

GEARI	Objekt 2040 sub 1
GEARO	Objekt 2040 sub 2
GEARMODE	Objekt 353F sub 1
EXTPOS	Objekt 3539 sub 1
IN20MODE	Objekt 3723 sub 1
IN20TRIG	Objekt 3724 sub 1 oder Objekt 20B0 sub 0 (auch mappbar)
ACCR	Objekt 3502 sub 1
DECR	Objekt 3524 sub 1

ACCR und DECR können auch gemeinsam über das Objekt 609A (auch mappbar) eingestellt werden.

Die Übersetzung des elektrischen Getriebes wird über die Getriebefaktoren GEARI und GEARO eingestellt. Der Eingang der Masterimpulse wird über die Parameter GEARMODE und EXTPOS definiert.

Folgende Bits haben in der Betriebsart elektrisches Getriebe eine besondere Bedeutung: CANopen Statusword Bit 10 = 1: Slaveachse ist in Synchronisation

CANopen Controlword Bit 13 = 1: Stopp der Mitfahrt der Slaveachse

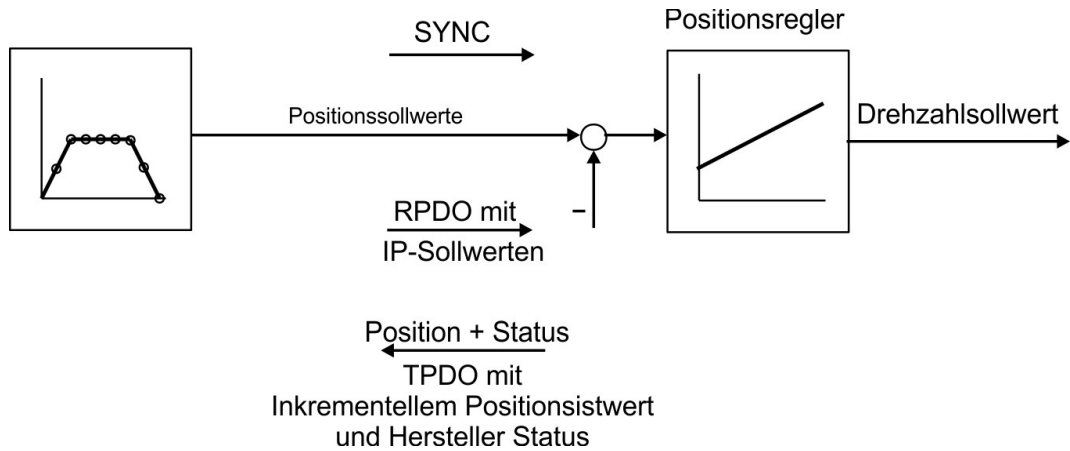
Über den virtuellen Eingang 20 kann bei folgenden Eingangsfunktionen (IN20MODE) die Art der Aufsynchonisierung und der Aussynchonisierung auf den Master verändert werden:

- IN20MODE 42:** Die Slaveachse wird über die Rampe ACCR aufsynchonisiert und über DECR aussynchonisiert.
- IN20MODE 43:** Beim Start der Slaveachse wird die Masterposition gelatcht. Die Slaveachse wird dann über die Rampe ACCR auf die Mastergeschwindigkeit gebracht und anschließend die Wegdifferenz zum Master ausgeglichen. Aussynchonisiert wird über DECR. Über IN20TRIG kann zusätzlich ein Positionsoffset auf die Masterposition vergeben werden.
- IN20MODE 51:** Über IN20TRIG kann eine Wegstrecke vorgegeben werden, auf der der Slave sich einsynchonisieren soll. Aussynchonisiert wird über DECR.

### 6.3.13 Anwendungen: Externe Trajektorie mit Interpolated Position Modus

Dieses Beispiel zeigt, wie die Positions-Sollwerte für zwei Achsen mit einem PDO übertragen werden können.

Reglerstruktur der Lageregelung im Servoverstärker:



#### Beschreibung

Alle Daten sind hexadezimal. Im Beispiel haben die beiden Achsen im System die Stationsadressen 1 und 2.

#### Voraussetzungen

- Für den IP-Modus muss die interne Synchronisation verwendet werden. Dafür muss der Parameter SYNCSRC (Objekt 3683 sub 1) auf 3 gesetzt werden.
- Die Parameters müssen im EEPROM gespeichert werden.
- Ein Kaltstart muss durchgeführt werden, um die Synchronisation freizugeben.
- Die Achsen müssen referenziert sein.

Das gemeinsame PDO beinhaltet 2 IP (interpolierte Position) Sollwerte und kann simultan an zwei Stationen übertragen werden, wobei jede Station die für sie relevanten Daten extrahieren kann.

Die anderen Daten können mit Dummy-Daten zur Nichtbeachtung kenntlich gemacht werden (Objekt 2100 sub 0). Um dies zu erreichen, müssen beide Achsen auf dieselbe RPDO-COB-ID reagieren.

**Durchführung**

RPDO Mapping für beide Achsen.

Achse 1:

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
601	2F	01	16	00h	00 00 00 00	RPDO2: Lösche Mapping
581	60	01	16	00h	00 00 00 00	
601	23	01	16	01h	20 01 C1 60	RPDO2, Eintrag 1: IP Sollwert Achse 1
581	60	01	16	01h	00 00 00 00	
601	23	01	16	02h	20 00 00 21	RPDO2, Eintrag 2: Dummy 4 Byte
581	60	01	16	02h	00 00 00 00	
601	2F	01	16	00h	02 00 00 00	RPDO2, Anzahl der gemappten Objekte
581	60	01	16	00h	00 00 00 00	

Achse 2:

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
602	2F	01	16	00h	00 00 00 00	RPDO2: Lösche Mapping
582	60	01	16	00h	00 00 00 00	
602	23	01	16	01h	20 00 00 21	RPDO2, Eintrag 1: Dummy 4 Byte
582	60	01	16	01h	00 00 00 00	
602	23	01	16	02h	20 01 C1 60	RPDO2, Eintrag 2: IP Sollwert Achse 2
582	60	01	16	02h	00 00 00 00	
602	2F	01	16	00h	02 00 00 00	RPDO2, Anzahl der gemappten Objekte
582	60	01	16	00h	00 00 00 00	
602	23	01	14	01h	01 03 00 00	RPDO2: Setze COB-ID identisch zu Achse 1
582	60	01	14	01h	00 00 00 00	

Jetzt reagieren beide Achsen auf denselben COB-Identifizier 0x301, Achse 1 nimmt Bytes 0 bis 3 als IP Sollwert, Achse 2 nimmt Bytes 4 bis 7. Das zweite TPDO soll die Ist-Position in Inkrementen und den Herstellerstatus beinhalten.

TPDO Mapping für Achse 1:

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
601	2F	01	1A	00h	00 00 00 00	TPDO2: Lösche Mapping
581	60	01	1A	00h	00 00 00 00	
601	23	01	1A	01h	20 00 63 60	TPDO2, Eintrag 1: Ist-Position in Inkrementen
581	60	01	1A	01h	00 00 00 00	
601	23	01	1A	02h	20 00 02 10	TPDO2, Eintrag 2: Dummy 4 Byte
581	60	01	1A	02h	00 00 00 00	
601	2F	01	1A	00h	02 00 00 00	TPDO2, Anzahl der gemappten Objekte
581	60	01	1A	00h	00 00 00 00	

Führen Sie dasselbe Mapping für Achse 2 durch.

Wir nehmen an, dass beide Verstärker neue Trajektorie-Werte mit jedem SYNC akzeptieren und ihre inkrementellen Positionswerte und Herstellerstati mit jedem SYNC zurückgeben. Die Kommunikationsparameter müssen entsprechend eingestellt werden:

Achse 1:

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
601	2F	01	14	02h	01 00 00 00	RPDO2 Achse 1, Reaktion bei jedem SYNC
581	60	01	14	02h	00 00 00 00	
602	2F	01	14	02h	01 00 00 00	RPDO2 Achse 2, Reaktion bei jedem SYNC
582	60	01	14	02h	00 00 00 00	
601	2F	01	18	02h	01 00 00 00	TPDO2 Achse 1, Reaktion bei jedem SYNC
581	60	01	18	02h	00 00 00 00	
602	2F	01	18	02h	01 00 00 00	TPDO2 Achse 2, Reaktion bei jedem SYNC
582	60	01	18	02h	00 00 00 00	

Die anderen TPDOs 3 und 4 sollten abgeschaltet werden um die Busbelastung zu minimieren:

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
601	23	02	18	01h	81 03 00 C0	TPDO3 abschalten
581	60	02	18	01h	00 00 00 00	
601	23	03	18	01h	81 04 00 C0	TPDO4 abschalten
581	60	03	18	01h	00 00 00 00	

Führen Sie dasselbe für Achse 2 durch.

Um Trajektorien Fahrten durchzuführen, müssen beide Servoverstärker in der korrekten Betriebsart arbeiten. Dies wird eingestellt durch Index 6060h:

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
601	2F	60	60	00h	07 00 00 00	Setze IP Modus für Achse 1
581	60	60	60	00h	00 00 00 00	
602	2F	60	60	00h	07 00 00 00	Setze IP Modus für Achse 2
582	60	60	60	00h	00 00 00 00	

Die Zykluszeit für den IP-Modus soll 1 ms lang sein. Dies muss mit Objekt 60C1 sub 1 und 2 definiert werden:

COB-ID	Control Byte	Index		Sub-index	Daten	Kommentar
		Low Byte	High Byte			
601	2F	C2	60	01h	01 00 00 00	Interpolationszeit Einheit 1
581	60	C2	60	01h	00 00 00 00	
601	2F	C2	60	02h	FD 00 00 00	Interpolationszeit Index 3 => Zykluszeit = 1 * 10 <sup>-3</sup> s
581	60	C2	60	02h	00 00 00 00	

Führen Sie dasselbe für Achse 2 durch.

Um die Achsen zu starten, muss der Servoverstärker in den Betriebsmodus gebracht werden (operation enable) und die Netzwerk Management Funktionen müssen gestartet werden.

Die Netzwerk Management Funktionen geben die Anwendung der Process Data Objects (PDOs) und werden von folgendem Telegramm für beide Achsen initialisiert:

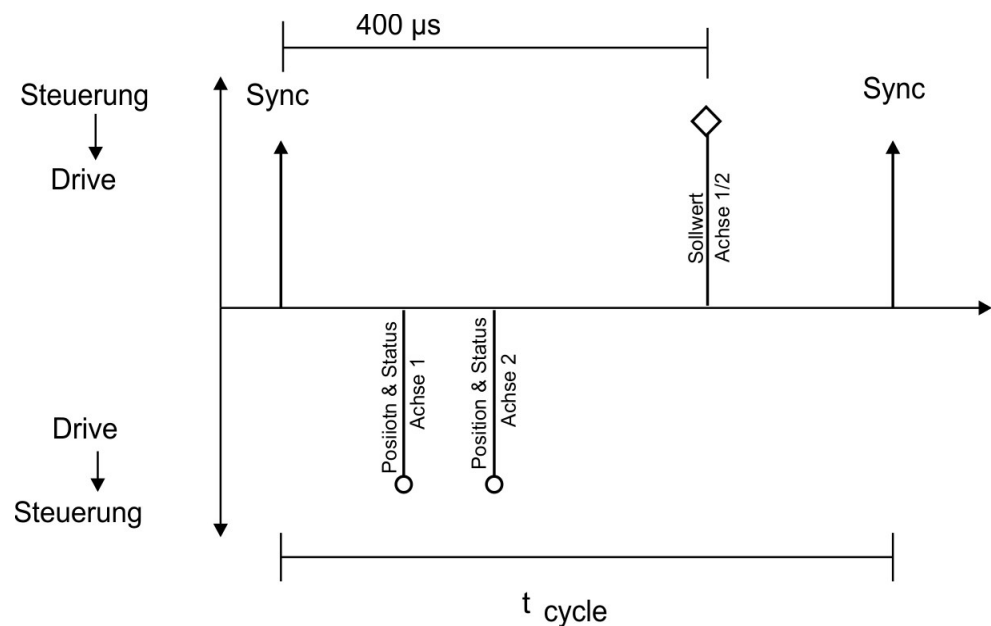
Schalte NMT (Network Management) Status Maschine in operation enable:

COB-ID	Command specifier (CS)	Node-ID	Kommentar
0	1	1	NMT freigegeben für alle Achsen

Als nächstes wird die Spannung für jeden Verstärker eingeschaltet und sie werden in den Betriebszustand operation enable geschaltet. Dies sollte schrittweise durchgeführt werden mit Warten auf entsprechende Antwort des Antrieb (z.B. von Achse 1):

COB-ID	Daten	Kommentar
201	06 00	Shutdown command
181	31 02	State Ready_to_switch_on
201	07 00	Switch_on command
181	33 02	State Switched_on
201	0F 00	Enable_operation command
181	37 02	State Operation_enabled
201	1F 00	Enable IP-mode
181	37 12	IP-mode enable

Diese Konfiguration gibt nun eine zyklische Sequenz frei:



z.B. 2 Achsen

$t_{\text{cycle}}$  1 ms pro Achse bei 1 Mbaud

RPDO 2 kann jetzt für die Übertragung von Trajektorien Daten für beide Achsen verwendet werden, z.B.:

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
301	F4	01	00	00	E8	03	00	00

Hier erhält die erste Achse den Trajektorien-Wert 500 Inkremente (Bytes 0 bis 3) und die zweite Achse erhält den Wert 1000 Inkremente (Bytes 4 bis 7).

Die Achsen akzeptieren diese Werte und die Positionierung wird durchgeführt, wenn das nächste SYNC Telegramm empfangen wird.

**SYNC Telegramm**

COB-ID
301

Afterwards, both axes send back their incremental positions and the contents of their status registers when the SYNC Objekt with the COB-ID for the 2nd TPDO is received:

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Kommentar
181	23	01	00	00	00	00	03	44	Position + Hersteller Status Register Achse 1
182	A5	02	00	00	00	00	03	44	Position + Hersteller Status Register Achse 2

Wenn während des Vorgangs ein Fehler auftritt, sendet die betroffene Achse eine Emergency Meldung, die so aussehen könnte:

**Emergency Objekt**

COB-ID	Emergency error		Error register	Kategorie		
	Low	High				
081	10	43	08	01	00 00 00 00	Motor Temperatur, Temperatur, herstellerspezifisch
081	00	00	08	00	00 00 00 00	

## 7 Index

### 1

1000h	28
1001h	28
1002h	29
1003h	30
1005h	31
1006h	31
1008h	31
1009h	32
100Ah	32
100Ch	32
100Dh	32
1010h	33
1011h	34
1014h	34
1016h	35
1017h	35
1018h	36
1026h	37
1400-1403h	51
1600-1603h	52
1800-1803h	53
1A00-1A03h	55

### 2

2000h	38
2014-2017h	39
2030h	39
2040h	40
2041h	41
2051h	42
2052h	43
2053h	43
2061h	44
2080h	44
2081h	44
2082h	45
2083h	45
2090h	45
20A0h	46
20A1h	46
20A2h	46
20A3h	47
20A4h	47
20B0h	47
20B1h	48
20B2h	48
2100h	49
2101h	49
2400h	110
2401h	110
2402h	111
2403h	112
2404h	113
2405h	113
2406h	114
2407h	114
2408h	115
2409h	115

240Ah	116
240Bh	116
240Ch	117

### 3

3500h bis 39xxh	92
-----------------	----

### 6

6040h	58
6041h	60
6060h	61
6061h	62
6063h	74
6064h	75
6065h	75
6067h	75
6068h	76
606Ch	72
6071h	73
6073h	73
6077h	73
607Ah	86
607Ch	82
607Dh	86
607Fh	87
6080h	87
6081h	87
6083h	88
6084h	88
6085h	88
6086h	89
6089h	63
608Ah	63
608Bh	69
608Ch	69
608Dh	70
608Eh	70
608Fh	64
6091h	65
6092h	66
6093h	67
6094h	68
6097h	71
6098h	83
6099h	84
609Ah	84
60C0h	77
60C1h	78
60C2h	79
60C3h	80
60C4h	80
60C5h	89
60F4h	76
60FDh	49
60FFh	72
6502h	50

### A

Abkürzungen	10
Allgemeine Definitionen	28
ASCII Kommando Referenz	94

<b>B</b>			
Basisdatentypen .....	17		
Beispiele .....	117		
Beispiele, Spezielle Anwendungen .....	130		
Bestimmungsgemäße Verwendung .....	10		
Busleitung .....	13		
<b>C</b>			
CANopen Antriebsprofil .....	27		
Control word .....	58		
<b>D</b>			
Data Frame .....	15		
Datentypen .....	17		
Device control (dc) .....	56		
<b>E</b>			
Emergency Messages .....	27		
Emergency Object (EMCY) .....	19		
Erweiterte Datentypen .....	18		
<b>F</b>			
Factor Groups (fg) .....	62		
<b>G</b>			
Grundfunktionen .....	11		
<b>H</b>			
Heartbeat .....	26		
Homing Mode (hm) .....	82		
<b>I</b>			
Inbetriebnahme .....	12		
Installation .....	12		
Interface .....	13		
Interpolated Position Mode (ip) .....	77		
<b>K</b>			
Kommunikationsobjekte .....	18		
Kommunikationsprofil .....	15		
Kommunikationsstörungen .....	11		
Konfigurationsparameter .....	14		
<b>M</b>			
Mapping .....	50		
<b>N</b>			
Network Management Objects (NMT) .....	19		
Nodeguard .....	25		
<b>O</b>			
Objektkanal .....	92		
Objektverzeichnis .....	103		
		Operating mode .....	61
<b>P</b>			
PDO Konfiguration .....	50		
Position Control Function (pc) .....	74		
Process Data Object (PDO) .....	23		
Profile Position Mode (pp) .....	85		
Profile Torque Mode (tq) .....	73		
Profile Velocity Mode (pv) .....	72		
<b>R</b>			
Receive PDOs (RXPDO) .....	51		
Remote Frame .....	15		
Response monitoring .....	32		
<b>S</b>			
SDO Abort Codes .....	23		
Service Data Objects (SDO) .....	21		
Stationsadresse .....	12		
Status word .....	60		
Synchronization Object (SYNC) .....	19		
<b>T</b>			
TIME .....	19		
Time-Stamp Object .....	19		
Transmit PDOs (TXPDO) .....	53		
Triggermodi .....	24		
<b>U</b>			
Übertragungsmodi .....	24		
Übertragungsrate .....	12		
Übertragungsverfahren .....	11		
<b>W</b>			
Weiterführende Dokumentation .....	9		
<b>Z</b>			
Zielgruppe .....	9		
Zusammengesetzte Datentypen .....	17		
Zustandsmaschine .....	56		



--/--

## Service

Kollmorgen bietet seinen Kunden einen umfassenden Kundendienst.



Besuchen Sie das [Kollmorgen Developer Network](#). Stellen Sie Fragen an die Community, durchsuchen Sie die "Knowledge Base", laden Sie Dateien herunter und schlagen Sie Verbesserungen vor.



### Europa

KOLLMORGEN

Internet: [www.kollmorgen.com/de-de](http://www.kollmorgen.com/de-de)

E-Mail: [technik@kollmorgen.com](mailto:technik@kollmorgen.com)

Tel.: +49 - 2102 - 9394 - 0

Fax: +49 - 2102 - 9394 - 3155



### Nordamerika

KOLLMORGEN

Internet: [www.kollmorgen.com/en-us](http://www.kollmorgen.com/en-us)

E-Mail: [support@kollmorgen.com](mailto:support@kollmorgen.com)

Tel.: +1 - 540 - 633 - 3545

Fax: +1 - 540 - 639 - 4162



### Südamerika

KOLLMORGEN

Internet: [www.kollmorgen.com/pt-br](http://www.kollmorgen.com/pt-br)

E-Mail: [contato@kollmorgen.com](mailto:contato@kollmorgen.com)

Tel.: +55 - 11 - 4615-6300



### Asien

KOLLMORGEN

Internet: [www.kollmorgen.cn](http://www.kollmorgen.cn)

E-Mail: [sales.china@kollmorgen.com](mailto:sales.china@kollmorgen.com)

Tel: +86 - 400 668 2802

Fax: +86 - 21 6248 5367

