

COMBICOM



CiA

CAN in AUTOMATION

F5

Betriebsanleitung

CANopen-Anschaltung

Originalanleitung		
Document	Part	Version
20103622	DEU	01

KEB

Inhaltsverzeichnis

1.	Vorwort	5
1.1	Hinweise auf besondere Maßnahmen.....	5
1.2	Dokumentation.....	5
1.3	Gültigkeit und Haftung	6
1.4	Urheberrecht	7
1.5	Bestimmungsgemäßer Gebrauch	7
1.6	Produktbeschreibung.....	7
1.7	Bestellbezeichnung	8
2.	Hardwarebeschreibung.....	9
2.1	Diagnoseschnittstelle.....	10
2.2	CAN-Schnittstelle	10
2.3	Grundlegendes zum CAN-BUS.....	11
3.	Funktionen	12
3.1	Charakteristik des High-Speed-PDO.....	14
3.2	Charakteristik des Low-Speed-PDO	14
3.3	Prozessdatenabbildung	14
3.4	CANopen Bootup-Sequenz.....	15
3.5	Bootup-Message.....	16
3.6	Node-Guarding.....	16
3.7	Life-Guarding	17
3.8	Emergency Objekt	17
4.	Kodierung der Daten in den vier CAN-Telegramm-Typen.....	18
4.1	SDO(rx)-Telegramm	18
4.1.1	Initiate Domain Download Request (Schreibenanforderung des Master).....	19
4.1.2	Initiate Domain Upload Request (Leseanforderung des Master).....	19
4.2	SDO(tx)-Telegramm	19
4.2.1	Initiate Domain Download Response (Schreibbestätigung vom FU).....	19
4.2.2	Initiate Domain Upload Response (Lesebestätigung vom FU).....	20
4.2.3	Abort Domain Transfer (Fehlerantwort vom FU).....	20
4.3	PDO1(rx)-Telegramm	21
4.4	PDO1(tx)-Telegramm	21
4.5	PDO2(rx)-Telegramm	21
4.6	PDO2(tx)-Telegramm	22
5.	Operatorparameter	23
5.1	Von KEB definierte Parameter.....	23
5.2	Vom Kommunikationsprofil [12] definierte Parameter	29
5.3	Parameter für das Life-Guarding.....	44
5.4	Parameter der Emergency-Bearbeitung	46
5.5	Parameter für den Synchronmodus.....	47

6.	Zugriff auf Operatorparameter über die Diagnoseschnittstelle	48
7.	Umschaltung des transmission-type der PDOs	58
7.1	Asynchron herstellerspezifisch (Wert = 254d/FEh) oder Asynchron profilspezifisch (Wert = 255d/FFh)	58
7.2	Synchron azyklisch (Wert = 0) oder synchron zyklisch (Werte = 1 bis 240)	58
7.3	Synchron / asynchron RTROnly (Werte = 252, 253)	58
8.	Synchronmodus	59
8.1	Funktionseinschränkungen im Synchronmodus	60
9.	DSP402-Unterstützung.....	61
9.1	Voreinstellungen für DSP402-Betrieb	61
9.2	Hinweise zu den DSP402-Velocity Rampen	62
9.3	DSP402-Profil und Synchronmodus	62
9.4	Allgemeine Parameter des DSP402-Profiles	63
9.5	Parameter des Velocity Mode	69
10.	Faktoren	75
10.1	Weitergehende Umrechnungen.....	76
10.2	Beispiel für die Bestimmung der Faktoren	77
10.2.1	Faktor 0: Anwender-Weg-Einheiten in Inkremente	77
10.2.2	Faktor 1: Anwender-Geschwindigkeit-Einheiten in $0,125 \text{ min}^{-1}$	77
10.2.3	Faktor 2: Anwender-Beschleunigung-Einheiten in eine KEB-Rampenzeit	78
11.	Anhang	79
11.1	CAN-Bit-Timing	79
11.1.1	Wichtiger Warnhinweis	79
11.2	Literaturverzeichnis.....	80
11.3	Übersicht der Operatorparameter nach CANopen	80
11.4	Kompakt-Übersicht der CAN-Kommunikation.....	82
11.5	F5 Operator interne Fehlermeldungen	83

1. Vorwort

Die beschriebene Hard- und Software sind Entwicklungen der KEB Automation KG. Die beigefügten Unterlagen entsprechen dem bei Drucklegung gültigen Stand. Druckfehler, Irrtümer und technische Änderungen vorbehalten.

1.1 Hinweise auf besondere Maßnahmen

Die in dieser Anleitung verwendeten Hinweise entsprechen folgender Bedeutung:

Gefahr		Wird verwendet, wenn Tod oder schwere Körperverletzung die Folge von Nichtbeachtung der Maßnahme sein kann.
Warnung		Wird verwendet, wenn Körperverletzung und/oder erheblicher Sachschaden die Folge von Nichtbeachtung der Maßnahme sein kann.
Vorsicht		Wird verwendet, wenn Sachschaden die Folge von Nichtbeachtung der Maßnahme sein kann.
Achtung		Wird verwendet, wenn ein störanfälliger oder unerwünschter Betrieb die Folge von Nichtbeachtung der Maßnahme sein kann.
Info		Wird verwendet, wenn ein besseres oder einfacheres Ergebnis die Folge der Maßnahme sein kann.

Die Hinweise können für den speziellen Fall durch zusätzliche Piktogramme und Texte ergänzt werden.

1.2 Dokumentation

Achtung  Dokumentation über www.keb.de	
	Das Herunterladen und Lesen der Dokumentation und insbesondere der Sicherheits- und Anwendungshinweise ist vor jeglichen Arbeiten mit dem Gerät zwingend erforderlich. Die Dokumentation ist wie folgt erhältlich.
Schritt 1	Materialnummer (Mat.No.) vom Typenschild ablesen
Schritt 2	<p>Materialnummer auf www.keb.de => Service => Downloads eingeben und auf „suchen“ klicken.</p> <p>Downloads</p> <div style="background-color: #f0f0f0; padding: 10px; border: 1px solid #ccc;"> <p>Suche nach Materialnummern</p> <p>Bitte geben Sie eine vollständige (11-stellige) Materialnummer ein.</p> <p>Suche nach: <input type="text" value="15G6DCD-3510"/> <input type="button" value="suchen"/></p> </div>
weiter auf nächster Seite	

Schritt 3	Daraufhin wird sämtliche zum Gerät gehörige Dokumentation in Deutsch und Englisch angezeigt. Sofern verfügbar, werden weitere Sprachen angezeigt. Es ist sicherzustellen, dass der Anwender die zur Verfügung gestellte Sprache versteht.
	Sollten Sie keine Möglichkeit haben, die Dokumentation zu lesen oder zu verstehen, unterlassen Sie alle weiteren Schritte und informieren Sie unseren Support für weitere Unterstützung.

Die Nichtbeachtung der Sicherheits- und Anwendungshinweise führt zum Verlust jeglicher Schadensersatzansprüche. Die in dieser Anleitung angeführten Warn- und Sicherheitshinweise wirken nur ergänzend. Sie bieten keinen Anspruch auf Vollständigkeit.

1.3 Gültigkeit und Haftung

Die Verwendung unserer Geräte in den Zielprodukten erfolgt außerhalb unserer Kontrollmöglichkeiten und liegt daher ausschließlich im Verantwortungsbereich des Maschinenherstellers, Systemintegrators oder Kunden.

Die in den technischen Unterlagen enthaltenen Informationen, sowie etwaige anwendungsspezifische Beratung in Wort, Schrift und durch Versuche, erfolgen nach bestem Wissen und Kenntnissen über die Applikation. Sie gelten jedoch nur als unverbindliche Hinweise. Dies gilt auch in Bezug auf eine etwaige Verletzung von Schutzrechten Dritter.

Eine Auswahl unserer Produkte im Hinblick auf ihre Eignung für den beabsichtigten Einsatz hat generell durch den Anwender zu erfolgen.

Prüfungen und Tests können nur im Rahmen der Applikation vom Maschinenhersteller erfolgen. Sie sind zu wiederholen, auch wenn nur Teile von Hardware, Software oder die Geräteeinstellung modifiziert worden sind.

Gefahr  durch unbefugte Eingriffe	
	Unbefugtes Öffnen und unsachgemäße Eingriffe in das Gerät können zu Tod, schweren Körperverletzungen, Sachschäden sowie Fehlfunktionen führen. Modifikation oder Instandsetzung ist nur durch von KEB autorisiertem Personal zulässig. Zuwiderhandlung hebt die Haftung für daraus entstehende Folgen auf.

Der Haftungsausschluss gilt insbesondere auch für Betriebsunterbrechungsschäden, entgangenen Gewinn, Datenverlust oder sonstige Folgeschäden. Mit dem Haftungsausschluss erlischt die Gewährleistung. Dies gilt auch, wenn wir vorab auf die Möglichkeit solcher Schäden hingewiesen worden sind.

Sollten einzelne Bestimmungen nichtig, unwirksam oder undurchführbar sein oder werden, so wird hiervon die Wirksamkeit aller sonstigen Bestimmungen oder Vereinbarungen nicht berührt.

Durch Vielzahl der Einsatzmöglichkeiten konnte nicht jeder denkbare Fall der Aufstellung, des Betriebes oder der Instandhaltung berücksichtigt werden. Sollten Sie weitere Informationen wünschen, oder sollten besondere Probleme auftreten, die in der Dokumentation nicht ausführlich genug behandelt werden, können Sie die erforderliche Auskunft über die örtliche Vertretung der Firma KEB Automation KG anfordern.

1.4 Urheberrecht

Der Kunde darf die Betriebsanleitung sowie weitere gerätebegleitenden Unterlagen oder Teile daraus für betriebseigene Zwecke weiterverwenden. Die Urheberrechte liegen bei KEB und bleiben auch in vollem Umfang bestehen.

KEB®, COMBIVERT®, COMBICONTROL® und COMBIVIS® sind eingetragene Marken der KEB Automation KG.

Andere Wort- und/oder Bildmarken sind Marken (™) oder eingetragene Marken (®) der jeweiligen Inhaber und werden beim ersten Auftreten in der Fußnote erwähnt.

Bei der Erstellung unserer Unterlagen achten wir mit größtmöglicher Sorgfalt auf die Rechte Dritter. Sollten wir eine Marke nicht gekennzeichnet oder ein Copyright missachtet haben, bitten wir sie, uns davon in Kenntnis zu setzen, damit wir die Möglichkeit der Nachbesserung wahrnehmen können.

1.5 Bestimmungsgemäßer Gebrauch

Diese Betriebsanleitung beschreibt den F5 CAN Operator.

Die bei der KEB Automation KG eingesetzten Halbleiter und Bauteile sind für den Einsatz in industriellen Produkten entwickelt und ausgelegt. Wenn das Produkt in Maschinen eingesetzt wird, die unter Ausnahmebedingungen arbeiten, lebenswichtige Funktionen, lebenserhaltende Maßnahmen oder eine außergewöhnliche Sicherheitsstufe erfüllen, ist die erforderliche Zuverlässigkeit und Sicherheit durch den Maschinenbauer sicherzustellen und zu gewährleisten.

Der Betrieb unserer Produkte außerhalb der in den technischen Daten angegebenen Grenzwerte führt zum Verlust jeglicher Schadensersatzansprüche.

1.6 Produktbeschreibung

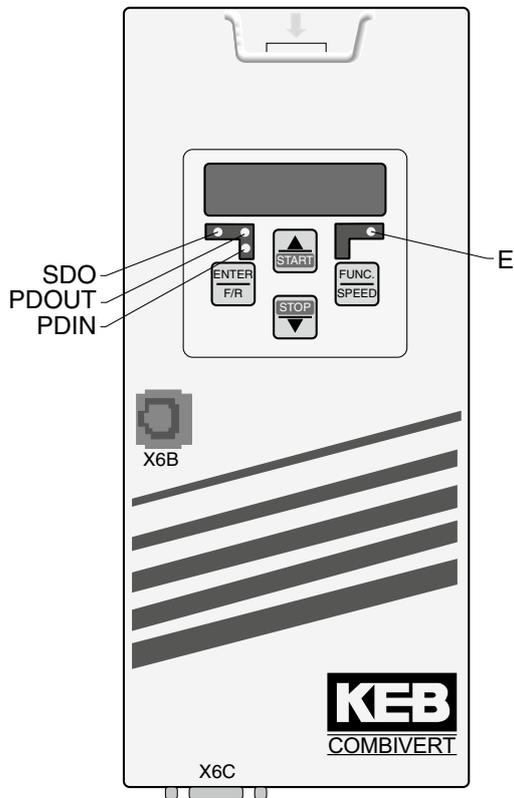
KEB-Antriebstechnik entwickelt, produziert und vertreibt weltweit statische Frequenzumrichter im industriellen Leistungsbereich. Die Umrichter des Typs F5 können optional mit einer CAN (Controller-Area-Network)-Schnittstelle ausgerüstet werden. Es handelt sich hierbei um eine intelligente Schnittstelle, die den Zugriff über CAN auf die Parameter des Frequenzumrichters steuert.

Der F5-CAN-Operator wird durch Einstecken in das FU-Gehäuse integriert und passt in alle KEB-F5-Frequenzumrichter. Parallel zum Feldbusbetrieb ist die Bedienung über die integrierte Anzeige / Tastatur sowie eine weitere Schnittstelle zur Diagnose/Parametrierung (KEB COMBIVIS) möglich.

1.7 Bestellbezeichnung

Diese Bedienungsanleitung:	CCF50D0-K002
F5 CAN Operator mit Anzeige und Tastatur:	00F5060-5010
F5 CAN Operator ohne Anzeige und Tastatur:	00F5060-5110
F5 CAN Operator mit Anzeige und Tastatur (Klemmleiste):	00F5060-5014
Zubehör für Diagnoseschnittstelle	
HSP5 Kabel zwischen PC und Adapter:	00F50C0-0001
Adapter DSUB9 / Western:	00F50C0-0002

2. Hardwarebeschreibung

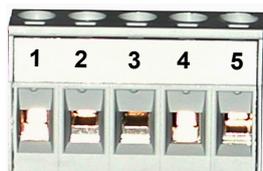


Tastatur und Anzeige nur bei: 00.F5.060-5010 und 00.F5.060-5014	
SDO (grün):	SDO-Kommunikation aktiv
PDO (grün):	PDO-Daten werden zur FU-Steuerung geschrieben.
PDIN (grün):	<ul style="list-style-type: none"> • PDIN-Daten werden von der FU-Steuerung gelesen. • Leuchtet kurz nach einem Reset-Kommando über CAN auf.
E (rot):	An → Umrichter betriebsbereit Blinkt → Umrichter in Fehler Aus → keine Versorgungsspannung
X6B:	Diagnoseschnittstelle zum PC (s. Kapitel 4.1)
X6C:	CAN-Schnittstelle (Buchsen-Stecker)



Wahlweise erhältlich mit Klemmleiste

X6D:	CAN-Schnittstelle als 5polige Klemmleiste (optional erhältlich)
------	---



Pinbelegung CAN:

Pin	Signal
1	V- (Bezugspotential externe Versorgungsspannung)*
2	CAN_L
3	Schirm
4	CAN_H
5	V+ (externe Versorgungsspannung)

2.1 Diagnoseschnittstelle

	<p>Um eine Zerstörung der PC-Schnittstelle zu vermeiden, darf die Diagnoseschnittstelle nur über ein spezielles HSP5-Kabel mit Spannungsanpassung an einen PC angeschlossen werden!</p>
---	---

An die Diagnoseschnittstelle wird über einen Adapter ein HSP5-Kabel angeschlossen (siehe Kapitel 2 Bestellbezeichnungen). Über die PC-Software KEB COMBIVIS 5 kann nun auf alle Umrichterparameter normal zugegriffen werden. Die Operator-internen Parameter können ebenfalls ausgelesen und zum Teil eingestellt oder mittels Download parametrierbar werden.

2.2 CAN-Schnittstelle

Die CAN-Schnittstelle besteht aus einem D-SUB-9pol.-Stift-Stecker (nach DIN41652 Teil 1). Die Belegung der CAN-Stecker gemäß [2]:

X6C	Signal	Beschreibung	SUB-D9 Buchse (Draufsicht)
1	-	Reserviert	
2	CAN_L	CAN-Bussignal dominant low	
3	CAN_GND	Hier nicht angeschlossen	
4	-	Reserviert	
5	(CAN_SHLD)	Hier nicht angeschlossen	
6	(GND)	Hier nicht angeschlossen	
7	CAN_H	CAN-Bussignal dominant high	
8	-	Reserviert	
9	(CAN_V+)	Hier nicht angeschlossen	

Übertragungspegel auf CAN:	Nach ISO/DIS 11898, ISO-High Speed
Übertragungsgeschwindigkeit auf CAN:	Einstellbar über CAN (10, 20, 25, 50, 100, 125, 250, 500, 800, 1000 Kbit/s)
Potentialtrennung:	Sichere Trennung nach VDE0160.
Busabschluss:	124 Ohm , muss extern erfolgen (zwischen Pin 2 und 7).

2.3 Grundlegendes zum CAN-BUS

Es soll an dieser Stelle das System des CAN (Controller-Area-Network)-BUS vorgestellt und dabei einige Begriffe erläutert werden, die im folgenden oft Verwendung finden.

Der CAN ist ein Multi-Master-System, d.h. jeder Teilnehmer kann auf den BUS zugreifen und Telegramme absenden. Damit bei gleichzeitigem Zugriff zweier Teilnehmer keine ungültigen Zustände entstehen, kennt der CAN-BUS eine sogenannte Arbitrierungs- (Schlichtungs-) Phase, die den Telegrammanfang bestimmt. Bei Zugriffskonflikten erkennen alle Teilnehmer während dieser Arbitrierung, wer die niedrigste Telegrammnummer (Identifizier) sendet. Dieser Teilnehmer kann dann sein Telegramm vollständig, ohne von vorne beginnen zu müssen, weitersenden. Alle anderen (sendewilligen) Teilnehmer gehen dann in den Empfangsstatus über und brechen ihr Telegramm zunächst ab. Somit ist festgelegt, dass niedrigere Telegrammnummern automatisch Vorrang haben vor höheren. Die Anzahl der Telegrammnummern ist beim CAN Version 2.0A begrenzt auf 2032 Identifizier (0...2031).

Die CAN-Telegramme können maximal 8 Byte Anwender-Daten enthalten.

Wenn im Folgenden der Begriff logischer CAN-Master gebraucht wird, so ist damit der CAN-Teilnehmer gemeint, dem die Steuerung des Gesamt-CAN-Systems obliegt. Auch wenn es physikalisch beim CAN nur Master gibt, so wird es in den meisten Einsatzfällen doch einen oder mehrere Teilnehmer geben, die die Kontrolle haben. Der KEB-Frequenzumrichter ist in diesem Zusammenhang als Befehlsempfänger (logischer Slave) zu sehen.

3. Funktionen

Das CAN-Protokoll ist bis zur Datensicherungsschicht einheitlich standardisiert. Die Abarbeitung dieses Protokolls übernimmt vollständig ein CAN-Controller. Weiterhin hat der CAN in Automation Verein (CiA) einen Standard für die höhere Protokollschicht verabschiedet, der mit CAN Application Layer (CAL) bezeichnet wurde. Auf diesem Standard aufbauend wurde dann im September 1995 das „CAL-based Communication Profile“ (CiA,DS301) veröffentlicht. Dieser Standard bildet die Basis für alle CANopen-Geräte-Profile. In diesem Standard wird eine bestimmte Untermenge des CAL-Standards ausgewählt. Das Kommunikationsprofil definiert u.a. ein Minimum Capability Device. Das ist die minimal erforderliche Funktionalität, die ein CANopen-Knoten zur Verfügung stellen muss. Die vorliegende CAN-Anschaltung realisiert ein solches Minimum Capability Device.

Ein wichtiger Punkt in jedem CAN-Netzwerk ist die Vergabe der Telegrammnummern (Identifier), zumal deren Anzahl bei CAN V2.0 A auf 2032 begrenzt ist. Im CAL-Standard ist hierzu ein eigenes Verfahren definiert, das diese Vergabe dynamisch über ein eigenes Protokoll (DBT = Distributor) abwickelt. Dieses relativ aufwendige Verfahren zur Vergabe der Identifier ist für ein Minimum Capability Device nicht zwingend vorgeschrieben und in der KEB-CAN-Anschaltung nicht integriert. Für diesen Fall wird im Kommunikationsprofil ein einfacheres Verfahren zur Absprache der Identifiervergabe festgelegt. Dieses Verfahren wird auch von der KEB-CAN-Anschaltung unterstützt und sieht wie folgt aus:

Jeder Frequenzumrichter erhält eine eindeutige CAN-Adresse, die Node_Id. Es gibt bei KEB zwei Möglichkeiten für die Quelle dieser Node_Id.

- Wenn der Wert des Parameters OP_Node_Id den Wert 255 (dez) besitzt:
→ Node_Id = Umrichteradresse (Sy.06) +1
- In allen anderen Fällen gibt der Wert des Parameters OP_Node_Id selbst den Wert für Node_Id vor:
→ Node_Id = OP_Node_Id

	Nach Auslieferung haben alle KEB-Frequenzumrichter die Umrichteradresse = 1. Sollen mehrere KEB-Frequenzumrichter über CAN vernetzt werden, müssen diesen zunächst unterschiedliche Umrichteradressen vorgegeben werden. Dies geschieht z. B. über die Tastatur am Operator.
---	--

Jedem Frequenzumrichter werden sechs Identifier zugeordnet.

Über einen Identifier kann ein beliebiger CAN-Knoten das Lesen oder das Schreiben eines Parameterwertes anfordern (Request-Identifier).

Ein weiterer Identifier ist reserviert für die zugehörige Antwort vom Frequenzumrichter (Response-Identifier). Der Mechanismus von Anforderung und Antwort wird auch als bestätigter Dienst bezeichnet. Das CANopen-Kommunikationsprofil fasst diese Funktion unter dem Begriff Service-Data-Object (SDO) zusammen:

$$\text{SDO(rx)} = \text{Request-Identifier} = 1536 + \text{Node_Id}$$

$$\text{SDO(tx)} = \text{Response-Identifier} = 1408 + \text{Node_Id}$$

Beispiel: Node_Id= 30 → Schreib/Leseanforderungen über Identifier = 1566(dez)
→ Schreib/Lesebestätigungen über Identifier = 1438(dez)

Hinweis: Grundsätzlich reicht die Funktion des SDO vollständig aus, um den KEB F5-Frequenzumrichter über CAN zu steuern. Jeder Parameterwert im Umrichter kann hierüber verändert oder erfragt werden.

Über den 3. Identifier kann der CAN-Master dem Frequenzumrichter Daten unadressiert und unbestätigt vorgeben. In Anlehnung an die Datenrichtung vom Master zum Slave wird dieser im Folgenden als OUT1-Identifier bezeichnet.

Über den 4. Identifier leitet der Frequenzumrichter seinerseits neue Daten unadressiert und unbestätigt an den CAN-Master (IN1-Identifier).

Diese Funktionalität wird vom Kommunikationsprofil als Process-Data-Object (PDO) bezeichnet. Die beiden Objektteile werden mit PDO1(rx) und PDO1(tx) benannt.

$$\begin{aligned} \text{PDO1(rx)} &= \text{Out-Identifier} &= 512 + \text{Node_Id} \\ \text{PDO1(tx)} &= \text{IN-Identifier} &= 384 + \text{Node_Id} \end{aligned}$$

Ab der Softwareversion 1.3 ist die PDO-Funktionalität in der KEB F5-CAN-Anschaltung zweimal vorhanden. Dieses sog. 2. PDO belegt dann die Identifier fünf bis sechs:

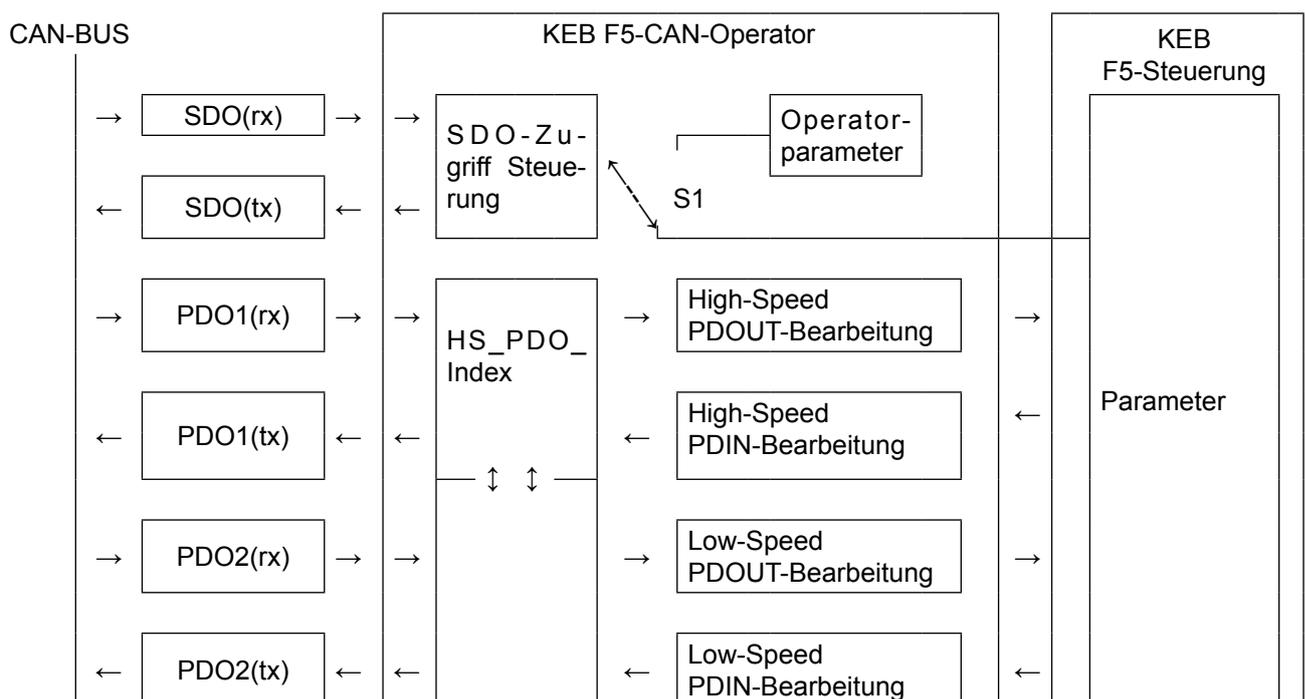
$$\begin{aligned} \text{PDO2(rx)} &= \text{Out-Identifier} &= 768 + \text{Node_Id} \\ \text{PDO2(tx)} &= \text{IN-Identifier} &= 640 + \text{Node_Id} \end{aligned}$$

Über den 5. Identifier kann der CAN-Master dem Frequenzumrichter Daten unadressiert und unbestätigt vorgeben. In Anlehnung an die Datenrichtung vom Master zum Slave wird dieser im Folgenden als OUT2-Identifier bezeichnet.

Über den 6. Identifier leitet der Frequenzumrichter seinerseits neue Daten unadressiert und unbestätigt an den CAN-Master (IN2-Identifier).

Die beiden PDO's sind zwar bzgl. der Verwaltung nach Außen identisch, unterscheiden sich aber deutlich in der Form der internen Bearbeitung. Nur eines der beiden kann wie in früheren Software-Versionen als High-Speed-PDO bearbeitet werden. Das hinzugekommene PDO ist von der Abarbeitung her gleichgestellt mit den SDO-Kommandos und wird im Folgenden als Low-Speed-PDO bezeichnet. Welches der beiden PDO's das High-Speed-PDO ist, kann eingestellt werden. Nach Auslieferung ist wie bisher das erste PDO 'High-Speed' und das zweite PDO abgeschaltet. Bestehende CAN-Applikationen müssen also nicht geändert werden.

Die CAN-Anschaltung steuert den Datenfluss vom CAN-BUS SDO(rx), PDO1(rx) und PDO2(rx) hin zur Frequenzumrichtersteuerung und ebenso vom Frequenzumrichter zum CAN-BUS SDO(tx), PDO1(tx) und PDO2(tx):



Das obige Bild zeigt die Funktion der CAN-Anschaltung. Die Stellung von Schalter S1 wird allein von der im CAN-SDO(rx)-Telegramm enthaltenen Parameter-Adresse (16 Bit Index plus 8 Bit Subindex) bestimmt. In einem bestimmten Index-Bereich liegen die sog. Konfigurationsdaten der CAN-Anschaltung. Diese Parameter bestimmen das Verhalten der CAN-Anschaltung und sind deshalb auch in dieser realisiert. Zugriffe auf Parameter im Index-Bereich 2000(hex) bis 5EFF(hex) werden als Schreib-/Leseaufträge zur Umrichter-Steuerung weitergeleitet.

3.1 Charakteristik des High-Speed-PDO

- Die Prozessdatenabbildung liegt in der Umrichtersteuerung. Die entsprechenden Parameter liegen in der System-Parametergruppe (SY). Da die Kodierung des PD-Mapping des Umrichters nicht mit der von CANopen übereinstimmt, wird diese vom CAN-Operator entsprechend automatisch umgewandelt.
- Das Vorgeben neuer Prozessausgangsdaten vom CAN wird auf nur einen speziellen Prozessdatendienst zur Umrichtersteuerung umgesetzt.
- Die minimale Zykluszeit für neue Prozessausgangsdaten liegt bei ca. 3ms.
- Das zyklische Lesen der Prozesseingangsdaten wird durch nur einen speziellen Prozessdaten-Lesedienst durchgeführt.
- Die minimal erreichbare Zykluszeit für das Lesen der Prozesseingangsdaten liegt bei ca. 3ms.
- Nicht alle Parameter der Umrichtersteuerung können auf das High-Speed-PDO abgebildet werden.

3.2 Charakteristik des Low-Speed-PDO

- Prozessdatenabbildung wird allein vom CAN-Operator verwaltet.
- Das Vorgeben neuer Prozessausgangsdaten vom CAN wird auf 'n' einfache Dienste (wie SDO-Kommandos) zur Umrichtersteuerung umgesetzt, wobei 'n' der Anzahl abgebildeter Parameter im PDO-Mapping entspricht.
- Die minimale Zykluszeit für neue Prozessausgangsdaten liegt bei ca. 'n' * 5 ms.
- Das zyklische Lesen der Prozesseingangsdaten wird durch 'n' einfache Lesedienste durchgeführt, wobei 'n' der Anzahl abgebildeter Parameter im PDO-Mapping entspricht.
- Die minimal erreichbare Zykluszeit für das Lesen der Prozesseingangsdaten liegt bei ca. 'n' * 5 ms.
- Alle Parameter der Umrichtersteuerung können auf das Low-Speed-PDO abgebildet werden.

3.3 Prozessdatenabbildung

Die Bestimmung des Ziels für die Daten in den PDO(rx)-Telegrammen bzw. der Quelle für die Daten in den PDO(tx)-Telegrammen hält sich vollständig an die Vorschriften des CANopen-Kommunikationsprofils [12]. Hierbei definiert für jede Datenrichtung jeweils ein komplex aufgebautes Objekt (Parameter) die PDO-Abbildung (PDO-Mapping).

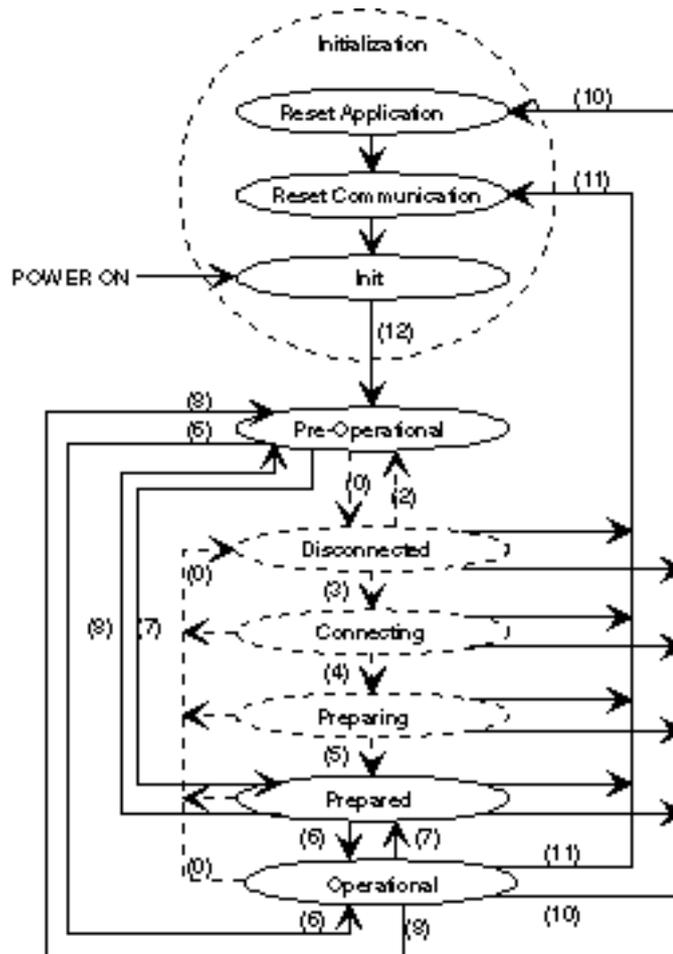
Ein weiteres Objekt je Datenrichtung bestimmt die Kommunikationsdefinitionen (PDO Communication Parameter). Siehe Parameterbeschreibungen von

- | | | | |
|---|-----------------------------------|---|-----------------------------------|
| - | 1st receive PDO Mapping | - | 2nd receive PDO Mapping |
| - | 1st transmit PDO Mapping | - | 2nd transmit PDO Mapping |
| - | 1st receive PDO Parameter | - | 2nd receive PDO Parameter |
| - | 1st transmit PDO Parameter | - | 2nd transmit PDO Parameter |

in dieser Bedienungsanleitung.

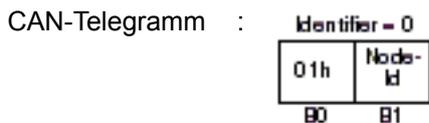
3.4 CANopen Bootup-Sequenz

Die KEB-CAN-Anschaltung geht automatisch nach der Initialisierungsphase in den Status Pre-Operational. In diesem Status ist bereits Kommunikation über das SDO(rx) und SDO(tx) mit den Diensten Domain Download (Parameter Schreiben) und Domain Upload (Parameter Lesen) aktiviert. Lediglich die Prozessdatenkommunikation ist in diesem Status noch inaktiv. Diese wird dann durch das NMT-Kommando Start_Remote_Node() freigegeben (Bild). Das Ziel dieser Startsequenz ist der Betriebszustand Operational. In diesem Status ist die Kommunikation vollständig aktiviert. Adressiert werden beim NMT-Protokoll bestimmte CAN-Knoten durch die oben bereits erwähnte Node-Id.



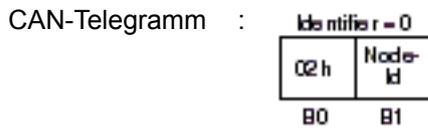
Die KEB-CANopen-Anschaltung realisiert folgende in obigem Schaubild mit durchgezogener Linie gezeichnete Übergänge:

6: Start_Remote_Node()



- Node_Id = 0 (alle NMT-Slaves sind angesprochen) oder
- Node_Id = Umrichteradresse + 1 (nur 1 Frequenzumrichter ist angesprochen)

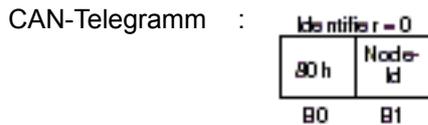
7: Stop_Remote_Node()



Node_Id = 0 (alle NMT-Slaves sind angesprochen) oder

Node_Id = Umrichteradresse + 1 (nur 1 Frequenzumrichter ist angesprochen)

8: Enter_Pre-Operational_State()



Node_Id = 0 (alle NMT-Slaves sind angesprochen) oder

Node_Id = Umrichteradresse + 1 (nur 1 Frequenzumrichter ist angesprochen)

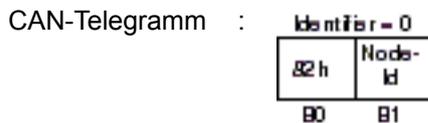
10: Reset_Node(): Beim Ausführen dieser Funktion wird ein Software-Reset in der KEB-CAN-Anschaltung durchgeführt.



Node_Id = 0 (alle NMT-Slaves sind angesprochen) oder

Node_Id = Umrichteradresse + 1 (nur 1 Frequenzumrichter ist angesprochen)

11: Reset_Communication(): Funktion wie bei Reset_Node().



Node_Id = 0 (alle NMT-Slaves sind angesprochen) oder

Node_Id = Umrichteradresse + 1 (nur 1 Frequenzumrichter ist angesprochen)

12: Enter Pre-Operational automatically(): s.o.

3.5 Bootup-Message

Der KEB-F5-CAN-Operator gibt eine Bootup-Message aus, wenn nach POWER ON die Initialisierungsphase abgeschlossen ist. Dies ist ein Telegramm auf Identifier = 1792 + Node_Id mit der Datenlänge = 1 und dem Wert = 0.

3.6 Node-Guarding

In [12] ist ein Protokoll vorgesehen, mit dem ein CAN-Knoten den aktuellen Status eines beliebigen Knoten erfragen kann. Dies gehört zur Netzwerkmanagement-Funktionalität (NMT) des CAN-Knoten und wird als Node-Guarding bezeichnet. Die KEB-CANopen-Anschaltung unterstützt das Node-Guarding. Die Node-Guarding-Anforderung wird durch einen Remote-Frame auf dem Node-Guarding-Identifier abgesetzt. Die Antwort kommt dann als Datentelegramm mit 1 Byte Daten auf dem gleichen Identifier. Das Datenbyte enthält den Knoten-Status plus ein Toggelbit (MSBit), welches von Meldung zu Meldung invertiert wird. Jeder Knoten hat seinen speziellen Node-Guarding-Identifier. Beim Minimum Capability Device ergibt sich dieser Identifier direkt aus der Node-Id:

Node-Guarding-Identifizier = 1792 + Node-Id

Wert des Knoten-Status	Bedeutung
1	DISCONNECTED
2	CONNECTING
3	PREPARING
4	PREPARED
5	OPERATIONAL
127d	PRE_OPERATIONAL

3.7 Life-Guarding

Der F5 CANopen Operator unterstützt das Life-Guarding. Es handelt sich dabei um eine Überwachung des zyklischen Node-Guarding des CAN-Masters. Aus diesem Grund sollte das Life-Guarding nur beim zyklischen Node-Guarding aktiviert werden. Das Life-Guarding arbeitet vollständig losgelöst von allen anderen Überwachungsfunktionen. Es wird aktiviert über das Produkt der beiden Parameterwerte Guard Time und Life Time Factor. Ist das Produkt = 0, so ist das Life-Guarding nicht aktiv. Sonst gibt das Produkt die Life-Guarding-Timeoutzeit an. Bei aktiviertem Life-Guarding beginnt die Node-Guarding-Überwachung, sobald der erste Node-Guard-Request empfangen wurde. Die Funktion, die bei Eintreten des Life-Guarding-Timeoutfalls ausgeführt wird, ist über zwei weitere Parameter (LifeGuardTout.Addr, LifeGuardTout.Data) einstellbar. Es handelt sich dabei zum Einen um einen Schreibzugriff auf einen beliebigen Parameter in der Umrichtersteuerung und zudem um einen Funktionscode, der bestimmt, welche Aktion im Operator ausgeführt werden soll. Nach Auslieferung ist der CAN-Operator so eingestellt, dass bei auftretendem Life-Guarding-Timeout der Parameter Sy.50(Steuerwort) in Satz0 mit dem Wert 1 geschrieben wird. Zudem schaltet der CAN-Operator in den Zustand Pre_Operational.

3.8 Emergency Objekt

Das CANopen-Kommunikationsprofil DS301 definiert einen Mechanismus, nachdem sich Knoten selbständig melden, wenn wichtige Ereignisse eingetreten sind. Diese Emergency-Message unterstützt auch der KEB-F5-CANopen-Operator. Die Funktion ist in der Standardeinstellung deaktiviert. Aktiviert wird die Emergency-Message durch Verändern des Parameters EmergencyCycle auf einen Wert ungleich 0. Dann liest der CAN-Operator in dieser Zykluszeit den Wert des Parameters Umrichterstatus (ru.00) von der Umrichtersteuerung und konvertiert diesen in den ErrorCode-Wert nach [13]. Hat sich der Wert geändert, wird eine Emergency-Message auf Identifizier 128d + Node_Id abgeschickt. Das bedeutet, dass auch der Übergang vom Fehlerzustand zu normalen Betriebszuständen durch eine Emergency-Message bekannt gemacht wird. Der Inhalt des Telegramms ist vom Profil nur zum Teil fest vorgegeben. Insgesamt sieht der Inhalt der Emergency-Message beim KEB-F5-CAN wie folgt aus:

Identifizier = 128 + Node_Id

B0		B1	B2	B3		B4	B5	B6	B7
Error-Code		Error-Register	Umrichter Status		00h	00h	00h	00h	
LB	HB		LB	HB					

Alle Fehler werden in dem vom Profil definierten ‚Predefined ErrorField‘ gespeichert. Dieses Feld enthält beim KEB-F5-CANopen-Operator maximal fünf Einträge. Wobei der erste Eintrag immer den zeitlich zuletzt aufgetretenen Fehler enthält. Die Kodierung der Einträge sind der Beschreibung des gleichnamigen Parameters zu entnehmen.

4. Kodierung der Daten in den vier CAN-Telegramm-Typen

Über dieses Telegramm kann der logische CAN-Master den Wert eines Parameters Erfragen (Lesen) oder Verändern (Schreiben). Im Kommunikationsprofil wird ein Schreib-Dienst als Domain Download und ein Lese-Dienst als Domain Upload bezeichnet. Die KEB-CAN-Anschaltung unterstützt lediglich die Kurzform dieser beiden Dienste, so dass nur ein Telegramm für die Dienstaufforderung und ein weiteres für die Dienstbestätigung zwischen logischem CAN-Master und der KEB-CAN-Anschaltung ausgetauscht werden.

4.1 SDO(rx)-Telegramm

Die Adressierung des Parameters geschieht über den vorzeichenlosen 16-Bit-Index plus den vorzeichenlosen 8-Bit-Subindex. Die Parameter der Frequenzumrichtersteuerung liegen im Indexbereich 2000(hex) bis 5EFF(hex). Dabei ergibt sich der CAN-Index aus der Parameteradresse (siehe Parameterbeschreibung der eingesetzten FU-Steuerung) durch Addition mit dem Offset 2000(hex):

$\text{CAN-Index} = \text{KEB-Parameteradresse} + 2000 \text{ (hex)}$

Der Subindex dient als Zusatzadressierung für komplexe Parameter des Operators. Ebenso kann er bei Parametern der Frequenzumrichtersteuerung zur Satzadressierung verwendet werden. Dabei gilt:

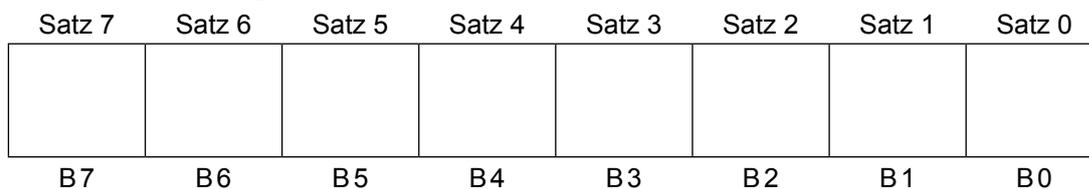
Subindex = 0

Bei satzprogrammierbaren Parametern bestimmt der Wert des Parameters FR.09 den gewählten Satz.

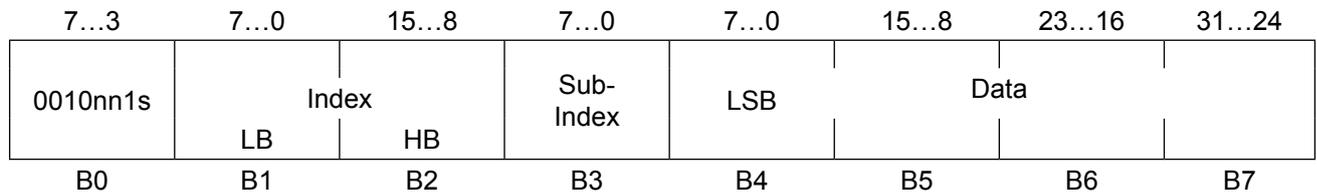
Subindex ungleich 0

Bei satzprogrammierbaren Parametern bestimmt der Subindex den gewählten Satz. Dabei ist zu beachten, dass der Satz bitkodiert ist. Dadurch ist es möglich, beim Schreiben den Wert des Parameters gleichzeitig in mehreren Sätzen zu verändern. Werden beim Lesen mehrere Sätze gleichzeitig adressiert, wird nur dann der Wert des Parameters zurückgegeben, wenn dieser in allen adressierten Sätzen gleich ist. Sind nicht alle Werte gleich, wird ein Fehler zurückgegeben.

Subindex (wenn ungleich 0):

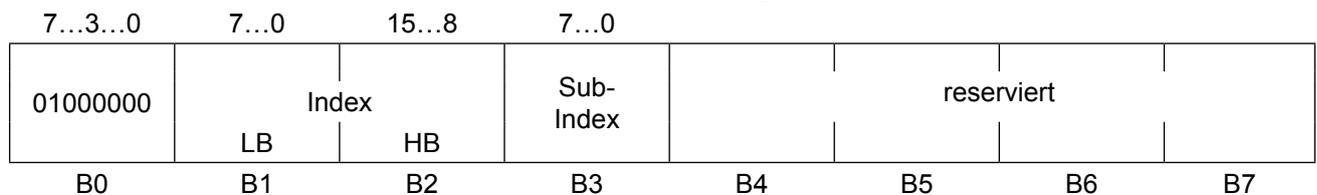


4.1.1 Initiate Domain Download Request (Schreibenanforderung des Master)



- nn: Nur gültig, wenn s = 1: Enthält die Anzahl Bytes des Data-Feldes, die keine Daten enthalten.
- s: Wenn gleich 1, dann enthält nn die Anzahl Bytes im Data-Feld, die keine Daten enthalten. Sonst keine Anzeige der Datenlänge in nn.
- Index: 16-Bit (vorzeichenlos) Adressierung des Parameters (s.o.).
- Subindex: 8-Bit (vorzeichenlos) Unteradressierung für komplexe Parameter und die direkte Satzadressierung.
- Data: Zu übertragende Daten. Das LS-Byte wird zuerst übertragen.

4.1.2 Initiate Domain Upload Request (Leseanforderung des Master)



- Index: 16-Bit (vorzeichenlos) Adressierung des Parameters (s.o.).
- Subindex: 8-Bit (vorzeichenlos) Unteradressierung für komplexe Parameter und die direkte Satzadressierung.

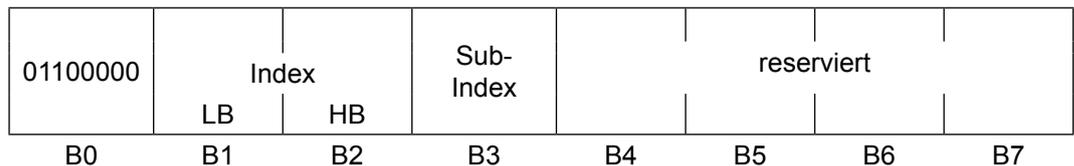
4.2 SDO(tx)-Telegramm

4.2.1 Initiate Domain Download Response (Schreibbestätigung vom FU)

Diese Antwort wird von der KEB-CAN-Anschaltung gesendet, wenn der angeforderte Schreibdienst fehlerfrei ausgeführt werden konnte.

Index: s.o. 7...3...0 7...0 15...8 7...0

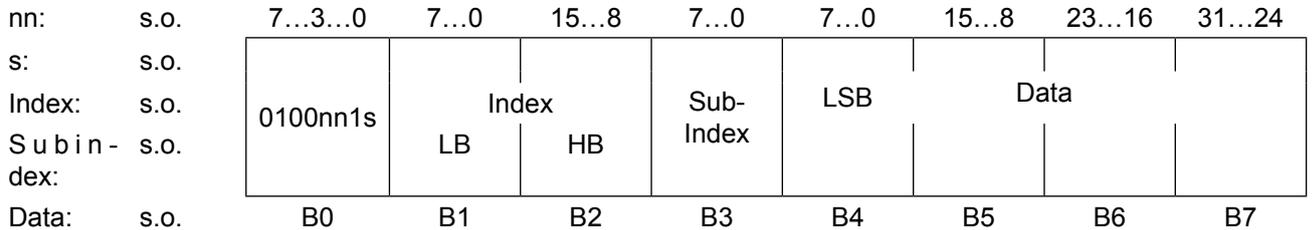
Subindex: s.o.



Kodierung der Daten in den vier CAN-Telegramm-Typen

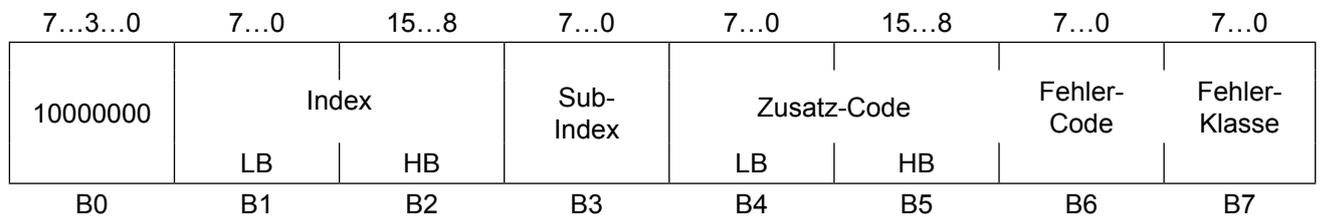
4.2.2 Initiate Domain Upload Response (Lesebestätigung vom FU)

Diese Antwort wird von der KEB-CAN-Anschaltung gesendet, wenn der angeforderte Lesedienst fehlerfrei ausgeführt werden konnte.



4.2.3 Abort Domain Transfer (Fehlerantwort vom FU)

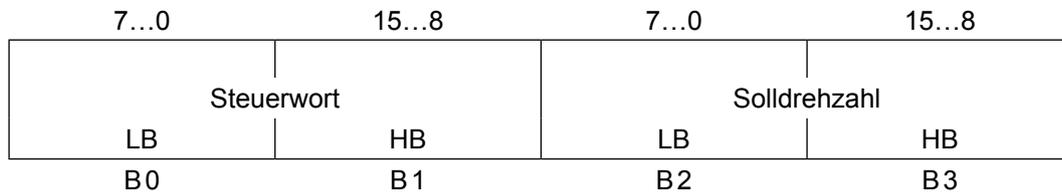
Diese Antwort sendet die KEB-CAN-Anschaltung, wenn der angeforderte Schreib- oder Lesedienst nicht ausgeführt werden konnte. In diesem Fall wird eine Fehlerbeschreibung zurückgeliefert.



Fehler-Klasse	Fehler-Code	Zusatz-Code	Bedeutung
6	1	0000h	Ungültiger Zugriff auf einen Parameter, z.B. Schreiben auf einen Read_Only-Parameter.
6	1	0010h	Ungültiges Passwort
6	1	0011h	Operation nicht möglich
6	4	0000h	Der adressierte Parameter existiert nicht.
6	4	0041h	Ungültige PD-Belegung
6	6	0000h	Die interne Kommunikation zwischen Operator und FU-Steuerung ist gestört.
6	7	0010h	Datenlänge ungültig
6	9	0011h	Ungültiger Subindex
6	9	0012h	Sprachkennung ungültig
6	9	0030h	Der geschriebene Wert liegt außerhalb des gültigen Wertebereichs.
8	0	0022h	Umrichter beschäftigt

4.3 PDO1(rx)-Telegramm

Mit diesem Telegramm übergibt der logische CAN-Master dem Frequenzumrichter neue Prozessausgangsdaten. In der Standardeinstellung erwartet die KEB-CAN-Anschaltung ein Telegramm mit > 4 Byte Daten mit folgendem Inhalt:



Die Länge und Belegung des PDO1(rx)-Telegramms kann durch verschiedene Operatorparameter verändert werden. Diese Veränderung kann nur über das SDO(tx)-Telegramm erfolgen (s.o.).

Folgende Operatorparameter haben Einfluss auf den Aufbau der Prozessausgangsdaten:

- 1st receive PDO Mapping
- 1st receive PDO Parameter

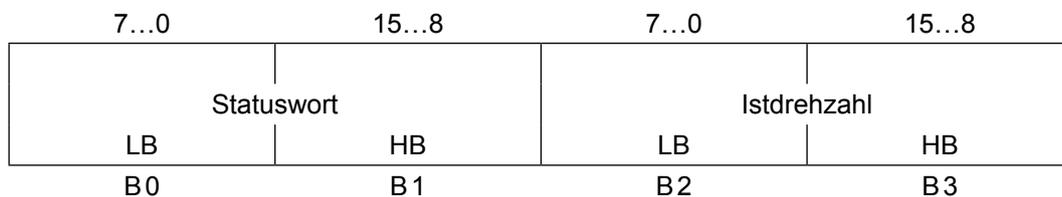
4.4 PDO1(tx)-Telegramm

Über dieses Telegramm gibt die KEB-CAN-Anschaltung dem (logischen) CAN-Master Prozesseingangsdaten bekannt.

Die Länge, Belegung und Steuerung dieses Telegramms wird durch folgende Operatorparameter beeinflusst:

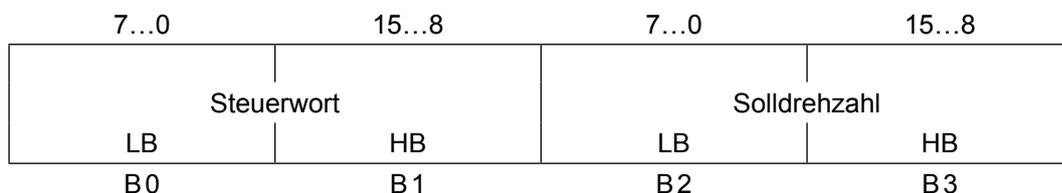
- 1st transmit PDO Mapping
- 1st transmit PDO Parameter

Die Standardeinstellung bewirkt folgenden Telegrammaufbau:



4.5 PDO2(rx)-Telegramm

Mit diesem Telegramm übergibt der logische CAN-Master dem Frequenzumrichter neue Prozessausgangsdaten. In der Standardeinstellung erwartet die KEB-CAN-Anschaltung ein Telegramm mit > 4 Byte Daten mit folgendem Inhalt:



Die Länge und Belegung des PDO2(rx)-Telegramms kann durch verschiedene Operatorparameter verändert werden. Diese Veränderung kann nur über das SDO(tx)-Telegramm erfolgen (s.o.).

Folgende Operatorparameter haben Einfluss auf den Aufbau der Prozessausgangsdaten:

- 2nd receive PDO Mapping
- 2nd receive PDO Parameter

Kodierung der Daten in den vier CAN-Telegramm-Typen

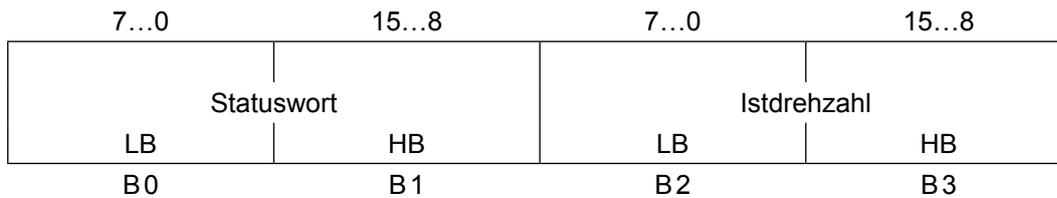
4.6 PDO2(tx)-Telegramm

Über dieses Telegramm gibt die KEB-CAN-Anschaltung dem (logischen) CAN-Master Prozesseingangsdaten bekannt.

Die Länge, Belegung und Steuerung dieses Telegramms wird durch folgende Operatorparameter beeinflusst:

- 2nd transmit PDO Mapping
- 2nd transmit PDO Parameter

Die Standardeinstellung bewirkt folgenden Telegrammaufbau:



5. Operatorparameter

5.1 Von KEB definierte Parameter

Diese Parameter bestimmen die Konfiguration der KEB F5-CAN-Anschaltung und sind deshalb auch in dieser und nicht in der Frequenzumrichtersteuerung realisiert:

PD_Stored

CAN-SDO-Index	5FE2h
Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	1 Byte
Zugriff	READ_WRITE
Bedeutung	Bestimmt, ob die aktuelle Prozessdatenbelegung aus dem EEPROM gelesen wird oder mit der Standard-PD-Belegung gearbeitet wird.
Kodierung	FFh → arbeitet mit der abgespeicherten PD-Belegung sonst → arbeitet mit der Standard-PD-Belegung.
Standardeinstellung	FFh
Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert.

OP_Nodeld

CAN-SDO-Index	5FE3h
Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	1 Byte
Zugriff	READ_WRITE
Bedeutung	Ermöglicht die Vorgabe der CANopen-Knotenadresse im CAN-Operator, unabhängig von der Umrichteradresse.
Kodierung	255 : Die Knotenadresse wird wie bisher aus der Umrichteradresse (Sy.06) bestimmt: Node_Id = Umrichteradresse + 1 1...127 : Die Knotenadresse wird im Operator gehalten und gespeichert: Node_Id = OP_Nodeld
Standardeinstellung	255
Bemerkung	Eine Wertänderung wird sofort aktiv und nichtflüchtig gespeichert.

Operatorparameter

Watchdog activation

CAN-SDO-Index	5FDAh
SDO-Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	1 Byte
Zugriff	READ_WRITE
Bedeutung	Ermöglicht die Verzögerung der Aktivierung des Feldbus-Watchdog nach POWER On bzw. einem Reset-Kommando über CAN.
Kodierung	<p>0 : Der Feldbus-Watchdog ist sofort aktiv.</p> <p>Werte ungleich Null sind bitkodiert mit der folgenden Bedeutung:</p> <p>Bit0 : Aktivierung des Feldbus-Watchdog nach erstem SYNC-Telegramm</p> <p>Bit1 : Aktivierung des Feldbus-Watchdog nach erstem Node-Guarding</p> <p>Bit2 : Aktivierung des Feldbus-Watchdog nach erstem Übergang in den Knotenstatus OPERATIONAL</p> <p>Bit3 : Aktivierung des Feldbus-Watchdog nach erstem PDOOUT1-Telegramm</p> <p>Bit4 : Aktivierung des Feldbus-Watchdog nach erstem PDOOUT2-Telegramm</p> <p>Bit5 : Aktivierung des Feldbus-Watchdog nach erstem SDO-Telegramm</p>
Standardeinstellung	0
Erlaubte PDO-Abbildung	keine Abbildung
Bemerkung	Eine Wertänderung wird sofort aktiv und nichtflüchtig gespeichert. Mehrere Ereignisse können als Feldbus-Watchdog-Aktivierung definiert werden. In diesem Fall wird der Watchdog aktiv, sobald eines der definierten Ereignisse eintritt.

B7	B6	B5	B4	B3	B2	B1	B0
		1. SDO	1. PDOOUT2	1. PDOOUT1	1. OPERATIONAL	1. Node-Guarding	1. SYNC

**Watchdog
inhibit**

CAN-SDO-Index	5FF9h
Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	1 Byte
Zugriff	READ_WRITE
Bedeutung	Bestimmt, auf welche Ereignisse der Feldbus-Watchdog getriggert wird. Der Feldbus-Watchdog dient dazu, den Frequenzumrichter in den Fehlerzustand zu bringen, wenn auf CAN keine Aktivitäten mehr stattfinden. Die eigentliche Aktivierung und Programmierung des Watchdog wird in der FU-Steuerung eingestellt. Die dazu einzustellenden Parameter sind der Anleitung der FU-Steuerung zu entnehmen.
Kodierung	<p>Bitkodiert:</p> <p>Bit0 = 1 Beim Starten eines PDOOUT-Telegramms zur FU-Steuerung wird der Watchdog zurückgesetzt. Beachten Sie, dass das Eintreten dieses Ereignisses auch von der Einstellung des Parameters 1st Receive PDO Parameter. Tx_type sowie von dem Wert des Parameters PDOOUT_Wr_Mode abhängt.</p> <p>Bit1 = 1 Beim Beginn der Bearbeitung eines SDO-Auftrages wird der Watchdog zurückgesetzt.</p> <p>Bit2 = 1 Wenn der Knoten keine Übertragungsprobleme auf CAN feststellt, wird der Watchdog zurückgesetzt.</p> <p>Bit3 = 1 Der Watchdog wird bei jedem Empfang eines SYNC-Telegramms zurückgesetzt.</p> <p>Bit4 = 1 Der Watchdog wird bei jedem Empfang eines Node-Guard-Request-Telegramms zurückgesetzt.</p> <p>Bit5 = 1 Der Watchdog wird bei jedem Empfang eines SYNC-Telegramms zurückgesetzt, vorausgesetzt, mindestens einmal wurden Prozessausgangsdaten zur Frequenzumrichtersteuerung gesendet.</p>
Standardeinstellung	<p>07h</p> <p>Der Watchdog wird zurückgesetzt, wenn:</p> <ul style="list-style-type: none"> - Prozessausgangsdaten zur FU-Steuerung geschrieben werden, - ein SDO-Auftrag gestartet wird und - keine Übertragungsprobleme auf CAN festgestellt werden.
Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert.

Operatorparameter

PDOOUT_WrMode

CAN-SDO-Index	5FE4h
Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	1 Byte
Zugriff	READ_WRITE
Bedeutung	Bestimmt unter welchen Bedingungen PDOOUT-Daten zur FU-Steuerung geschrieben werden. Hiermit kann die Kommunikation zwischen CAN-Operator und FU-Steuerung entlastet werden.
Kodierung	0 : Es werden immer alle PDOOUT-Daten zur FU-Steuerung geschrieben, egal ob diese geändert sind oder nicht. 255 : Es werden immer alle PDOOUT-Daten zum Umrichter geschrieben, wenn mindestens einer der Werte geändert wurde. sonst : Es werden nur die geänderten Werte geschrieben.
Standardeinstellung	255
Bemerkung	Eine Wertänderung wird sofort aktiv und nichtflüchtig gespeichert.

HS_PDO_Index

CAN-SDO-Index	5FE5h
Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	1 Byte
Zugriff	READ_WRITE
Bedeutung	Mit diesem Parameter wird bestimmt, welches PDO das High-Speed PDO sein soll.
Kodierung	0 : 1. PDO ist High-Speed-PDO 1 : 2. PDO ist High-Speed-PDO
Standardeinstellung	0
Bemerkung	Ein geänderter Wert wird nichtflüchtig gespeichert, aber erst beim nächsten Einschalten oder Reset-Node-Kommando aktiv. Zudem werden bei Wertänderung alle PDO's deaktiviert (Bit 31 von PDO Parameter CobID = 1).

PDIN1_Cycle_Time

CAN-SDO-Index	5FE6h
Subindex	0
Objektyp	Einfach Variable (Var)
Datenlänge	2 Byte
Zugriff	READ_WRITE
Bedeutung	Gibt die Zykluszeit an, in der die Prozesseingangsdaten des PDO1 im Zustand OPERATIONAL von der FU-Steuerung gelesen werden.
Kodierung	1 ms
Standardeinstellung	25 = 25 ms
Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert.

PDIN2_Cycle_Time

CAN-SDO-Index	5FE7h
Subindex	0
Objektyp	Einfach Variable (Var)
Datenlänge	2 Byte
Zugriff	READ_WRITE
Bedeutung	Gibt die Zykluszeit an, in der die Prozesseingangsdaten des PDO2 im Zustand OPERATIONAL von der FU-Steuerung gelesen werden.
Kodierung	1 ms
Standardeinstellung	100 = 100 ms
Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert.

SAVE_CAN_Baud

CAN-SDO-Index	5FFEh
Subindex	0
Objektyp	Einfach Variable (Var)
Datenlänge	1 Byte
Zugriff	READ_WRITE
Bedeutung	Dient zum nichtflüchtigen Abspeichern der eingestellten CAN-Übertragungsgeschwindigkeit.
Kodierung	Schreiben: FFh : nichtflüchtiges Abspeichern von CAN_Baud 0 : kein Abspeichern Lesen: FFh : eingestellter Wert stimmt mit abgespeichertem überein 00h : eingestellter Wert ungleich abgespeicherten Wert

Operatorparameter

CAN_Baud

CAN-SDO-Index	5FFFh	
Subindex	0	
Objekttyp	Einfach Variable (Var)	
Datenlänge	1 Byte	
Zugriff	READ_WRITE	
Bedeutung	Index für CAN-Übertragungsgeschwindigkeit	
Kodierung	0 = 10 Kbit/s 1 = 20 Kbit/s 2 = 50 Kbit/s 3 = 100 Kbit/s 4 = 125 Kbit/s	5 = 250 Kbit/s 6 = 500 Kbit/s* 7 = 1000 Kbit/s* 8 = 800 Kbit/s* 9 = 25 Kbit/s
Standardeinstellung	1	
Bemerkung	Ein geänderter Wert wird sofort aktiv aber nicht automatisch nichtflüchtig gespeichert. Das Bit-Timing hält sich an die Vorgaben des Arbeitskreises Physical-Layer der CiA [2]. Zum Bit-Timing siehe Anhang. Welche Übertragungsgeschwindigkeiten gefahren werden können, hängt von der Leitungslänge, der Summe der Verzögerungszeiten und dem Bit-Timing ab und muss im Einzelfall geklärt werden.	

CAN_Baud2

CAN-SDO-Index	5FECh	
Subindex	0	
Objekttyp	Einfach Variable (Var)	
Datenlänge	1 Byte	
Zugriff	READ_WRITE	
Bedeutung	Index für CAN-Übertragungsgeschwindigkeit alternativ zu CAN_Baud (s.o.)	
Kodierung	0 = 10 Kbit/s 1 = 20 Kbit/s 2 = 50 Kbit/s 3 = 100 Kbit/s 4 = 125 Kbit/s	5 = 250 Kbit/s 6 = 500 Kbit/s* 7 = 1000 Kbit/s* 8 = 800 Kbit/s* 9 = 25 Kbit/s
Standardeinstellung	1	
Bemerkung	Ein geänderter Wert wird im Gegensatz zum Parameter CAN_Baud sofort nichtflüchtig gespeichert, aber erst nach einem Reset-Node-Kommando oder nach dem nächsten Einschalten aktiv.	

* Beachte Kapitel "Wichtiger Warnhinweis"

5.2 Vom Kommunikationsprofil [12] definierte Parameter

Device type	(nach CANopen [13])	
	CAN-SDO-Index	1000h
	Subindex	0
	Objekttyp	Einfach Variable (Var)
	Datenlänge	4 Byte
	Zugriff	READ_ONLY
	Bedeutung	Beschreibt den Gerätetyp nach CANopen-Kommunikationsprofil.
	Kodierung	Bisher keine Festlegungen
	Standardeinstellung	0
	Bemerkung	Dieser Parameter ist konstant und kann deshalb nur gelesen werden.

Error register	(nach CANopen [13])	
	CAN-SDO-Index	1001h
	Subindex	0
	Objekttyp	Einfach Variable (Var)
	Datenlänge	1 Byte
	Zugriff	READ_ONLY
	Bedeutung	Zeigt den Fehlerstatus des CANopen-Teilnehmers an.
	Kodierung	Bit0 = 1 → Fehler liegt vor
	Standardeinstellung	0
	Bemerkung	Dieser Parameter kann nur gelesen werden. Als Quelle für diesen Parameter dient der Umrichterparameterstatus (ru.00). Die Umsetzung der ru.00-Werte in Werte des Error registers ist der Tabelle im Anhang zu entnehmen. Beachten Sie, dass der Wert des Error register nur bei aktivierter Emergencybearbeitung aktualisiert wird (→ EmergencyCycle).

Operatorparameter

Manufacturer Status Register

CAN-SDO-Index	1002h			
Subindex	0			
Objektyp	Einfach Variable (Var)			
Datenlänge	4 Byte			
Zugriff	READ_ONLY			
Bedeutung	Direkte Abbildung des Parameters Umrichterstatus (ru.00) im DS301-Parameterbereich.			
Kodierung	Siehe Beschreibung des Parameters Umrichterstatus (ru.00) in der Bedienungsanleitung der Umrichtersteuerung.			
Standardeinstellung	0			
Erlaubte PDO-Abbildung	High-Speed-PDO		Low-Speed-PDO	
	rx	tx	rx	tx
	NEIN	JA	NEIN	JA
Bemerkung	Wird intern auf den Parameter ru.00 abgebildet.			

Manufacturer Device Name

CAN-SDO-Index	1008h			
Subindex	0			
Objektyp	Einfach Variable (Var)			
Datenlänge	4 Byte			
Zugriff	READ_ONLY			
Bedeutung	Gibt den Wert des Parameters Umrichter_Identifikation (Sy.02) der FU-Steuerung als 4-Zeichen hexadezimal-String aus.			
Kodierung	Der Wert 1234h würde wie folgt im SDO-Response-Telegramm übertragen:			
	B4	B5	B6	B7
	31h	32h	33h	34h
Standardeinstellung	Je nach Umrichertyp			
Erlaubte PDO-Abbildung	Keine Abbildung			

Identify object

CAN-SDO-Index	1018h
Subindex	0
Objekttyp	Strukturierte Variable (Record)
Datenlänge	1 Byte
Zugriff	READ_ONLY
Bedeutung	Gibt die Anzahl der Einträge in diesem Objekt an.
Kodierung	1
Standardeinstellung	2
Bemerkung	Der Wert dieses Parameters kann nur gelesen werden.

Subindex	1
Datenlänge	4 Byte
Bedeutung	Herstellerbezeichnung von der CAN in Automation Nutzergruppe vergeben.
Kodierung	Bit31...Bit24 : Abteilung Bit23...Bit0 : Firma
Standardeinstellung	00000014h
Bemerkung	Der Wert dieses Parameters kann nur gelesen werden

Subindex	2
Datenlänge	4 Byte
Bedeutung	Produktbezeichnung
Kodierung	00000004h = Typ F4 00000005h = Typ F5
Standardeinstellung	00000005h
Bemerkung	Der Wert dieses Parameters kann nur gelesen werden.

Manufacturer software version

CAN-SDO-Index	100Ah								
SDO-Subindex	0								
Objekttyp	Einfach Variable (Var)								
Datenlänge	4 Byte								
Zugriff	READ_ONLY								
Bedeutung	Gibt den Wert des Parameters Softwareversion (In.06) der FU-Steuerung als 4-Zeichen hexadezimal-String aus.								
Kodierung	Der Wert 140h / 260d (= Version 2.60) würde wie folgt im SDO-Response-Telegramm übertragen: <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">B4</td> <td style="text-align: center;">B5</td> <td style="text-align: center;">B6</td> <td style="text-align: center;">B7</td> </tr> <tr> <td style="text-align: center;">30h</td> <td style="text-align: center;">31h</td> <td style="text-align: center;">30h</td> <td style="text-align: center;">34h</td> </tr> </table>	B4	B5	B6	B7	30h	31h	30h	34h
B4	B5	B6	B7						
30h	31h	30h	34h						
Standardeinstellung	Je nach Softwareversion der Umrückersteuerung								
Erlaubte PDO-Abbildung	Keine Abbildung								

Operatorparameter

1st receive PDO Parameter

CAN-SDO-Index	1400h
Subindex	0
Objektyp	Strukturierte Variable (Record)
Datenlänge	1 Byte
Zugriff	READ_WRITE
Bedeutung	Gibt die Anzahl der Einträge an, die unter diesem Objekt angesprochen werden können.
Kodierung	1
Standardeinstellung	2
Bemerkung	Der Wert dieses Parameters kann nur gelesen werden.

Subindex	1
Datenlänge	4 Byte
Bedeutung	Gibt an, auf welchem Identifier das PDO(rx) für den Transfer der Prozessausgangsdaten gesendet wird. Zusätzlich sind noch Steuerinformationen für dieses PDO in den obersten Bits enthalten.
Kodierung	<p>Bit31(MSB) = 0 → Die Bearbeitung der Prozessausgangsdaten ist aktiviert.</p> <p>Bit31(MSB) = 1 → Die Bearbeitung der Prozessausgangsdaten ist ausgeschaltet.</p> <p>Bit30 = 0 → Remote Frame auf dem entspr. Identifier wird beantwortet.</p> <p>Bit30 = 1 → Remote Frame wird nicht beantwortet.</p> <p>Bit29 = 0 → 11-Bit Identifier (CAN V2.0A)</p> <p>Bit29 = 1 → 29-Bit Identifier (CAN V2.0B), hier nicht einstellbar. Es werden aber 29-Bit-Identifier-Telegramme empfangen und bearbeitet.</p> <p>Bit28...Bit0 = Identifier (Bit0 = LSB), hier für Bit28 bis Bit 11=fest=0.</p>
Standardeinstellung	00000200h + Node_Id
weiter auf nächster Seite	

Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert. Beim Einschalten der Prozessdatenbearbeitung (Bit31 von "1" auf "0") wird die Einstellung des Parameters 1st receive PDO Mapping (Index = 1600h) zur Umrichtersteuerung transferiert. Sollte die FU-Steuerung die Abbildung nicht akzeptieren, wird an dieser Stelle eine Fehlerantwort zurückgegeben und die Prozessausgangsdatenbearbeitung bleibt ausgeschaltet. Wird die PD-Abbildung vom FU akzeptiert, wird diese automatisch nichtflüchtig gespeichert und die Prozessausgangsdatenbearbeitung wie gewünscht eingeschaltet. Da die Identifizierung der PDOs direkt von der Node_Id abgeleitet wird, können die Bits Bit28 bis Bit0 nur gelesen werden. Beim Schreiben werden diese Bits ignoriert.
-----------	--

Subindex	2
Datenlänge	1 Byte
Bedeutung	Bestimmt, wann und wie dieses Objekt auf dem CAN-Bus gesendet wird.
Kodierung	0 ... 240: Bei Empfang eines SYNC-Kommandos (Identifier = 128d, Datenlänge = 0) werden die aktuellen Prozessausgangsdaten zur FU-Steuerung transferiert. 254 (asynchron, herstellerepezifisch): Die Prozessausgangsdaten werden zur FU-Steuerung transferiert, sobald sich mindestens ein Byte geändert hat. 255 (asynchron, profilspezifisch): Siehe asynchron, herstellerepezifisch.
Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert. Beachten Sie auch den Einfluss des Parameters PDOOUT_WrMode.

Operatorparameter

2nd receive PDO Parameter

CAN-SDO-Index	1401h
Subindex	0
Objektyp	Strukturierte Variable (Record)
Datenlänge	1 Byte
Zugriff	READ_WRITE
Bedeutung	Gibt die Anzahl der Einträge an, die unter diesem Objekt angesprochen werden können.
Kodierung	1
Standardeinstellung	2
Bemerkung	Der Wert dieses Parameters kann nur gelesen werden.

Subindex	1
Datenlänge	4 Byte
Bedeutung	Gibt an, auf welchem Identifier das PDO(rx) für den Transfer der Prozessausgangsdaten gesendet wird. Zusätzlich sind noch Steuerinformationen für dieses PDO in den obersten Bits enthalten.
Kodierung	<p>Bit31(MSB) = 0 → Die Bearbeitung der Prozessausgangsdaten ist aktiviert.</p> <p>Bit31(MSB) = 1 → Die Bearbeitung der Prozessausgangsdaten ist ausgeschaltet.</p> <p>Bit30 = 0 → Remote Frame auf dem entspr. Identifier wird beantwortet.</p> <p>Bit30 = 1 → Remote Frame wird nicht beantwortet.</p> <p>Bit29 = 0 → 11-Bit Identifier (CAN V2.0A)</p> <p>Bit29 = 1 → 29-Bit Identifier (CAN V2.0B), hier nicht einstellbar. Es werden aber 29-Bit-Identifier-Telegramme empfangen und bearbeitet.</p> <p>Bit28...Bit0 = Identifier (Bit0 = LSB), hier für Bit28 bis Bit 11=fest=0.</p>
Standardeinstellung	80000300h + Node_Id
Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert. Beim Einschalten der Prozessdatenbearbeitung (Bit31 von "1" auf "0") wird die Einstellung des 2nd receive PDO Mapping auf ein entsprechendes Umrichter Mapping umgesetzt. Konnte dies erfolgreich durchgeführt werden, wird das Mapping automatisch nichtflüchtig abgespeichert. Da die Identifiervergabe der PDOs direkt von der Node_Id abgeleitet wird, können die Bits Bit28 bis Bit0 nur gelesen werden. Beim Schreiben werden diese Bits ignoriert.
weiter auf nächster Seite	

Subindex	2
Datenlänge	1 Byte
Bedeutung	Bestimmt, wann und wie dieses Objekt auf dem CAN-Bus gesendet wird.
Kodierung	0 ... 240: Bei Empfang eines SYNC-Kommandos (Identifier = 128d, Datenlänge = 0) werden die aktuellen Prozessausgangsdaten zur FU-Steuerung transferiert. 254 (asynchron, herstellerspezifisch): Die Prozessausgangsdaten werden zur FU-Steuerung transferiert, sobald sich mindestens ein Byte geändert hat. 255 (asynchron, profilspezifisch): Siehe asynchron, herstellerspezifisch.
Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert. Beachten Sie auch den Einfluss des Parameters PDOOUT_WrMode.

1st receive PDO Mapping

CAN-SDO-Index	1600h
Subindex	0
Objekttyp	Strukturierte Variable (Record)
Datenlänge	1 Byte
Zugriff	READ_WRITE
Bedeutung	Gibt die Anzahl der Einträge an, die unter diesem Objekt angesprochen werden können.
Kodierung	1 (Maximal gültiger Wertebereich 1.....4).
Standardeinstellung	2
Bemerkung	Ein Schreiben dieses Parameters bedingt das automatische Abschalten der Prozessausgangsdatenbearbeitung (Bit31 von Index 1400h, Subindex = 1 wird auf "1" gesetzt).

Subindex	1 bis maximal 4												
Datenlänge	4 Byte												
Bedeutung	Bezeichnet eine Objektabbildung. Es wird der Index, Subindex und die Objektlänge in Bits angegeben.												
Kodierung	<table border="1" style="width: 100%; text-align: center;"> <tr> <td colspan="2">Index</td> <td>Sub-Index</td> <td>Objektlänge</td> </tr> <tr> <td>HB</td> <td>LB</td> <td></td> <td></td> </tr> <tr> <td>B3</td> <td>B2</td> <td>B1</td> <td>B0</td> </tr> </table>	Index		Sub-Index	Objektlänge	HB	LB			B3	B2	B1	B0
Index		Sub-Index	Objektlänge										
HB	LB												
B3	B2	B1	B0										
Standardeinstellung	s.u.												

weiter auf nächster Seite

Operatorparameter

Bemerkung	<p>Ein Schreiben dieses Parameters bedingt das automatische Abschalten der Prozessausgangsdatenbearbeitung (Bit31 von Index 1400h, Subindex = 1 wird auf "1" gesetzt).</p> <p>Der Zusammenhang zwischen Prozessausgangsdatenabbildung und dem entsprechenden PDO1(rx)-Telegrammaufbau soll an der Standardbelegung nochmals zusammengefasst werden:</p> <div style="text-align: right; margin-bottom: 10px;"> <p>PDO1(rx)-Telegramm:</p> <table style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="padding: 0 10px;">7...0</td> <td style="padding: 0 10px;">15...8</td> <td style="padding: 0 10px;">7...0</td> <td style="padding: 0 10px;">15...8</td> </tr> <tr> <td colspan="2" style="border: 1px solid black; text-align: center;">Steuerwort</td> <td colspan="2" style="border: 1px solid black; text-align: center;">Solldrehzahl</td> </tr> <tr> <td style="border: 1px solid black; text-align: center;">LB</td> <td style="border: 1px solid black; text-align: center;">HB</td> <td style="border: 1px solid black; text-align: center;">LB</td> <td style="border: 1px solid black; text-align: center;">HB</td> </tr> </table> </div> <div style="margin-bottom: 10px;"> <table style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="padding-right: 10px;">Sub-Index</td> <td style="padding-right: 10px;">1st receive PDO Mapping</td> <td style="border: 1px solid black; width: 20px; height: 20px;"></td> <td style="border: 1px solid black; width: 20px; height: 20px;"></td> <td style="border: 1px solid black; width: 20px; height: 20px;"></td> <td style="border: 1px solid black; width: 20px; height: 20px;"></td> </tr> <tr> <td style="padding-right: 10px;">0</td> <td style="padding-right: 10px;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black; text-align: center;">02h</td> <td style="border: 1px solid black;"></td> </tr> <tr> <td style="padding-right: 10px;">1</td> <td style="padding-right: 10px;">20h 32h 01h 10h</td> <td style="border: 1px solid black;"></td> </tr> <tr> <td style="padding-right: 10px;">2</td> <td style="padding-right: 10px;">20h 34h 01h 10h</td> <td style="border: 1px solid black;"></td> </tr> </table> </div> <div> <p>Beispiel für die Kodierung:</p> <p>Das erste abgebildete Objekt im receive PDO soll nicht das Steuerwort, sondern der Parameter mit dem Index 2302h und Subindex = 1 sein. In diesem Fall sind die 8 Bytes des Initiate-Domain-Download-Requests wie folgt zu füllen (Beispiel der Daten auf dem CAN-BUS):</p> <table style="margin-left: auto; margin-right: auto; border-collapse: collapse; text-align: center;"> <tr> <td style="border: 1px solid black; padding: 2px 5px;">23h</td> <td style="border: 1px solid black; padding: 2px 5px;">00h</td> <td style="border: 1px solid black; padding: 2px 5px;">18h</td> <td style="border: 1px solid black; padding: 2px 5px;">01h</td> <td style="border: 1px solid black; padding: 2px 5px;">10h</td> <td style="border: 1px solid black; padding: 2px 5px;">01h</td> <td style="border: 1px solid black; padding: 2px 5px;">02h</td> <td style="border: 1px solid black; padding: 2px 5px;">23h</td> </tr> <tr> <td style="padding: 0 5px;">B0</td> <td style="padding: 0 5px;">B1</td> <td style="padding: 0 5px;">B2</td> <td style="padding: 0 5px;">B3</td> <td style="padding: 0 5px;">B4</td> <td style="padding: 0 5px;">B5</td> <td style="padding: 0 5px;">B6</td> <td style="padding: 0 5px;">B7</td> </tr> </table> </div>	7...0	15...8	7...0	15...8	Steuerwort		Solldrehzahl		LB	HB	LB	HB	Sub-Index	1st receive PDO Mapping					0				02h		1	20h 32h 01h 10h					2	20h 34h 01h 10h					23h	00h	18h	01h	10h	01h	02h	23h	B0	B1	B2	B3	B4	B5	B6	B7
7...0	15...8	7...0	15...8																																																		
Steuerwort		Solldrehzahl																																																			
LB	HB	LB	HB																																																		
Sub-Index	1st receive PDO Mapping																																																				
0				02h																																																	
1	20h 32h 01h 10h																																																				
2	20h 34h 01h 10h																																																				
23h	00h	18h	01h	10h	01h	02h	23h																																														
B0	B1	B2	B3	B4	B5	B6	B7																																														

Operatorparameter

1st transmit PDO Parameter

CAN-SDO-Index	1800h
Subindex	0
Objektyp	Strukturierte Variable (Record)
Datenlänge	1 Byte
Zugriff	READ_WRITE
Bedeutung	Gibt die Anzahl der Einträge in diesem Objekt an.
Kodierung	1
Standardeinstellung	3
Bemerkung	Der Wert dieses Parameters kann nur gelesen werden.

Subindex	1
Datenlänge	4 Byte
Bedeutung	Gibt an, auf welchem Identifier das PDO(tx) für den Transfer der Prozesseingangsdaten gesendet wird. Zusätzlich sind noch einige Zusatzinformationen in den obersten Bits enthalten.
Kodierung	<p>Bit31(MSB) = 0 → Die Bearbeitung der Prozesseingangsdaten ist aktiv.</p> <p>Bit31(MSB) = 1 → Die Bearbeitung der Prozesseingangsdaten ist ausgeschaltet.</p> <p>Bit30 = 0 → Remote Frame auf dem entspr. Identifier wird beantwortet.</p> <p>Bit30 = 1 → Remote Frame wird nicht beantwortet.</p> <p>Bit29 = 0 → 11-Bit Identifier (CAN V2.0A)</p> <p>Bit29 = 1 → 29-Bit Identifier (CAN V2.0B), hier nicht einstellbar.</p> <p>Bit28...Bit0 = Identifier (Bit0 = LSB), hier für Bit28 bis Bit11=fest=0.</p>
Standardeinstellung	00000180h + Node_Id
Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert. Beim Einschalten der Prozessdatenbearbeitung (Bit31 von "1" auf "0") wird die Einstellung des Parameters 1st transmit PDO Mapping (Index 1A00h) zur Umrichtersteuerung transferiert. Sollte die FU-Steuerung die Abbildung nicht akzeptieren, wird an dieser Stelle eine Fehlerantwort zurückgegeben und die Prozesseingangsdatenbearbeitung bleibt ausgeschaltet. Wird die PDO-Abbildung vom FU akzeptiert, wird diese automatisch nichtflüchtig gespeichert und die Prozesseingangsdatenbearbeitung wie gewünscht eingeschaltet. Da die Identifiervergabe der PDOs direkt von der Node_Id abgeleitet wird, können die Bits Bit28 bis Bit0 nur gelesen werden. Beim Schreiben werden diese Bits ignoriert.
weiter auf nächster Seite	

Subindex	2
Datenlänge	1 Byte
Bedeutung	Bestimmt, wann und wie dieses Objekt auf dem CAN-Bus gesendet wird.
Kodierung	<p>0 (synchron azyklisch): Bei jedem Empfang eines SYNC wird ein PDO(tx)-Telegramm auf CAN gesendet.</p> <p>1 - 240 (synchron, zyklisch): In diesem Einstellbereich wird über den Wert eingestellt, wie viele SYNC-Telegramme empfangen werden müssen, bevor ein PDO(tx)-Telegramm auf CAN gesendet wird.</p> <p>252 (synchron-RTROnly): Ein PDO(tx)-Telegramm wird nur nach einem Remote-Request auf dem PDO(tx)-Identifizier gesendet.</p> <p>253 (asynchron-RTROnly): Ein PDO(tx)-Telegramm wird nur nach einem Remote-Request auf dem PDO(tx)-Identifizier gesendet.</p> <p>254 (asynchron, herstellerspezifisch): Ein PDO(tx)-Telegramm wird gesendet, sobald sich mindestens ein Byte geändert hat.</p> <p>255 (asynchron, profilspezifisch): Ein PDO(tx)-Telegramm wird gesendet, sobald sich mindestens ein Byte geändert hat.</p>
Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert.

Subindex	3
Datenlänge	2 Byte
Bedeutung	Bezeichnet den minimalen zeitlichen Abstand zwischen zwei CAN-Telegrammen auf diesem Identifizier.
Kodierung	100 μ s
Standard Einstellung	150 (= 15 ms)
Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert. Die interne Auflösung für die Inhibittime beträgt 1 ms. Somit hat der eingestellte Wert eine Ungenauigkeit von +/- 1 ms.

Operatorparameter

2nd transmit PDO Parameter

CAN-SDO-Index	1801h
Subindex	0
Objekttyp	Strukturierte Variable (Record)
Datenlänge	1 Byte
Zugriff	READ_WRITE
Bedeutung	Gibt die Anzahl der Einträge in diesem Objekt an.
Kodierung	1
Standardeinstellung	3
Bemerkung	Der Wert dieses Parameters kann nur gelesen werden.

Subindex	1
Datenlänge	4 Byte
Bedeutung	Gibt an, auf welchem Identifier das PDO(tx) für den Transfer der Prozesseingangsdaten gesendet wird. Zusätzlich sind noch einige Zusatzinformationen in den obersten Bits enthalten.
Kodierung	<p>Bit31(MSB) = 0 → Die Bearbeitung der Prozesseingangsdaten ist aktiv.</p> <p>Bit31(MSB) = 1 → Die Bearbeitung der Prozesseingangsdaten ist ausgeschaltet.</p> <p>Bit30 = 0 → Remote Frame auf dem entspr. Identifier wird beantwortet.</p> <p>Bit30 = 1 → Remote Frame wird nicht beantwortet.</p> <p>Bit29 = 0 → 11-Bit Identifier (CAN V2.0A)</p> <p>Bit29 = 1 → 29-Bit Identifier (CAN V2.0B), hier nicht einstellbar.</p> <p>Bit28...Bit0 = Identifier (Bit0 = LSB), hier für Bit28 bis Bit11=fest=0.</p>
Standardeinstellung	80000280h + Node_Id
Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert. Beim Einschalten der Prozessdatenbearbeitung (Bit31 von "1" auf "0") wird die Einstellung des Parameters 2nd transmit PDO Mapping auf ein entsprechendes Umrichter Mapping umgesetzt. Konnte dies erfolgreich durchgeführt werden, wird das Mapping automatisch nichtflüchtig abgespeichert. Da die Identifiervergabe der PDOs direkt von der Node_Id abgeleitet wird, können die Bits Bit28 bis Bit0 nur gelesen werden. Beim Schreiben werden diese Bits ignoriert.
Subindex	2
Datenlänge	1 Byte
Bedeutung	Bestimmt, wann und wie dieses Objekt auf dem CAN-Bus gesendet wird.
weiter auf nächster Seite	

Kodierung	<p>0 (synchron azyklisch): Bei jedem Empfang eines SYNC wird ein PDO(tx)-Telegramm auf CAN gesendet.</p> <p>1 - 240 (synchron, zyklisch): In diesem Einstellbereich wird über den Wert eingestellt, wie viele SYNC-Telegramme empfangen werden müssen, bevor ein PDO(tx)-Telegramm auf CAN gesendet wird.</p> <p>252 (synchron-RTROnly): Ein PDO(tx)-Telegramm wird nur nach einem Remote-Request auf dem PDO(tx)-Identifizier gesendet.</p> <p>253 (asynchron-RTROnly): Ein PDO(tx)-Telegramm wird nur nach einem Remote-Request auf dem PDO(tx)-Identifizier gesendet.</p> <p>254 (asynchron, herstellerspezifisch): Ein PDO(tx)-Telegramm wird gesendet, sobald sich mindestens ein Byte geändert hat.</p> <p>255 (asynchron, profilspezifisch): Ein PDO(tx)-Telegramm wird gesendet, sobald sich mindestens ein Byte geändert hat.</p>
Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert.

Subindex	3
Datenlänge	2 Byte
Bedeutung	Bezeichnet den minimalen zeitlichen Abstand zwischen zwei CAN-Telegrammen auf diesem Identifizier.
Kodierung	100 μ s
Standardeinstellung	1000 (= 100 ms)
Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert. Die interne Auflösung für die Inhibitzeit beträgt 1 ms. Somit hat der eingestellte Wert eine Ungenauigkeit von +/- 1 ms.

5.3 Parameter für das Life-Guarding

Guard Time

CAN-SDO-Index	100Ch
SDO-Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	2 Byte
Zugriff	READ_WRITE
Bedeutung	Gibt zusammen mit dem Life-Time-Faktor die Überwachungszeit für das Life-Guarding an.
Kodierung	0 = Life-Guarding abgeschaltet sonst = 1 ms
Erlaubte PDO-Abbildung	Nicht abbildbar
Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert.

Life Time Factor

CAN-SDO-Index	100Dh
SDO-Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	1 Byte
Zugriff	READ_WRITE
Bedeutung	Gibt zusammen mit der Guard Time die Überwachungszeit für das Life-Guarding an.
Kodierung	0 = Life-Guarding abgeschaltet sonst = 1
Erlaubte PDO-Abbildung	Nicht abbildbar
Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert.

LifeGuardTout.Addr

CAN-SDO-Index	5FDFh								
SDO-Subindex	0								
Objekttyp	Einfach Variable (Var)								
Datenlänge	4 Byte								
Zugriff	READ_WRITE								
Bedeutung	Bestimmt zusammen mit LifeGuardTout.Data die Funktion, die einmalig nach Auftreten des Life-Guarding-Timeout ausgeführt wird.								
Kodierung	<p>Der Wert besteht aus der zu schreibenden Parameteradresse und dem Parametersatz sowie einem Funktionscode für den Operator. Die unten stehende Abbildung zeigt den Aufbau des Wertes, wie er im CAN-SDO-Telegramm erscheint:</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">B0</td> <td style="text-align: center;">B1</td> <td style="text-align: center;">B2</td> <td style="text-align: center;">B3</td> </tr> <tr> <td style="text-align: center;">Opfunc</td> <td style="text-align: center;">Satz</td> <td style="text-align: center;">Parameteradresse LB</td> <td style="text-align: center;">HB</td> </tr> </table> <p>Opfunc = 0 → Keine Aktion im Operator 1 → Übergang in Pre_Operational</p> <p>Parameteradresse = Adresse des Parameters, der geschrieben wird</p> <p>Satz = Satzauswahlbyte des zu schreibenden Parameters.</p>	B0	B1	B2	B3	Opfunc	Satz	Parameteradresse LB	HB
B0	B1	B2	B3						
Opfunc	Satz	Parameteradresse LB	HB						
Erlaubte PDO-Abbildung	Nicht abbildbar								
Standardeinstellung	Parameteradresse = 0032h(Sy.50) Parametersatz = 1 (Satz 0) Opfunc = 1: Übergang in Pre_Operational								
Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert.								

LifeGuardTout.Data

CAN-SDO-Index	5FDEh
SDO-Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	4 Byte
Zugriff	READ_WRITE
Bedeutung	Bestimmt den Wert des zu schreibenden Umrichterparameters bei auftretendem Life-Guarding-Timeout.
Kodierung	Je nach ausgewähltem Umrichterparameter.
Erlaubte PDO-Abbildung	Nicht abbildbar
Standardeinstellung	1
Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert.

5.4 Parameter der Emergency-Bearbeitung

EmergencyCycle

CAN-SDO-Index	5FDDh
SDO-Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	2 Byte
Zugriff	READ_WRITE
Bedeutung	Dient zur Aktivierung/Deaktivierung der Emergency-Bearbeitung. Der Wert gibt zudem die Zykluszeit an, in der der Wert des Parameters "Umrichterstatus" gelesen wird, um einen eventuellen Fehler anzuzeigen.
Kodierung	0 = abgeschaltet sonst = 1 ms
Erlaubte PDO-Abbildung	Nicht abbildbar
Standardeinstellung	0 (abgeschaltet)
Bemerkung	Ein geänderter Wert wird sofort aktiv und nichtflüchtig gespeichert.

Predefined ErrorField

CAN-SDO-Index	1003h													
SDO-Subindex	0, 1, 2, 3, 4, 5													
Objekttyp	Einfach Variable (Var)													
Datenlänge	4 Byte													
Zugriff	Subindex = 0 → READ_WRITE sonst = READ_ONLY													
Bedeutung	Dieses Feld enthält die letzten fünf Fehlermeldungen. Dabei enthält das erste Fehlerfeld (Subindex = 1) immer den zuletzt aufgetretenen Fehler. Einträge mit höherem Subindex sind dementsprechend früher aufgetreten.													
Kodierung	Subindex = 0 → Anzahl gefüllter Einträge Subindex! = 0	<table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td>B0</td> <td>B1</td> <td>B2</td> <td>B3</td> </tr> <tr> <td colspan="2">ErrorCode</td> <td colspan="2">Umrichterstatus</td> </tr> <tr> <td>LB</td> <td>HB</td> <td>LB</td> <td>HB</td> </tr> </table>	B0	B1	B2	B3	ErrorCode		Umrichterstatus		LB	HB	LB	HB
B0	B1	B2	B3											
ErrorCode		Umrichterstatus												
LB	HB	LB	HB											
Erlaubte PDO-Abbildung	Nicht abbildbar													
Standardeinstellung	0 (abgeschaltet)													
Bemerkung	Nur der Subindex = 0 ist schreibbar. Beim Schreiben mit dem Wert = 0 wird das komplette Feld zurückgesetzt.													

5.5 Parameter für den Synchronmodus

ComCycle

CAN-SDO-Index	1006h
SDO-Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	4 Byte
Zugriff	READ_WRITE
Bedeutung	Dient zum Aktivieren / Deaktivieren des Synchronmodus. Der Wert ist kodiert in μs , hat aber eine interne Auflösung von 1ms.
Kodierung	0 = OFF (normaler Modus) sonst = 1 μs
Wertebereich	0, 1000, 2000, 3000...65000
Bemerkung	Der Parameter ist nur über CAN-SDO verfügbar. Der Parameter wird nicht nichtflüchtig gespeichert und hat nach jedem Neustart den Wert Null. Der CAN-Operator schaltet automatisch wieder in den normalen Modus zurück, wenn nach der vierfachen ComCycle-Zeit kein SYNC-Telegramm empfangen wird (s. auch HS_SyncToutDelay).

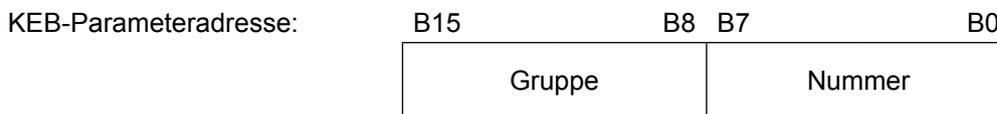
HS_SyncToutDelay

CAN-SDO-Index	5FE0h
SDO-Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	2 Byte
Zugriff	READ_WRITE
Bedeutung	Mit diesem Parameter kann die SYNC-Timeoutüberwachung um die gewünschte Anzahl von SYNCs verzögert werden.
Kodierung	0 = Die SYNC-Timeoutüberwachung ist sofort nach der Umschaltung in den Synchronmodus aktiv. sonst = Anzahl SYNC-Telegramme, bis die Timeoutüberwachung aktiv wird.
Wertebereich	0...65535
Bemerkung	Der Parameter wird nichtflüchtig gespeichert.

6. Zugriff auf Operatorparameter über die Diagnoseschnittstelle

Die Operatorparameter werden im Operator selbst verwaltet. Auf diese Parameter kann sowohl über die CAN-Schnittstelle als auch über die Diagnoseschnittstelle (per COMBIVIS) zugegriffen werden. Dabei ist zu beachten, dass ein CAN-Parameter teilweise auf mehrere Operatorparameter verteilt wird und somit das äußere Erscheinungsbild eines Parameters auf CAN und in COMBIVIS leicht unterschiedlich sein kann. In diesem Kapitel werden die für den Anwender relevanten Operatorparameter aufgelistet und der Verweis auf den entsprechenden CAN-Parameter genannt. Die komplette Beschreibung ist dann dem Kapitel Operatorparameter zu entnehmen.

Zur Adressierung von Operatorparametern und Parametern der Umrichtersteuerung folgende Information: Alle Parameter eines KEB-Umrichters (FU-Parameter + Operatorparameter) werden über die Diagnoseschnittstelle über eine 16-Bit Parameteradresse plus 8-Bit Satzauswahlbyte adressiert. Dabei gilt für die Parameteradresse, dass diese in die Parametergruppenadresse (High-Byte) und eine laufende Nummer (Low-Byte) aufgeteilt ist:



Jede Parametergruppe kann sowohl mit Operatorparametern als auch mit Parametern der Umrichtersteuerung belegt sein. Zur Unterscheidung der Lokalisierung des Parameters ist die laufende „Nummer“ in zwei Bereiche aufgeteilt:

Nummer = 0...127 → Parameter der Umrichtersteuerung
Nummer = 128...255 → Operatorparameter

Des Weiteren ist zu bemerken, dass, genau wie bei den Umrichterparametern, einige Operatorparameter mehrfach vorhanden sind. Diese nutzen dann das Satzauswahlbyte in KEB gewohnter Weise. Wobei gilt, dass über Satz 0 das erste Abbild des Parameters, in Satz 1 das zweite angesprochen wird. Derzeit beschränken sich die satzprogrammierbaren Operatorparameter auf die Prozessdatenparameter. Da der CAN-Operator derzeit zwei PDOs unterstützt, existieren diese Parameter in Satz 0 für das erste PDO und in Satz 1 für das zweite.

Alle satzprogrammierbaren Parameter haben unter Bemerkung einen entsprechenden Hinweis. Alle anderen Parameter existieren nur in Satz 0.

Beispiel:

Der Parameter PD_In_Para_CobId des ersten PDO wird über die Diagnoseschnittstelle adressiert mit Satzauswahlbyte = 01 (hex). Der gleiche Parameter des zweiten PDO liegt in Satz 1 (Satzauswahlbyte = 02 (hex)). Bitte beachten Sie, dass die gleichzeitige Adressierung eines Operatorparameters in mehreren Sätzen generell verboten ist.

Diag response delay time

Parameteradresse	0184h
Parameter-Id	OS.04
Datenlänge	1 Byte
Bedeutung	Minimale Antwortverzugszeit für Anfragen über die Diagnoseschnittstelle.
Kodierung	1 ms
Bemerkung	Dieser Parameter ist über CAN nicht verfügbar.

Diag Baudrate

Parameteradresse	0185h
Parameter-Id	OS.05
Datenlänge	1 Byte
Bedeutung	Gibt die Übertragungsgeschwindigkeit auf der Diagnoseschnittstelle an.
Kodierung	0 : 1200 Bit/s 1 : 2400 Bit/s 2 : 4800 Bit/s 3 : 9600 Bit/s 4 : 19200 Bit/s 5 : 38400 Bit/s
Bemerkung	Der Wert des Parameters ist ein Abbild des Parameters Sy.07 und ist deshalb über die Diagnoseschnittstelle nur lesbar.

HSP5 Max InvBusy retries

Parameteradresse	0186h
Parameter-Id	OS.06
Datenlänge	1 Byte
Bedeutung	Gibt an, wie oft ein HSP5-Dienst zur Umrichtersteuerung wiederholt wird, wenn dieser den Dienst mit dem Fehler 'Umrichter beschäftigt' ablehnt.
Kodierung	1

HSP5 Tout Count

Parameteradresse	0187h
Parameter-Id	OS.07
Datenlänge	2 Byte
Bedeutung	Zählt die Zeitüberschreitungen bei der internen Kommunikation zwischen Operator und FU-Steuerung.
Kodierung	1
Bemerkung	Dieser Parameter ist über CAN nicht verfügbar und existiert nur in Satz 0.

OP_Node_Id

Parameteradresse	0280h
Parameter-Id	FB.00
Datenlänge	1 Byte
Bemerkung	Dieser Parameter ist identisch mit dem CAN-Parameter OP_Node_Id und existiert nur in Satz 0.

CAN_Baud2

Parameteradresse	0281h
Parameter-Id	FB.01
Datenlänge	1 Byte
Bemerkung	Dieser Parameter entspricht dem CAN-Parameter CAN_Baud2 (s.o.) und existiert nur in Satz 0.

Act_CAN_Baud

Parameteradresse	0282h
Parameter-Id	FB.02
Datenlänge	1 Byte
Bedeutung	Zeigt die aktuell eingestellte CAN-Bitrate an.
Kodierung	Siehe CAN_BaUd.
Bemerkung	Dieser Parameter ist Read_Only und auf CAN nicht verfügbar und existiert nur in Satz 0.

Watchdog activation

Parameteradresse	0283h
Parameter-Id	FB.03
Datenlänge	1 Byte
Bemerkung	Dieser Parameter ist identisch mit dem CAN-Parameter Watchdog activation und existiert nur in Satz 0.

**Watchdog
inhibit**

Parameteradresse	0284h
Parameter-Id	FB.04
Datenlänge	1 Byte
Bemerkung	Dieser Parameter ist identisch mit dem CAN-Parameter Watchdog inhibit (s. o.) und existiert nur in Satz 0.

HS_PDO_Index

Parameteradresse	0285h
Parameter-Id	FB.05
Bemerkung	Dieser Parameter entspricht dem Operatorparameter HS_PDO_Index und existiert nur in Satz 0.

DSP402_OpMode

Parameteradresse	0286h
Parameter-Id	FB.06
Datenlänge	4 Byte
Bemerkung	Dieser Parameter entspricht dem Operatorparameter DSP402_ModesOfOperation und existiert nur in Satz 0.

PD_In_Para_CobID

Parameteradresse	0287h
Parameter-Id	FB.07
Datenlänge	4 Byte
Bemerkung	Dieser Parameter entspricht dem CAN-Parameter „nth transmit PDO Parameter, Cob ID“ * und existiert in Satz 0 und Satz 1.

PD_In_Para_TxType

Parameteradresse	0288h
Parameter-Id	FB.08
Datenlänge	1 Byte
Bemerkung	Dieser Parameter entspricht dem CAN-Parameter „nth transmit PDO Parameter, TxType“ und existiert in Satz 0 und Satz 1.

PD_In_Para_Inhibit

Parameteradresse	0289h
Parameter-Id	FB.09
Datenlänge	2 Byte
Bemerkung	Dieser Parameter entspricht dem CAN-Parameter „nth transmit PDO Parameter, Inhibit Time“ und existiert in Satz 0 und Satz 1.

PD_In_Cycle

Parameteradresse	028Ah
Parameter-Id	FB.10
Bemerkung	Dieser Parameter entspricht dem CAN-Parameter PDIN_Cycle_Time und existiert in Satz 0 (PDIN1_Cycle_Time) und Satz 1 (PDIN2_Cycle_Time).

Nr_PDIn_Obj

Parameteradresse	028Bh
Parameter-Id	FB.11
Datenlänge	1 Byte
Bemerkung	Dieser Parameter entspricht dem niederwertigsten Byte (LSB) des CAN-Parameters nth transmit PDO Mapping, Nr Mapped Objects und existiert in Satz 0 und Satz 1.

PD_Inx Index

(mit x = 1...4)

Parameteradresse	028Ch, 028Fh, 0292h, 0295h
Parameter-Id	FB.12, 15, 18, 21
Datenlänge	2 Byte
Bemerkung	Diese Parameter entsprechen dem höchstwertigen Wort (Bit31...Bit16) des Parameters ‚nth transmit PDO Mapping‘. Damit wird der Index für das Applikationsobjekt der nten PDO Abbildung definiert. Diese Parameter existieren jeweils in Satz0 (PDO1) und Satz1 (PDO2).

PD_Inx Set

(mit x = 1...4)

Parameteradresse	028Dh, 0290h, 0293h, 0296h
Parameter-Id	FB.13, 16, 19, 22
Datenlänge	1 Byte
Bemerkung	Diese Parameter entsprechen dem dritt höchstwertigen Byte (Bit15...Bit8) des Parameters ‚nth transmit PDO Mapping‘. Damit wird der Satz für das Applikationsobjekt der nten PDO Abbildung definiert. Diese Parameter existieren jeweils in Satz0 (PDO1) und Satz1 (PDO2).

PD_Inx_BitDien (mit x=1...4)

Parameteradresse	028Eh, 0291h, 0294h, 0297h
Parameter-Id	FB.14, 17, 20, 23
Bemerkung	Diese Parameter entsprechen dem niederwertigsten Wort (Bit7...Bit0) des Parameters ‚nth transmit PDO Mapping‘. Damit wird der Index für das Applikationsobjekt der nten PDO Abbildung definiert. Diese Parameter existieren jeweils in Satz0 (PDO1) und Satz1 (PDO2).

PDOOUT_WrMode

Parameteradresse	0299h
Parameter-Id	FB.25
Bemerkung	Dieser Parameter entspricht dem CAN-Parameter „PDOOUT_WrMode“ und existiert nur in Satz 0.

**PD_Out_Para_Co-
bID**

Parameteradresse	029Ah
Parameter-Id	FB.26
Datenlänge	4 Byte
Bemerkung	Dieser Parameter entspricht dem CAN-Parameter „nth receive PDO Parameter, Cob ID“ und existiert in Satz 0 und Satz 1.

**PD_Out_Para_Tx-
Type**

Parameteradresse	029Bh
Parameter-Id	FB.27
Datenlänge	1 Byte
Bemerkung	Dieser Parameter entspricht dem CAN-Parameter „nth receive PDO Parameter, TxType“ und existiert in Satz 0 und Satz 1.

Nr_PDOut_Obj

Parameteradresse	029Ch
Parameter-Id	FB.28
Datenlänge	1 Byte
Bemerkung	Diese Parameter entsprechen dem niederwertigsten Wort (Bit7...Bit0) des Parameters ‚nth receive PDO Mapping‘. Damit wird der Index für das Applikationsobjekt der nten PDO Abbildung definiert. Diese Parameter existieren jeweils in Satz0 (PDO1) und Satz1 (PDO2).

PD_Outx Index (mit x = 1...4)

Parameteradresse	029Dh, 02A0h, 02A3h, 02A6h
Parameter-Id	FB.29, 32, 35, 38
Datenlänge	2 Byte
Bemerkung	Diese Parameter entsprechen dem höchstwertigen Wort (Bit 31...Bit 16) des Parameters ‚nth receive PDO Mapping‘. Damit wird der Index für das Applikationsobjekt der nten PDO-Abbildung definiert. Diese Parameter existieren jeweils in Satz0 (PDO1) und Satz1 (PDO2).

PD_Outx Set (mit x = 1...4)

Parameteradresse	029Eh, 02A1h, 02A4h, 02A7h
Parameter-Id	FB.30, 33, 36, 39
Datenlänge	1 Byte
Bemerkung	Diese Parameter entsprechen dem dritt höchstwertigen Byte (Bit 15...Bit 8) des Parameters ‚nth receive PDO Mapping‘. Damit wird der Satz für das Applikationsobjekt der nten PDO-Abbildung definiert. Diese Parameter existieren jeweils in Satz0 (PDO1) und Satz1 (PDO2).

PD_Outx_BitDlen (mit x = 1...4)

Parameteradresse	029Fh, 02A2h, 02A5h, 02A8h
Parameter-Id	FB.31, 34, 37, 40
Bemerkung	Diese Parameter entsprechen dem niederwertigsten Wort (Bit 7...Bit 0) des Parameters ‚nth receive PDO Mapping‘. Damit wird der Index für das Applikationsobjekt der nten PDO-Abbildung definiert. Diese Parameter existieren jeweils in Satz0 (PDO1) und Satz1 (PDO2).

ProcessData Inx (mit x = 1...4)

Parameteradresse	02AAh - 02ADh
Parameter-Id	FB.42...45
Datenlänge	2 Byte
Bedeutung	x. Prozesseingangsdatenwort
Kodierung	Je nach abgebildetem Parameter
Bemerkung	Dieser Parameter ist Read_Only und entspricht dem x. Wort des PDO(tx)-Telegramms auf CAN und existiert in Satz 0 und Satz 1.

ProcessData Outx (mit x = 1...4)

Parameteradresse	02AEh - 02B1h
Parameter-Id	FB.46...49
Datenlänge	2 Byte
Bedeutung	x. Prozessausgangsdatenwort
Kodierung	Je nach abgebildetem Parameter
Bemerkung	Dieser Parameter ist Read_Only und entspricht dem x. Wort des PDO(rx)-Telegramms auf CAN und existiert in Satz 0 und Satz 1.

Take Stored PD-Map

Parameteradresse	02B2h
Parameter-Id	FB.50
Bemerkung	Dieser Parameter entspricht dem CAN Parameter PD_Stored (Index = 5FE2h), (s. o.) und existiert nur in Satz 0.

Check PD Setting

Parameteradresse	02B3h
Parameter-Id	FB.51
Datenlänge	1 Byte
Bedeutung	Zeigt an, ob die zuletzt eingestellte PD-Belegungsänderung fehlerfrei durchgeführt wurde.
Kodierung	0 : Fehler aufgetreten in letzter PD-Belegungsänderung. 255d : Letzte PD-Belegungsänderung ist fehlerfrei durchgeführt worden.
Bemerkung	Dieser Parameter ist auf CAN nicht verfügbar und existiert nur in Satz 0.

ComCycle

Parameteradresse	02B4h
Parameter-Id	FB.52
Bemerkung	Dieser Parameter entspricht dem CAN-Parameter ComCycle und existiert nur in Satz 0.

HS_SyncToutDelay

	02B5h
Parameteradresse	
Parameter-Id	FB.53
Bemerkung	Dieser Parameter entspricht dem CAN-Parameter HS_SyncToutDelay und existiert nur in Satz 0.

LifeGuardTout.Addr

Parameteradresse	02B6h
Parameter-Id	FB.54
Bemerkung	Dieser Parameter entspricht dem CAN-Parameter LifeGuardTout.Addr und existiert nur in Satz 0.

LifeGuardTout.Data

Parameteradresse	02B7h
Parameter-Id	FB.55
Bemerkung	Dieser Parameter entspricht dem CAN-Parameter LifeGuardTout.Data und existiert nur in Satz 0.

EmergencyCycle

Parameteradresse	02B8h
Parameter-Id	FB.56
Bemerkung	Dieser Parameter entspricht dem CAN-Parameter EmergencyCycle und existiert nur in Satz 0.

Save_VLRamps

Parameteradresse	02B9h
Parameter-Id	FB.57
Bemerkung	Dieser Parameter entspricht dem CAN-Parameter Save_VLRamps und existiert nur in Satz 0.

VL_Ramp_CalcMode

Parameteradresse	02BAh
Parameter-Id	FB.58
Bemerkung	Dieser Parameter entspricht dem CAN-Parameter VL_Ramp_CalcMode und existiert nur in Satz 0.

7. Umschaltung des transmission-type der PDOs

Der transmission-type sowohl des Parameters 1st/2nd receive PDO Parameter als auch des 1st/2nd transmit PDO Parameter ist veränderbar. Die gültigen Werte sind:

- Asynchron herstellerspezifisch (Wert = 254 = Standard)
- Asynchron profilspezifisch (Wert = 255)
- Synchron azyklisch (Wert = 0)
- Synchron zyklisch (Wert = 1)
- Synchron zyklisch (Werte = 1, 2...240)
- Synchron RTROnly (Wert = 252)
- Asynchron RTROnly (Wert = 253)

Nach CANopen besitzen beim PDO(rx) die Werte 0 bis 240 identisches Verhalten. D. h., dass beim PDO(rx) die PDOOUT-Daten bei jedem SYNC aktualisiert werden, egal welchen Wert der tx-type genau hat. Beachten Sie bitte auch den Einfluss des Parameters PDOOUT_WrMode für die PDO1, PDO2(rx)-Bearbeitung.

7.1 Asynchron herstellerspezifisch (Wert = 254d/FEh) oder Asynchron profilspezifisch (Wert = 255d/FFh)

Wenn im Parameter 1st/2nd receive PDO Parameter der transmission-type auf einen dieser Werte eingestellt ist, bedeutet dies, dass die Prozessausgangsdaten im Zustand OPERATIONAL beim Empfang eines gültigen PDO(rx)-Telegramms zur Umrichtersteuerung transferiert werden, wenn sich mindestens 1 Byte geändert hat. Ein gültiges PDO(rx)-Telegramm ist ein Telegramm auf dem entsprechenden Identifier mit einer Datenlänge von > der Datenlänge, die sich aus dem PDO(rx)-Mapping ergibt. Im Standardfall bedeutet das, dass alle Telegramme auf dem OUT-Identifier mit einer Datenlänge von > 4 Byte akzeptiert werden.

Im Zustand OPERATIONAL werden zudem die Prozesseingangsdaten zyklisch von der FU-Steuerung gelesen. Wenn im Parameter 1st/2nd transmit PDO Parameter der Wert 254d oder 255d eingestellt ist, bedeutet dies, dass ein PDO(tx)-Telegramm auf den CAN gesendet wird, wenn sich die Prozesseingangsdaten geändert haben.

7.2 Synchron azyklisch (Wert = 0) oder synchron zyklisch (Werte = 1 bis 240)

Wenn im Parameter 1st/2nd receive PDO Parameter der transmission-type auf einen dieser Werte eingestellt ist, bedeutet dies, dass die Prozessausgangsdaten im Zustand OPERATIONAL nach Empfang eines SYNC-Telegramms zur Umrichtersteuerung transferiert werden. Vorausgesetzt, zuvor wurde ein gültiges PDO(rx)-Telegramm empfangen.

Für den Parameter 1st/2nd transmit PDO Parameter bedeutet der Wert transmission-type = 0/1, dass im Zustand OPERATIONAL ein PDO(tx)-Telegramm umgehend nach Empfang eines SYNC-Telegramms auf den CAN gesendet wird.

Für alle synchronen Werte des tx_type gilt, dass das SYNC-Telegramm das Senden des betreffenden PDO(tx) bzw. die Weiterleitung des PDO(rx) auslöst. Durch den genauen Wert wird dann nur noch bestimmt, wie viele SYNC-Telegramme dazu erforderlich sind. Bei den Werten 0 und 1 aktiviert jeder SYNC das entsprechende Ereignis. Bei den Werten 2 bis 240 gibt der Wert selbst die Anzahl erforderlicher SYNC-Telegramme an. Allerdings ist hierbei zu beachten, dass das Verhalten in diesem Wertebereich für alle PDO(rx)-Telegramme identisch mit dem Wert = 1 ist.

Beispiel:

- PDO1(tx).tx_type = 10 → Nach jeweils 10 SYNC-Telegrammen sendet der Slave eine PDO1(tx) auf CAN mit den aktuellen PDIN1-Daten.
- PDO1(rx).tx_type = 10 → Nach jedem SYNC-Telegramm werden die aktuellen PDOOUT1-Daten weitergegeben.

7.3 Synchron / asynchron RTROnly (Werte = 252, 253)

Diese Werte sind nur gültig für tx-PDOs. Das Senden der aktuellen PDIN-Daten über das entsprechende PDO(tx)-Telegramm auf CAN wird hierbei nur bei Empfang eines Remote-Frame-Request auf dem entsprechenden Identifier gestartet.

8. Synchronmodus

Im Synchronmodus wird über das SYNC-Telegramm auf CAN der interne Verarbeitungstakt des CAN-Operators und der angeschlossenen Frequenzumrichtersteuerung vorgegeben. Kürzeste Verzögerungszeiten und vor allem extrem wenig zeitliche Abweichung in den Verzögerungszeiten sind bei dieser Betriebsart das oberste Ziel. Dieses wird erreicht bei gleichzeitiger Kompatibilität auf CAN. Allerdings sind mit dem Synchronmodus deutliche Funktionseinschränkungen verbunden. Der generelle Betrieb bleibt aber erhalten.

Im Synchronmodus liegt die höchste Priorität auf dem schnellst möglichen Transfer der Prozessdaten. Die Abbildung der Prozessdaten ist über die Prozessdatenabbildung einstellbar und unterliegt lediglich den Einschränkungen des schon bekannten High-Speed-PDOs im CAN-Operator.

Folgende Bedingungen gelten für den Synchronmodus:

Der Synchron-Modus arbeitet nur im OPERATIONAL-Zustand des Knoten.

Es darf nur das High-Speed-PDO aktiv sein.

Das PDO arbeitet in beiden Richtungen Synchron.

Das PDO-Mapping in beiden Richtungen erfüllt folgende Bedingungen:

- Anzahl abgebildeter Parameter in beiden Richtungen = 2 oder 3
- Die erste Abbildung belegt 32-Bit
- Jede weitere Abbildung belegt 16-Bit

Aktiviert wird der Synchronmodus durch Schreiben auf den neuen Parameter ComCycle mit einem Wert ungleich Null.

Im Synchronmodus müssen die SYNC-Telegramme auf CAN in der eingestellten Zeit (ComCycle) gesendet werden. Die maximale zeitliche Abweichung zweier aufeinander folgender SYNC-Telegramme darf ca. 80µs nicht überschreiten. Wird diese maximale Abweichung überschritten, muss im Einzelfall geklärt werden, ob die Applikation in gewünschter Weise funktioniert. Der CAN-Operator überwacht das Empfangen der SYNC-Telegramme. Wenn innerhalb der Timeoutzeit kein SYNC empfangen wird, schaltet der Operator automatisch zurück in den normalen Modus. Die Timeoutzeit entspricht dem vierfachen der erwarteten SYNC-Zykluszeit (ComCycle).

Die notwendigen Voreinstellungen sind nochmals in folgender Tabelle zusammengefasst:

Index	Subindex	Dlen	Wert
5FE5h	0	1	0
1801h	1	4	8000XXXXh
1401h	1	4	8000XXXXh
1800h	2	1	0 oder 1
1400h	2	1	0 oder 1
1A00h	0	1	< 3
1A00h	1	4	XXXXXX20h
1A00h	2	4	XXXXXX10h
1A00h	3	4	XXXXXX10h
1600h	0	1	< 3
1600h	1	4	XXXXXX20h
1600h	2	4	XXXXXX10h
1600h	3	4	XXXXXX10h
1006h	0	4	Vielfaches von 1000 (dez)

8.1 Funktionseinschränkungen im Synchronmodus

Im Synchronmodus werden alle CAN-SDO-Aufträge und Aufträge von der Diagnoseschnittstelle mit in den Prozessdatentransfer eingeflochten. Aus diesem Grund sind dann nur noch CAN-SDO-Zugriffe auf Parameter in der Umrichtersteuerung mit Subindex = 0 möglich. D.h. es können Parameter im Umrichter nur noch in dem durch den Satzzeiger (Fr.09) bestimmten Satz angesprochen werden (indirekte Satzadressierung). Beachten Sie bitte auch, dass jedes CAN-Telegramm den SYNC zeitlich verschieben kann, auch wenn das SYNC-Telegramm durch seinen niedrigen Identifier eine sehr hohe Priorität besitzt. Aus diesem Grund ist nach Möglichkeit im Synchronmodus auf jegliche andere CAN-Kommunikation zu verzichten (Node-Guarding, SDO-Kommandos, NMT-Kommandos). Es sollten dann wirklich nur PDO(rx)-Telegramme, PDO(tx)-Telegramme und der SYNC gesendet werden.

Die Tastatur wird im Synchronmodus nicht bearbeitet. Die Anzeige ist statisch und zeigt ‚Synch‘ an. Die Diagnoseschnittstelle arbeitet weiter mit ähnlichen Einschränkungen, die für die CAN-SDO-Kommunikation gilt: Es können Parameter im Umrichter nur noch über den HSP5-Dienst = 1 mit dem Satzauswahl-Byte = 1 (indirekte Satzadressierung über Fr.09) gelesen oder geschrieben werden.

9. DSP402-Unterstützung

Welche DSP402-Modi unterstützt werden, ist der Beschreibung des Parameters DSP402_ModesOfOperation zu entnehmen.

Die CAN in Automation Nutzergruppe hat am 26.07.2002 die Version 2.0 des DSP402-Geräteprofils für Antriebe veröffentlicht. Die KEB-F5-CANopen-Anschaltung unterstützt eine Teilmenge der Funktionen und Parameter, die in dem DSP402 definiert sind. Dabei übernimmt der CAN-Operator die Umsetzung der DSP402-Parameter auf Parameter der Umrichtersteuerung. Diese Umsetzung ist teilweise aufwendig und deshalb auch laufzeitintensiv. Aus diesem Grund ist eine Abbildung solcher Parameter, die umkodiert werden müssen, auf das High-Speed-PDO in den meisten Fällen nicht erlaubt. Die DSP402-Parameter können aber über SDO-Kommandos angesprochen werden. Ebenso sind fast alle DSP402-Parameter auf das Low-Speed-PDO abbildbar.

Einige Parameter im KEB-F5-Frequenzumrichter, die als Basis für realisierte DSP402-Parameter dienen, sind satzprogrammierbar. Da das DSP402-Profil keine Satzprogrammierung unterstützt, wurde für die DSP402-Realisierung folgende Festlegung getroffen: Alle DSP402-Profilparameter, die auf Parameter in der Frequenzumrichtersteuerung umgesetzt werden, werden im Satz 0 abgelegt. Auf die Parameter in anderen Sätzen haben diese keinen Einfluss:

Das DSP402-Profil arbeitet ausschließlich in Parametersatz 0

9.1 Voreinstellungen für DSP402-Betrieb

Das DSP402-Profil unterstützt keine Unterscheidung der Rampen für Rechtslauf und für Linkslauf. Aus diesem Grund müssen die Rampenzeiten für Rechts- und Linkslauf gleiche Werte haben. Das bedingt die folgenden Voreinstellungen in der Umrichtersteuerung:

Parameter	Parameteradresse	Parametersatz	Parameterwert
oP.29	031Dh	Satz 0	-1
oP.31	31Fh	Satz 0	-1

Zum Betrieb über das DSP402-Steuer-und Statuswort sind folgende Voreinstellungen in der Umrichtersteuerung vorzunehmen:

Parameter	Parameteradresse	Parametersatz	Parameterwert
Ud.01	0801h	Satz 0	440
oP.00	0300h	Satz 0	5
oP.01	0301h	Satz 0	6
oP.02	0302h	Satz 0	0
oP.60	033Ch	Satz 0	0
oP.61	033Dh	Satz 0	0
di.01	0B01h	Satz 0	Bit0 = 1
di.02	0B02h	Satz 0	Bit0 = 1
di.09	0B09h	Satz 0	2

9.2 Hinweise zu den DSP402-Velocity Rampen

Das DSP402-Profil definiert eine Velocity-Rampe (VL-Rampe) als eine Struktur aufgebaut aus zwei Teilen:

- VL-Rampe.Dspeed : Delta-Geschwindigkeitswert der Rampe in min^{-1}
- VL-Rampe.Dtime : Delta-Zeitwert der Rampe in Sekunden.

Intern in der Umrichtersteuerung wird eine Rampe durch einen festen Teil, den sogenannten Rampenreferenzwert und einen einstellbaren Wert, die Rampenzeit festgelegt. Die DSP402-Rampenwerte werden im CAN-Operator gehalten. Bei einem Lesezugriff wird auf diese Zwischenspeicherwerte zugegriffen. Wird einer der Werte geschrieben ist zusätzlich ein Schreibzugriff auf die entsprechende Rampenzeit in der Umrichtersteuerung erforderlich. Die Rampenwerte nach DSP402-Kodierung werden im Operator nicht automatisch nichtflüchtig gespeichert. Dies kann der Anwender explizit anfordern über den Parameter Save_VL_Ramps.

Die Umsetzung einer VL-Rampe in eine FU-Rampenzeit ist eindeutig. Allerdings ist die ebenfalls notwendige Umsetzung einer FU-Rampenzeit in eine VL-Rampe nicht eindeutig. Aus diesem Grund musste eine Methode für diese Rückkonvertierung gefunden werden. Der KEB-F5-CANopen-Operator unterstützt verschiedene Rückkonvertierungsmodi, welche über den Parameter VL_Ramp_CalcMode ausgewählt werden können (s.u.). Weiterhin ist zu beachten, dass bei jeder Veränderung einer der beiden VL-Rampenteile immer ein Schreibzugriff auf die relevante FU-Rampenzeit durchgeführt wird. Das bedeutet, dass bei einer Änderung beider VL-Rampenteile zunächst der eine Teil in eine FU-Rampenzeit umgesetzt wird. Diese FU-Rampenzeit entspricht zu diesem Zeitpunkt nicht der gewünschten Rampe. Erst wenn danach auch der zweite VL-Rampenteil geschrieben wird, wird die gewünschte Rampe als FU-Rampenzeit vorgegeben. Diese Problematik gilt auch für das Low-Speed-PDO. Das DSP402-Profil macht für die konsistente Vorgabe der VL-Rampen keine Vorschriften. Das eben geschilderte Problem muss demnach der Anwender lösen:

- Z.B. kann man einen der beiden VL-Rampenteile immer unverändert lassen und die Rampe nur über den anderen Teil der VL-Rampe variieren.
- Ebenso wäre ein Ansatz die Rampen niemals zu ändern während die FU Rampe fährt.

9.3 DSP402-Profil und Synchronmodus

Es ist zwar generell möglich auch im Synchronmodus über DSP402-Profilparameter zu arbeiten, allerdings ist dabei zu berücksichtigen, dass im Synchronmodus keine Konvertierung / Dekonvertierung von Parameterwerten unterstützt wird. Demzufolge sind umzurechnende DSP402-Parameter im Synchronmodus nicht auf Prozessdaten abbildbar. Die Parameter, die keine Umrechnung benötigen sind:

- VL_TargetVelocity (Index=6042h)
- VL_ControlEffort (Index=6043h)

Ebenso ist im Synchronmodus der direkte satzadressierte SDO-Zugriff nicht erlaubt. Also würden die meisten SDO-Zugriffe auf DSP402-Parameter mit Fehler abgelehnt.

Aus diesen Gründen ist es praktisch unrealistisch, im Synchronmodus mit DSP402-Parametern zu arbeiten.

9.4 Allgemeine Parameter des DSP402-Profiles

DSP402_ErrorCode

CAN-SDO-Index	603Fh			
SDO-Subindex	0			
Objekttyp	Einfach Variable (Var)			
Datenlänge	2 Byte			
Zugriff	READ_ONLY			
Bedeutung	Gibt den aktuellen Fehlerstatus des Gerätes an.			
Kodierung	Nach DSP402-Vorgabe, s. Tabelle im Anhang			
Erlaubte PDO-Abbildung	High-Speed-PDO		Low-Speed-PDO	
	rx	tx	rx	tx
	NEIN	JA	NEIN	JA
Bemerkung	Wird intern auf den Parameter ru.00 abgebildet. Wenn der FU keinen Fehlerzustand meldet, aber der CAN-Operator einen Fehler erkannt hat, wird dieser zurückgegeben.			

DSP402-Unterstützung

DSP402_Controlword

CAN-SDO-Index	6040h																																													
SDO-Subindex	0																																													
Objekttyp	Einfach Variable (Var)																																													
Datenlänge	2 Byte																																													
Zugriff	READ_WRITE																																													
Bedeutung	Dient zur Vorgabe von Steuerungskommandos. Der Parameter ist bitkodiert und wird in der Umrichtersteuerung auf den Parameter Sy.50 (Steuerwort) abgebildet.																																													
Kodierung	<p>Im folgenden Bild werden nur die unterstützten Bits aufgeführt:</p> <table border="1" style="width: 100%; text-align: center;"> <thead> <tr> <th>B15</th> <th>B7</th> <th>B6</th> <th>B5</th> <th>B4</th> <th>B3</th> <th>B2</th> <th>B1</th> <th>B0</th> </tr> </thead> <tbody> <tr> <td>...</td> <td>Fault reset</td> <td colspan="3">Operation mode specific</td> <td>Enable operation</td> <td>Quick stop</td> <td>Enable voltage</td> <td>Switch on</td> </tr> <tr> <td colspan="2"></td> <td>RFG use ref</td> <td>RFG unlock</td> <td>RFG enable</td> <td colspan="4">Velocity Mode</td> </tr> <tr> <td colspan="2"></td> <td>reserved</td> <td>reserved</td> <td>Homing operation start</td> <td colspan="4">Homing Mode</td> </tr> <tr> <td colspan="2"></td> <td>Abs / Rel</td> <td>Change set im-med.</td> <td>New setpoint</td> <td colspan="4">Profile Position Mode</td> </tr> </tbody> </table> <p>Die Bits B6 bis B4 sind modeabhängig definiert. Grau hinterlegte Bits sind derzeit in der KEB-CANopen-Anschaltung nicht realisiert.</p>	B15	B7	B6	B5	B4	B3	B2	B1	B0	...	Fault reset	Operation mode specific			Enable operation	Quick stop	Enable voltage	Switch on			RFG use ref	RFG unlock	RFG enable	Velocity Mode						reserved	reserved	Homing operation start	Homing Mode						Abs / Rel	Change set im-med.	New setpoint	Profile Position Mode			
B15	B7	B6	B5	B4	B3	B2	B1	B0																																						
...	Fault reset	Operation mode specific			Enable operation	Quick stop	Enable voltage	Switch on																																						
		RFG use ref	RFG unlock	RFG enable	Velocity Mode																																									
		reserved	reserved	Homing operation start	Homing Mode																																									
		Abs / Rel	Change set im-med.	New setpoint	Profile Position Mode																																									
Erlaubte PDO-Abbildung	High-Speed-PDO	Low-Speed-PDO																																												
	rx tx	rx tx																																												
	JA JA	JA JA																																												
Bemerkung	Wird intern auf den Parameter Sy.50 abgebildet.																																													

DSP402_
Statusword

CAN-SDO-Index	6041h																																																																							
SDO-Subindex	0																																																																							
Objekttyp	Einfach Variable (Var)																																																																							
Datenlänge	2 Byte																																																																							
Zugriff	READ_ONLY																																																																							
Bedeutung	Dient zur Bekanntgabe des aktuellen Zustands. Der Parameter ist bitkodiert und wird in der Umrichtersteuerung auf den Parameter Sy.51 (Statusword) abgebildet.																																																																							
<p>Kodierung</p> <p>Im folgenden Bild werden nur die unterstützten Bits aufgeführt:</p> <table border="1" style="width: 100%; text-align: center;"> <thead> <tr> <th>B15</th> <th>B13</th> <th>B12</th> <th>B11</th> <th>B10</th> <th>B9</th> <th>...</th> <th>B6</th> <th>B5</th> <th>B4</th> <th>B3</th> <th>B2</th> <th>B1</th> <th>B0</th> </tr> </thead> <tbody> <tr> <td>...</td> <td>Operation mode specific</td> <td>Operation mode specific</td> <td>Internal limit active</td> <td>Target reached</td> <td>Re-mote</td> <td>...</td> <td>Switch on disabled</td> <td>Quick stop</td> <td>Voltage enabled</td> <td>Fault</td> <td>Operation enabled</td> <td>Switched on</td> <td>Ready to switch on</td> </tr> <tr> <td></td> <td>reser-ved</td> <td>reser-ved</td> <td colspan="2">Velocity Mode</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td>Ho-ming error</td> <td>Ho-ming attained</td> <td colspan="2">Homing Mode</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td>Fol-lowing error</td> <td>Set-point ackn.</td> <td colspan="2">Profile Position Mode</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table> <p>Die Bits B13, B12 sind modeabhängig definiert. Grau hinterlegte Bits sind derzeit in der KEB-CANopen-Anschaltung nicht realisiert.</p>			B15	B13	B12	B11	B10	B9	...	B6	B5	B4	B3	B2	B1	B0	...	Operation mode specific	Operation mode specific	Internal limit active	Target reached	Re-mote	...	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on		reser-ved	reser-ved	Velocity Mode												Ho-ming error	Ho-ming attained	Homing Mode												Fol-lowing error	Set-point ackn.	Profile Position Mode										
B15	B13	B12	B11	B10	B9	...	B6	B5	B4	B3	B2	B1	B0																																																											
...	Operation mode specific	Operation mode specific	Internal limit active	Target reached	Re-mote	...	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on																																																											
	reser-ved	reser-ved	Velocity Mode																																																																					
	Ho-ming error	Ho-ming attained	Homing Mode																																																																					
	Fol-lowing error	Set-point ackn.	Profile Position Mode																																																																					
Erlaubte PDO-Abbildung	High-Speed-PDO		Low-Speed-PDO																																																																					
	rx	tx	rx	tx																																																																				
	NEIN	JA	NEIN	JA																																																																				
Bemerkung	Wird intern auf den Parameter Sy.51 abgebildet.																																																																							

DSP402-Unterstützung

DSP402_ModesOf-
Operation

CAN-SDO-Index	6060h
SDO-Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	1 Byte
Zugriff	READ_WRITE
Bedeutung	Dient zur Vorgabe des gewünschten DSP402-Arbeitsmodus.
Kodierung	(-1) : Herstellerspezifisch 0 : Reserviert 1 : Profile Position Mode*1 2 : Velocity Mode 3 : Profile Velocity Mode(hier nicht möglich) 4 : Torque Profile Mode(hier nicht möglich) 5 : Reserviert 6 : Homing Mode*1 7 : Interpolated Position Mode(hier nicht möglich)
Standardeinstellung	(-1) : Herstellerspezifisch
Erlaubte PDO-Abbildung	Nicht abbildbar
Bemerkung	Derzeit besteht kein realer Unterschied zwischen den Modi (-1) und 2.

*1: Diese Modi sind nur bei Steuerungstypen Ud.02 = 4, 5, 6, 8, 9, 10 erlaubt.

DSP402_ModesOf-
OperationDisp

CAN-SDO-Index	6061h
SDO-Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	1 Byte
Zugriff	READ_ONLY
Bedeutung	Gibt den aktuellen DSP402-Arbeitsmodus an.
Kodierung	s. DSP402_ModesOfOperation
Standardeinstellung	(-1) : Herstellerspezifisch
Erlaubte PDO-Abbildung	Nicht abbildbar

DSP402_Abort-
Conn
OptionCode

CAN-SDO-Index	6007h
SDO-Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	2 Byte
Zugriff	READ_WRITE
Bedeutung	Bestimmt das Verhalten nachdem die Verbindung zum CAN abgebrochen wurde. Abbruch der Verbindung zum CAN ist hier gleich bedeutend mit den folgenden Ereignissen: <ul style="list-style-type: none"> - Ansprechen der Life-Guarding-Timeout-Überwachung - BusOff-Zustand des CAN-Controllers
Kodierung	(-1) : Das Verhalten nach einem Verbindungsabbruch auf CAN wird durch die beiden Parameter LifeGuardTout_Addr und LifeGuardTout_Data bestimmt, s.o. 0 : Keine Aktion 1 : Übergang in PRE_OPERATIONAL, wenn aktueller Zustand OPERATIONAL und explizites Auslösen des Feldbus-Watchdog (E.Bus) bei der Umrichtersteuerung (wenn diese im FU aktiviert). 2 : Übergang in PRE_OPERATIONAL, wenn aktueller Zustand OPERATIONAL und Kommando, 'Disable Voltage' über das DSP402_Controlword. 3 : Übergang in PRE_OPERATIONAL, wenn aktueller Zustand OPERATIONAL und Kommando, 'Quick Stop' über das DSP402_Controlword.
Standardeinstellung	(-1)
Erlaubte PDO-Abbildung	Nicht abbildbar
Bemerkung	Eine Wertänderung wird sofort aktiv und nichtflüchtig gespeichert.

DSP402-Unterstützung

DSP402_Motion-ProfileType

CAN-SDO-Index	6086h
SDO-Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	2 Byte
Zugriff	READ_WRITE
Bedeutung	Gibt die Art der Realisierung von Bewegungen an.
Kodierung	(-1) : Lineare Rampen mit aktivierbaren s-Kurven 0 : Lineare Rampen (hier nicht einstellbar) 1 : sin2-Rampen (hier nicht einstellbar) 2 : Ruckfreie Rampen (hier nicht einstellbar) 3 : Ruckbegrenzte Rampen (hier nicht einstellbar)
Standardeinstellung	(-1)
Erlaubte PDO-Abbildung	Nicht abbildbar
Bemerkung	Derzeit ist nur ein Wert wählbar.

DSP402_SuppDrive Modes

CAN-SDO-Index	6502h																						
SDO-Subindex	0																						
Objekttyp	Einfach Variable (Var)																						
Datenlänge	4 Byte																						
Zugriff	READ_ONLY																						
Bedeutung	Gibt bitkodiert die unterstützten Modi der CANopen-Anschaltung an.																						
Kodierung	<table border="1"> <thead> <tr> <th>B31</th> <th>B16</th> <th>B15</th> <th>B7</th> <th>B6</th> <th>B5</th> <th>B4</th> <th>B3</th> <th>B2</th> <th>B1</th> <th>B0</th> </tr> </thead> <tbody> <tr> <td>hersteller-spezifisch</td> <td>reserviert</td> <td></td> <td></td> <td>Interpolated position</td> <td>Hom-ing</td> <td>reserviert</td> <td>Torque</td> <td>Profile Velocity</td> <td>Velocity</td> <td>Profile position</td> </tr> </tbody> </table> <p>Die herstellereigenen Bits werden hier nicht genutzt.</p>	B31	B16	B15	B7	B6	B5	B4	B3	B2	B1	B0	hersteller-spezifisch	reserviert			Interpolated position	Hom-ing	reserviert	Torque	Profile Velocity	Velocity	Profile position
B31	B16	B15	B7	B6	B5	B4	B3	B2	B1	B0													
hersteller-spezifisch	reserviert			Interpolated position	Hom-ing	reserviert	Torque	Profile Velocity	Velocity	Profile position													
Erlaubte PDO-Abbildung	Nicht abbildbar																						

9.5 Parameter des Velocity Mode

VL_TargetVelocity

CAN-SDO-Index	6042h			
SDO-Subindex	0			
Objekttyp	Einfach Variable (Var)			
Datenlänge	2 Byte			
Zugriff	READ_WRITE			
Bedeutung	Gibt die Sollgeschwindigkeit vor.			
Kodierung	1 min-1			
Erlaubte PDO-Abbildung	High-Speed-PDO		Low-Speed-PDO	
	rx	tx	rx	tx
	JA	JA	JA	JA
Bemerkung	Wird intern auf den Parameter Sy.52 abgebildet.			

VL_VelocityDemand

CAN-SDO-Index	6043h			
SDO-Subindex	0			
Objekttyp	Einfach Variable (Var)			
Datenlänge	2 Byte			
Zugriff	READ_ONLY			
Bedeutung	Gibt den Geschwindigkeitswert am Ausgang des Rampengenerators an.			
Kodierung	1 min-1			
Erlaubte PDO-Abbildung	High-Speed-PDO		Low-Speed-PDO	
	rx	tx	rx	tx
	NEIN	JA	NEIN	JA
Bemerkung	Wird intern auf den Parameter ru.02 abgebildet.			

VL_ControlEffort

CAN-SDO-Index	6044h			
SDO-Subindex	0			
Objekttyp	Einfach Variable (Var)			
Datenlänge	2 Byte			
Zugriff	READ_ONLY			
Bedeutung	Gibt den Geschwindigkeitswert an.			
Kodierung	1 min-1			
Erlaubte PDO-Abbildung	High-Speed-PDO		Low-Speed-PDO	
	rx	tx	rx	tx
	NEIN	JA	NEIN	JA
Bemerkung	Wird intern auf den Parameter Sy.53 abgebildet.			

VL_VelocityMin
Amount

CAN-SDO-Index	6046h			
SDO-Subindex	1			
Objekttyp	Einfach Variable (Var)			
Datenlänge	4 Byte			
Zugriff	READ_WRITE			
Bedeutung	Gibt den Betrag des unteren Grenzwertes des Sollwertes an.			
Kodierung	1 min-1			
Erlaubte PDO-Abbildung	High-Speed-PDO		Low-Speed-PDO	
	rx	tx	rx	tx
	NEIN	NEIN	JA	JA
Bemerkung	Wird intern auf den Parameter oP.06 abgebildet.			

VL_VelocityMax
Amount

CAN-SDO-Index	6046h			
SDO-Subindex	2			
Objekttyp	Einfach Variable (Var)			
Datenlänge	4 Byte			
Zugriff	READ_WRITE			
Bedeutung	Gibt den Betrag des oberen Grenzwertes des Sollwertes an.			
Kodierung	1 min-1			
Erlaubte PDO-Abbildung	High-Speed-PDO		Low-Speed-PDO	
	rx	tx	rx	tx
	NEIN	NEIN	JA	JA
Bemerkung	Wird intern auf den Parameter oP.10 abgebildet.			

VL_VelocityAcceleration.Dspeed

CAN-SDO-Index	6048h			
SDO-Subindex	1			
Objekttyp	Einfach Variable (Var)			
Datenlänge	4 Byte			
Zugriff	READ_WRITE			
Bedeutung	Gibt zusammen mit VL_VelocityAcceleration.Dtime die Beschleunigungsrampe vor.			
Kodierung	1 min-1			
Erlaubte PDO-Abbildung	High-Speed-PDO		Low-Speed-PDO	
	rx	tx	rx	tx
	NEIN	NEIN	JA	JA
Bemerkung	Wird intern auf den Parameter oP.28 abgebildet.			

VL_VelocityAcceleration.Dtime

CAN-SDO-Index	6048h			
SDO-Subindex	2			
Objekttyp	Einfach Variable (Var)			
Datenlänge	2 Byte			
Zugriff	READ_WRITE			
Bedeutung	Gibt zusammen mit VL_VelocityAcceleration.Dspeed die Beschleunigungsrampe vor.			
Kodierung	1 s			
Erlaubte PDO-Abbildung	High-Speed-PDO		Low-Speed-PDO	
	rx	tx	rx	tx
	NEIN	NEIN	JA	JA
Bemerkung	Wird intern auf den Parameter oP.28 abgebildet.			

VL_VelocityDeceleration.Dspeed

CAN-SDO-Index	6049h			
SDO-Subindex	1			
Objekttyp	Einfach Variable (Var)			
Datenlänge	4 Byte			
Zugriff	READ_WRITE			
Bedeutung	Gibt zusammen mit VL_VelocityDeceleration.Dtime die Verzögerungsrampe vor.			
Kodierung	1 min-1			
Erlaubte PDO-Abbildung	High-Speed-PDO		Low-Speed-PDO	
	rx	tx	rx	tx
	NEIN	NEIN	JA	JA
Bemerkung	Wird intern auf den Parameter oP.30 abgebildet.			

VL_VelocityDeceleration.Dtime

CAN-SDO-Index	6049h			
SDO-Subindex	2			
Objekttyp	Einfach Variable (Var)			
Datenlänge	2 Byte			
Zugriff	READ_WRITE			
Bedeutung	Gibt zusammen mit VL_VelocityDeceleration.Dspeed die Verzögerungsrampe vor.			
Kodierung	1 s			
Erlaubte PDO-Abbildung	High-Speed-PDO		Low-Speed-PDO	
	rx	tx	rx	tx
	NEIN	NEIN	JA	JA
Bemerkung	Wird intern auf den Parameter oP.30 abgebildet.			

VL_VelocityQuick- Stop.Dspeed

CAN-SDO-Index	604Ah			
SDO-Subindex	1			
Objekttyp	Einfach Variable (Var)			
Datenlänge	4 Byte			
Zugriff	READ_WRITE			
Bedeutung	Gibt zusammen mit VL_VelocityQuickStop.Dtime die Schnellhaltrampe vor.			
Kodierung	1min-1			
Erlaubte PDO-Abbildung	High-Speed-PDO		Low-Speed-PDO	
	rx	tx	rx	tx
	NEIN	NEIN	JA	JA
Bemerkung	Wird intern auf den Parameter Pn.60 abgebildet.			

VL_VelocityQuick- Stop.Dtime

CAN-SDO-Index	604Ah			
SDO-Subindex	2			
Objekttyp	Einfach Variable (Var)			
Datenlänge	2 Byte			
Zugriff	READ_WRITE			
Bedeutung	Gibt zusammen mit VL_VelocityQuickStop.Dspeed die Schnellhaltrampe vor.			
Kodierung	1 s			
Erlaubte PDO-Abbildung	High-Speed-PDO		Low-Speed-PDO	
	rx	tx	rx	tx
	NEIN	NEIN	JA	JA
Bemerkung	Wird intern auf den Parameter Pn.60 abgebildet.			

VL_QuickStopOption-Code

CAN-SDO-Index	605Ah
SDO-Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	2 Byte
Zugriff	READ_WRITE
Bedeutung	Gibt das Verhalten des Schnellhalt vor.
Kodierung	Das DSP402-Profil bestimmt den Wertebereich für herstellereinspezifische Modi in dem Bereich -32768...-1. Aus diesem Grund wird im Operator der Wert auf die interne Kodierung des Parameters Pn.58 wie folgt umgerechnet: $\text{Wert (Pn.58)} = \text{Betrag(VL_QuickStopOptionCode)} - 1$ Die Bedeutung der einzelnen Modi sind der Applikationsanleitung der eingesetzten Umrichtersteuerung zu entnehmen. -1 → Pn.58 = 0 -2 → Pn.58 = 1 -3 → Pn.58 = 2 usw.
Erlaubte PDO-Abbildung	Nicht abbildbar
Bemerkung	Wird intern auf den Parameter Pn.58 abgebildet.

VL_PoleNr

CAN-SDO-Index	604Dh
SDO-Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	1 Byte
Zugriff	READ_ONLY
Bedeutung	Gibt die Polzahl des Motors an. Dieser Wert wird aus den Parametern dr.01 und dr.05 berechnet und für alle Umrechnungen von Drehzahl (min-1) in Frequenzen (Hz) benötigt.
Kodierung	1
Erlaubte PDO-Abbildung	Nicht abbildbar
Bemerkung	Wird intern aus den Parametern dr.01 und dr.05 berechnet.

VL_Ramp_Calc Mode

CAN-SDO-Index	5FDBh
SDO-Subindex	0
Objekttyp	Einfach Variable (Var)
Datenlänge	1 Byte
Zugriff	READ_WRITE
Bedeutung	Bestimmt die Berechnungsweise für die Umrechnung einer KEB-Rampenzeit in eine DSP402-Velocity-Rampe.
Kodierung	<p>0 : Beide Teile der VL-Rampe (Dspeed, Dtime) werden so bestimmt, dass die Werte möglichst klein werden, aber die Genauigkeit der umzusetzenden Rampenzeit erhalten bleibt.</p> <p>1 : Nur VL-Rampe.Dtime wird berechnet, VL-Rampe.Dspeed bleibt unverändert.</p> <p>2 : Als Wert für VL-Rampe.Dtime wird der Wert der KEB-Rampenzeit übernommen. VL-Rampe.Dspeed wird dementsprechend gesetzt.</p>
Erlaubte PDO-Abbildung	Nicht abbildbar
Bemerkung	Eine Wertänderung wird sofort aktiv und nichtflüchtig gespeichert.

Save_VL_Ramps

CAN-SDO-Index	5FDCh																
SDO-Subindex	0																
Objekttyp	Einfach Variable (Var)																
Datenlänge	1 Byte																
Zugriff	READ_WRITE																
Bedeutung	Dient zum nichtflüchtigen Abspeichern der Velocity-Mode Rampen im CAN-Operator.																
Kodierung	<table border="1"> <thead> <tr> <th>B7</th> <th>B6</th> <th>B5</th> <th>B4</th> <th>B3</th> <th>B2</th> <th>B1</th> <th>B0</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td>Save VL-QST.Dtime</td> <td>Save VL-DEC.Dtime</td> <td>Save VL-ACC.Dtime</td> <td>Save VL-QST.Dspeed</td> <td>Save VL-DEC.Dspeed</td> <td>Save VL-ACC.Dspeed</td> </tr> </tbody> </table> <p>Mit VL-ACC.Dspeed : Index=6048h, Subindex= 1 Mit VL-DEC.Dspeed : Index=6049h, Subindex= 1 Mit VL-QST.Dspeed : Index=604Ah, Subindex= 1 Mit VL-ACC.Dtime : Index=6048h, Subindex= 2 Mit VL-DEC.Dtime : Index=6049h, Subindex= 2 Mit VL-QST.Dtime : Index=604Ah, Subindex= 2</p>	B7	B6	B5	B4	B3	B2	B1	B0			Save VL-QST.Dtime	Save VL-DEC.Dtime	Save VL-ACC.Dtime	Save VL-QST.Dspeed	Save VL-DEC.Dspeed	Save VL-ACC.Dspeed
B7	B6	B5	B4	B3	B2	B1	B0										
		Save VL-QST.Dtime	Save VL-DEC.Dtime	Save VL-ACC.Dtime	Save VL-QST.Dspeed	Save VL-DEC.Dspeed	Save VL-ACC.Dspeed										
Erlaubte PDO-Abbildung	Nicht abbildbar																
Bemerkung	Beim Lesen wird immer der Wert 0 zurückgeliefert.																

10. Faktoren

Das DSP402-Profil definiert viele Parameter mit sog. Anwendereinheiten. Um diese Parameter zu realisieren, müssen Umrechnungsfaktoren vorhanden sein, die die Umrechnung in die internen Größen vornehmen. Zu diesem Zweck spezifiziert das Profil eine eigene Gruppe von Parametern die sog. Factor Group. Die KEB-DSP402-Realisierung unterstützt keine Parameter dieser Gruppe. Es werden aber die folgenden Faktoren zur Umrechnung von Einheiten unterstützt, die bei verschiedenen DSP402-Parametern zum Einsatz kommen.

Jeder Faktor besteht aus einem vorzeichenlosen 32-Bit Zähler und einem vorzeichenlosen 32-Bit Nenner. Jeder Faktor ist als Struktur mit drei Mitgliedern definiert (wie unten beschrieben). Die genaue Berechnungsformel wird jeweils bei der Beschreibung der betroffenen Parameter aufgeführt:

- **Faktor 0** : Umrechnung von anwenderspezifischen Wegeinheiten in die vom Frequenzumrichter verwendeten Wegeinheiten.
- **Faktor 1** : Umrechnung von anwenderspezifischen Geschwindigkeitseinheiten in die vom Frequenzumrichter verwendeten Geschwindigkeitseinheiten.
- **Faktor 2** : Umrechnung von anwenderspezifischen Beschleunigungseinheiten in die vom Frequenzumrichter verwendeten Beschleunigungseinheiten.

FactorX.NrEntries

CAN-SDO-Index	5FC0h + X
SDO-Subindex	0
Objektyp	Einfach Variable (Var)
Datenlänge	1 Byte
Zugriff	READ_ONLY
Bedeutung	Anzahl der Mitglieder in der Struktur Faktor X
Kodierung	1
Erlaubte PDO-Abbildung	Nicht abbildbar
Bemerkung	Eine Wertänderung wird sofort aktiv und nichtflüchtig gespeichert.

FactorX.Numerator

CAN-SDO-Index	5FC0h + X
SDO-Subindex	1
Objektyp	Einfach Variable (Var)
Datenlänge	4 Byte
Zugriff	READ_WRITE
Bedeutung	Zählerwert Faktor X
Kodierung	1
Standardeinstellung	1
Erlaubte PDO-Abbildung	Nicht abbildbar
Bemerkung	Eine Wertänderung wird sofort aktiv und nichtflüchtig gespeichert.

FactorX.Divisor

CAN-SDO-Index	5FC0h+X
SDO-Subindex	2
Objekttyp	Einfach Variable (Var)
Datenlänge	4 Byte
Zugriff	READ_WRITE
Bedeutung	Zählerwert Faktor X
Kodierung	1
Standardeinstellung	1
Erlaubte PDO-Abbildung	Nicht abbildbar
Bemerkung	Eine Wertänderung wird sofort aktiv und nichtflüchtig gespeichert.

10.1 Weitergehende Umrechnungen

Für einige Parameter reicht die Umrechnung über einen Faktor bestehend aus Zähler und Nenner nicht aus. Diese Umrechnungen beziehen noch Referenzwerte der FU-Steuerung für die Geschwindigkeit mit ein. Dies ist z.B. erforderlich bei der Umrechnung einer Beschleunigung (Delta Geschwindigkeit / Delta Zeit) in eine Rampenzeit. Die Referenzwerte sind dazu noch abhängig vom F5-Steuerungstyp (siehe auch Beschreibung des Parameters Ud.02 in der Applikationsanleitung der Umrichtersteuerung). Die folgende Liste von Referenzwerten ist im CAN-Operator abgelegt:

Ud.02 - Wert	Geschwindigkeit-Referenzwert (VRef)	Normierung
0	100	Hz
1	200	Hz
2	400	Hz
3	reserviert	reserviert
4	1000	min ⁻¹
5	2000	min ⁻¹
6	4000	min ⁻¹
7	reserviert	reserviert
8	1000	min ⁻¹
9	2000	min ⁻¹
10	4000	min ⁻¹

10.2 Beispiel für die Bestimmung der Faktoren

Generell gilt für die optimale Bestimmung der Faktorenwerte folgendes:

- Begrenzen Sie den Zähler und Nenner der Faktoren, wenn möglich, auf 16-Bit-Breite
- Der lauffzeitintensivste Teil der Berechnung ist die Division durch FactorX.Divisor. Deshalb ist dieser Wert nach Möglichkeit = 1 zu setzen.

10.2.1 Faktor 0: Anwender-Weg-Einheiten in Inkremente

Für diese Umrechnung müssen die Werte der Parameter Ec.01 (Geberstrichzahl Geber 1) bzw. Ec.11 (Geberstrichzahl Geber 2) und Ec.07 (Vielfachauswertung Geber 1) bzw. Ec.17 (Vielfachauswertung Geber 2) bekannt sein. Hier wird von Geber 1 ausgegangen.

Wenn der Faktor 0 bestimmt werden soll für eine Vorgabe in μm (lineare Bewegung), dann gilt folgendes:

$$F0 = \frac{\text{Ec.01} * 2^{\text{Ec.07}}}{U_{\text{Treibscheibe}}} \quad (\text{Umfang der Treibscheibe in } \mu\text{m})$$

$$\rightarrow \text{Factor0.Numerator} = \text{Ec.01} * 2^{\text{Ec.07}}$$

$$\rightarrow \text{Factor0.Divisor} = U_{\text{Treibscheibe}} \quad (\text{Umfang der Treibscheibe in } \mu\text{m})$$

Wenn der Faktor 0 für die Vorgabe in 0,01 Winkelgrad (Drehbewegung) bestimmt werden soll, gilt folgendes:

$$\rightarrow \text{Factor0.Numerator} = \text{Ec.01} * 2^{\text{Ec.07}}$$

$$\rightarrow \text{Factor0.Divisor} = U_{\text{Treibscheibe}} \quad (\text{Umfang der Treibscheibe in } \mu\text{m})$$

$$F0 = \frac{\text{Ec.01} * 2^{\text{Ec.07}}}{36000}$$

$$\rightarrow \text{Factor0.Numerator} = \text{Ec.01} * 2^{\text{Ec.07}}$$

$$\rightarrow \text{Factor0.Divisor} = 36000$$

10.2.2 Faktor 1: Anwender-Geschwindigkeit-Einheiten in $0,125 \text{ min}^{-1}$

Wenn der Faktor 1 für die Vorgabeauflösung $0,1 \text{ min}^{-1}$ bestimmt werden soll, gilt folgendes:

$$\rightarrow \text{Factor1.Numerator} = 8$$

$$\rightarrow \text{Factor1.Divisor} = 10$$

Wenn der Faktor 1 für die Vorgabe $1 \mu\text{m/s}$ bestimmt werden soll, gilt folgendes:

$$\rightarrow \text{Factor1.Numerator} = 480$$

$$\rightarrow \text{Factor1.Divisor} = U_{\text{Treibscheibe}} \quad (\text{Umfang der Treibscheibe in } \mu\text{m})$$

Wenn der Faktor 1 für die Vorgabe $0,01 \text{ Winkelgrad/s}$ bestimmt werden soll, gilt folgendes:

$$\rightarrow \text{Factor1.Numerator} = 8$$

$$\rightarrow \text{Factor1.Divisor} = 600$$

10.2.3 Faktor 2: Anwender-Beschleunigung-Einheiten in eine KEB-Rampenzeit

Die Bestimmung des Faktors 2 ist etwas schwieriger. Beachten Sie, dass bei KEB die Beschleunigungs- / Verzögerungsparameter als Rampenzeit definiert sind. Deshalb ist die Umrechnung in diesem Fall umfangreicher. Hier soll der Faktor am Beispiel des Parameters HM_Homing_Acc gezeigt werden, der auf den Parameter PS.20 in der F5-Steuerung abgebildet wird:

Zur Vereinfachung kann man den Faktor 2 zunächst als ein Element schreiben. Dann sieht die Berechnungsformel wie folgt aus:

$$\text{PS.20} = \frac{V_{\text{ref}}}{\text{HM_Homing_Acc}} * \text{Faktor 2}$$

Umgestellt nach Faktor 2:

$$\text{Faktor 2} = \frac{\text{PS.20} * \text{HM_Homing_Acc}}{V_{\text{ref}}}$$

Angenommen:

- Ud.02 = 4: F5-M, Maximaldrehzahl = 4000 min⁻¹ , Vref = 1000 min⁻¹
- Der Parameter HM_Homing_Acc soll in min⁻² aufgelöst sein

Der Wert PS.20 = 100 bedeutet eine Beschleunigung von 1000 min⁻¹ pro Sekunde. Dies entspricht einem Wert von 60000 min⁻². Wenn man in die obige Gleichung jetzt für PS.20 den Wert 100 und für HM_Homing_Acc den Wert 60000 einsetzt, ergibt sich für Faktor 2:

$$\text{Faktor 2} = \frac{100 * 60000}{1000} = 6000$$

→ Factor2.Numerator = 6000

→ Factor2.Divisor = 1

11. Anhang

11.1 CAN-Bit-Timing

Die KEB-CAN-Anschaltung(en) halten sich bzgl. des eingestellten Bit-Timings an die Vorgaben des CiA-Standards [2]:

Das nominale Bit-Timing sieht wie folgt aus:

SYNC	TSEG1	TSEG2
Samplepoint		

Für alle einstellbaren Baudraten gilt:

- t_q = Basiszeiteinheit. Alle Segmente des Bit-Timing ergeben sich als Vielfaches dieser Zeiteinheit.
- SYNC = 0 → Nur die Flanken von Rezessiv zu Dominant werden zur Synchronisation genutzt.
- SJW = 0 → Synchronisationssprungweite = $1 * t_q$
- TSEG2 = 1 → $t_{SEG2} = 2 * t_q$

Baudrate	Timequantum (t_q)	Samplepoint	TSEG1
10 Kbit/s	6,25 μ s	$14 * t_q = 87,5 \mu$ s	$t_{SEG1} = 13 * t_q$
20 Kbit/s	3,125 μ s	$14 * t_q = 43,75 \mu$ s	$t_{SEG1} = 13 * t_q$
25 Kbit/s	2,5 μ s	$14 * t_q = 35,0 \mu$ s	$t_{SEG1} = 13 * t_q$
50 Kbit/s	1,25 μ s	$14 * t_q = 17,5 \mu$ s	$t_{SEG1} = 13 * t_q$
100 Kbit/s	625 ns	$14 * t_q = 8,75 \mu$ s	$t_{SEG1} = 13 * t_q$
125 Kbit/s	500 ns	$14 * t_q = 7,0 \mu$ s	$t_{SEG1} = 13 * t_q$
250 Kbit/s	250 ns	$14 * t_q = 3,5 \mu$ s	$t_{SEG1} = 13 * t_q$
500 Kbit/s	125 ns	$13 * t_q = 1,625 \mu$ s	$t_{SEG1} = 12 * t_q$
800 Kbit/s	125 ns	$7 * t_q = 1,25 \mu$ s	$t_{SEG1} = 6 * t_q$
1000 Kbit/s	125 ns	$5 * t_q = 625$ ns	$t_{SEG1} = 4 * t_q$

Die in der Tabelle grau hinterlegten Übertragungsgeschwindigkeiten sind bezüglich der möglichen Leitungslänge als besonders kritisch anzusehen. Außerdem weicht das Bit-Timing für diese Bitraten leicht von dem von [2] empfohlenen ab.

11.1.1 Wichtiger Warnhinweis

	<p>Die KEB-CAN-Anschaltung besitzt eine potentialgetrennte CAN-Schnittstelle. Durch die zusätzlichen Verzögerungsglieder (Optokoppler) im Signalverlauf verringert sich die mögliche Leitungslänge oder die mögliche Übertragungsgeschwindigkeit. Welche Leitungslänge bzw. Übertragungsgeschwindigkeit möglich ist, hängt von den Verzögerungszeiten aller im CAN-Netzwerk betriebenen Teilnehmer ab. Es obliegt dem Kunden eine Abschätzung vorzunehmen, bei welcher Bitrate welche Leitungslängen möglich sind. Die dazu notwendigen Angaben für die KEB-CAN-Anschaltung werden unten aufgeführt:</p>
---	--

Sende-Verzögerungszeit des CAN-Treibers	: ≤	80 ns.
Empfangs-Verzögerungszeit des CAN-Treibers	: ≤	70 ns.
Sende-Verzögerungszeit der eingesetzten Optokoppler	: ≤	40 ns.
Empfangs-Verzögerungszeit der eingesetzten Optokoppler	: ≤	40 ns.

Es sollte immer die kleinste CAN-Übertragungsgeschwindigkeit gewählt werden, die von der Abarbeitung des Prozesses her erforderlich ist.

11.2 Literaturverzeichnis

- [1] : Betriebsanleitung Frequenzumrichtersteuerung KEB COMBIVERT F5 mit Applikationsanleitung.
- [2] : Dokument zur Vereinbarung des Arbeitskreises Physical-Layer der CAN in Automation (CiA) Nutzergruppe: CiA/DS 102-1. Herausgeber: CiA International Users and Manufacturers Group e.V., Am Weichselgarten 26, D-91058 Erlangen. Dokumente zur Vereinbarung des Arbeitskreises Higher-Layer-Protocols der CiA (Herausgeber s.o.):
- [3] : CiA/WG2/DS201 : CAN in the OSI Reference Model
- [4] : CiA/WG2/DS202-1 : CMS Service Specification
- [5] : CiA/WG2/DS202-2 : CMS Protocol Specification
- [6] : CiA/WG2/DS202-3 : CMS Encoding Rules
- [7] : CiA/WG2/DS203-1 : NMT Service Specification
- [8] : CiA/WG2/DS203-2 : NMT Protocol Specification
- [9] : CiA/WG2/DS204-1 : DBT Service Specification
- [10] : CiA/WG2/DS204-2 : DBT Protocol Specification
- [11] : CiA/WG2/DS207 : Application Layer Naming Conventions
- [12] : CiA/DS301 V. 4.01 : Application Layer and Communication Profile vom 01.06.2000
- [13] : CiA/DSP402 V.2.0 : Device Profile Drives and Motion Control

11.3 Übersicht der Operatorparameter nach CANopen

Index	Name	Objekttyp	Subindex	Datenlänge in Byte	Zugriff
1000h	Device type	VAR	0	4	ro
1001h	Error register	VAR	0	1	ro
1002h	Manufacturer Status Register	VAR	0	4	ro
1003h	Predefined ErrorField	ARRAY	1 bis max. 5	4	rw
1006h	ComCycle	VAR	0	4	rw
1008h	Manufacturer Device Name	VAR	0	4	ro
1018h	Identify object	RECORD			
1400h	1st receive PDO Parameter	RECORD			
1400h	Number of supported entries	VAR	0	1	ro
1400h	COB-ID	VAR	1	4	rw
1400h	transmission type	VAR	2	1	rw
1401h	2nd receive PDO Parameter	RECORD			
1401h	Number of supported entries	VAR	0	1	ro
1401h	COB-ID	VAR	1	4	rw
1401h	transmission type	VAR	2	1	rw
1600h	1st receive PDO Mapping	RECORD			
1600h	Number of mapped objects	VAR	0	1	rw
1600h	nth object to be mapped	VAR	1 bis max. 4	4	rw
1601h	2nd receive PDO Mapping	RECORD			
1601h	Number of mapped objects	VAR	0	1	rw
1601h	nth object to be mapped	VAR	1 bis max. 4	4	rw
1800h	1st transmit PDO Parameter	RECORD			
1800h	Number of supported entries	VAR	0	1	ro

weiter auf nächster Seite

Index	Name	Objekttyp	Subindex	Datenlänge in Byte	Zugriff
1800h	COB-ID	VAR	1	4	rw
1800h	transmission type	VAR	2	1	rw
1800h	Inhibit time	VAR	3	2	rw
1801h	2nd transmit PDO Parameter	RECORD			
1801h	Number of supported entries	VAR	0	1	ro
1801h	COB-ID	VAR	1	4	rw
1801h	transmission type	VAR	2	1	rw
1801h	Inhibit time	VAR	3	2	rw
100Ah	Manufacturer software version	VAR	0	4	ro
100Ch	Guard Time	VAR	0	2	rw
100Dh	Life Time Factor	VAR	0	1	rw
1A00h	1st transmit PDO Mapping	RECORD			
1A00h	Number of mapped objects	VAR	0	1	rw
1A00h	nth object to be mapped	VAR	1 bis max. 4	4	rw
1A01h	2nd transmit PDO Mapping	RECORD			
1A01h	number of mapped objects	VAR	0	1	rw
1A01h	nth objects to be mapped	VAR	1 bis max. 4	4	rw
5FDAh	Watchdog activation	VAR	0	1	rw
5FDBh	VL_Ramp_CalcMode	VAR	0	1	rw
5FDCh	Save_VL_Ramps	VAR	0	1	rw
5FDDh	EmergencyCycle	VAR	0	2	rw
5FDEh	LifeGuardTout.Data	VAR	0	4	rw
5FDFh	LifeGuardTout.Addr	VAR	0	4	rw
5FE0h	HS_SyncToutDelay	VAR	0	2	rw
5FE2h	PD_Stored	VAR	0	1	rw
5FE3h	OP_NoId	VAR	0	1	rw
5FE4h	PDOOUT_WrMode	VAR	0	1	rw
5FE5h	HS_PDO_Index	VAR	0	1	rw
5FE6h	PDIN1_Cycle_Time	VAR	0	2	rw
5FE7h	PDIN2_Cycle_Time	VAR	0	2	rw
5FF9h	Watchdog inhibit	VAR	0	1	rw
5FECh	CAN_Baud2	VAR	0	1	rw
5FFEh	SAVE_CAN_Baud	VAR	0	1	rw
5FFFh	CAN_Baud	VAR	0	1	rw

11.4 Kompakt-Übersicht der CAN-Kommunikation

Feste Identifizierung:

- SDO(rx)-Identifizierung = 1536 + Node_Id : SDO-Anforderung zum KEB-FU
- SDO(tx)-Identifizierung = 1408 + Node_Id : SDO-Bestätigung vom KEB-FU
- PDO1(rx)-Identifizierung = 512 + Node_Id : Prozessdaten zum KEB-FU
- PDO1(tx)-Identifizierung = 384 + Node_Id : Prozessdaten vom KEB-FU
- PDO2(rx)-Identifizierung = 768 + Node_Id : Prozessdaten zum KEB-FU
- PDO2(tx)-Identifizierung = 640 + Node_Id : Prozessdaten vom KEB-FU
- Node-Guarding-Identifizierung = 1792 + Node_Id
- Emergency-Identifizierung = 128 + Node_Id : Emergency Message vom KEB-FU

Die wichtigsten NMT-Kommandos (Telegramme) auf Identifizierung = 0

Start_Remote_Node		Enter_Pre_Operational_State		Reset_Node	
01h	Node-Id	80h	Node-Id	81h	Node-Id
B0	B1	B0	B1	B0	B1

Die wichtigsten Werte des Node-State:

- PRE_ OPERATIONAL = 7Fh : Kommunikation aktiv bis auf die PDO's
- OPERATIONAL = 05h : Kommunikation komplett aktiv

Beispiele für SDO-Kommunikation:

Lesen des Parameters Digitale Sollfrequenzvorgabe (oP.03) in Satz 4 → Index = 2303h, Subindex = 10h

SDO(rx)-Telegramm (zum KEB-FU)								SDO(tx)-Telegramm (vom KEB-FU)							
40h	03h	23h	10h	XXh	XXh	XXh	XXh	43h	03h	23h	10h	E8h	03h	00h	00h
B0	B1	B2	B3	B4	B5	B6	B7	B0	B1	B2	B3	B4	B5	B6	B7

In diesem Beispiel ist der gelesene Wert = 1000 (03E8h)

Schreiben Wert = 5 auf den Parameter Sollwertquelle (oP.00) in Satz 0 → Index = 2300h, Subindex = 01h

SDO(rx)-Telegramm (zum KEB-FU)								SDO(tx)-Telegramm (vom KEB-FU)							
22h	00h	23h	01h	05h	00h	00h	00h	60h	00h	23h	01h	05h	00h	00h	00h
B0	B1	B2	B3	B4	B5	B6	B7	B0	B1	B2	B3	B4	B5	B6	B7

Beispiel für die Vorgabe von neuen Prozessdaten mit dem PDO1(rx)-Telegramm:

Vorausgesetzt ist hier die Standardprozessdatenbelegung.

- Der Parameter Steuerwort (Sy.50) soll den Wert = 1 erhalten,
- der Parameter Solldrehzahl (Sy.52) den Wert = 1500 (05DCh).

PDO1(rx)-Telegramm (zum KEB-FU)

01h	00h	DCh	05h
-----	-----	-----	-----

11.5 F5 Operator interne Fehlermeldungen

Allgemeine Fehlermeldungen

Error	Übertragungsstörung während der Initialisierung
o_Flo	Überlauf bei einer Werteberechnung
t_out	Timeout, Steuerkarte hat nicht geantwortet
IDAtA	Daten ungültig
rOnly	Parameter schreibgeschützt
E_Bcc	Übertragungsstörung: Checksumme falsch
Busy	Umrichter beschäftigt
ISruc	Übertragungsstörung: Dienst ungültig
No PA	Parameter passwordgeschützt
I_FrA	Übertragungsstörung: Zeichen ungültig
E_PAr	Übertragungsstörung: Parität falsch
I_SEt	Satz ungültig
I_Adr	Adresse ungültig
I_OPE	Operation ungültig
E xx	xx=Hexadezimaler Fehlercode : alle anderen Fehler
EEEPX	mit X = 1,2,3,...: Fehler beim Test des seriellen EEPROMs
EEEEPR	Fehler beim Test des seriellen EEPROMs

Spezielle Fehlermeldungen des CAN Operator

ECAXX	mit XX = 00,01,02,...: Fehler beim Test des externen CAN-Controllers
Synch	Operator befindet sich im Synchron-Modus
SAuto	Operator befindet sich im Auto-Synchron-Modus. D.h. Die SYNC-Telegramme werden auf CAN nicht empfangen, der Operator generiert sich selbst den Synchron-Modus-Takt, damit die Kommunikation zum FU erhalten bleibt.



KEB Automation KG

Südstraße 38 • 32683 Barntrup
fon: +49 5263 401-0 • fax: +49 5263 401-116
net: www.keb.de • mail: info@keb.de

KEB worldwide...

KEB Antriebstechnik Austria GmbH

Ritzstraße 8 • 4614 Marchtrenk
fon: +43 7243 53586-0 • fax: +43 7243 53586-21
net: www.keb.at • mail: info@keb.at

KEB Antriebstechnik

Herenveld 2 • 9500 Geraadsbergen
fon: +32 5443 7860 • fax: +32 5443 7898
mail: vb.belgien@keb.de

KEB Power Transmission Technology (Shanghai) Co.,Ltd.

No. 435 Qianpu Road, Chedun Town, Songjiang District,
Shanghai 201611, P.R. China
fon: +86 21 37746688 • fax: +86 21 37746600
net: www.keb.de • mail: info@keb.cn

KEB Antriebstechnik Austria GmbH

Organizační složka
Suchovrbsenske nam. 2724/4 • 370 06 České Budějovice
fon: +420 387 699 111 • fax: +420 387 699 119
mail: info@keb.cz

KEB Antriebstechnik GmbH

Wildbacher Str. 5 • 08289 Schneeberg
fon: +49 3772 67-0 • fax: +49 3772 67-281
mail: info@keb-drive.de

KEB España

C/ Mitjer, Nave 8 - Pol. Ind. LA MASIA
08798 Sant Cugat Sesgarrigues (Barcelona)
fon: +34 93 897 0268 • fax: +34 93 899 2035
mail: vb.espana@keb.de

Société Française KEB

Z.I. de la Croix St. Nicolas • 14, rue Gustave Eiffel
94510 LA QUEUE EN BRIE
fon: +33 1 49620101 • fax: +33 1 45767495
net: www.keb.fr • mail: info@keb.fr

KEB (UK) Ltd.

Morris Close, Park Farm Industrial Estate
Wellingborough, NN8 6 XF
fon: +44 1933 402220 • fax: +44 1933 400724
net: www.keb.co.uk • mail: info@keb.co.uk

KEB Italia S.r.l.

Via Newton, 2 • 20019 Settimo Milanese (Milano)
fon: +39 02 3353531 • fax: +39 02 33500790
net: www.keb.de • mail: kebitalia@keb.it

KEB Japan Ltd.

15-16, 2-Chome, Takanawa Minato-ku
Tokyo 108-0074
fon: +81 33 445-8515 • fax: +81 33 445-8215
mail: info@keb.jp

KEB Korea Seoul

Room 1709, 415 Missy 2000
725 Su Seo Dong, Gang Nam Gu
135-757 Seoul/South Korea
fon: +82 2 6253 6771 • fax: +82 2 6253 6770
mail: vb.korea@keb.de

KEB RUS Ltd.

Lesnaya Str. House 30, Dzerzhinsky (MO)
140091 Moscow region
fon: +7 495 632 0217 • fax: +7 495 632 0217
net: www.keb.ru • mail: info@keb.ru

KEB America, Inc.

5100 Valley Industrial Blvd. South
Shakopee, MN 55379
fon: +1 952 224-1400 • fax: +1 952 224-1499
net: www.kebamerica.com • mail: info@kebamerica.com

More and latest addresses at <http://www.keb.de>

© KEB		
Document	20103622	
Part/Version	DEU	01
Date	2018-02-13	