

System 300S

CP | 342-1CA70 | Handbuch

HB140 | CP | 342-1CA70 | de | 17-16

SPEED7 CP 342S-CAN



YASKAWA Europe GmbH
Hauptstraße 185
65760 Eschborn
Deutschland
Tel.: +49 6196 569-300
Fax: +49 6196 569-398
E-Mail: info@yaskawa.eu.com
Internet: www.yaskawa.eu.com

Inhaltsverzeichnis

| | | |
|----------|--|-----------|
| 1 | Allgemeines | 4 |
| 1.1 | Copyright © YASKAWA Europe GmbH..... | 4 |
| 1.2 | Über dieses Handbuch..... | 5 |
| 1.3 | Sicherheitshinweise..... | 6 |
| 2 | Grundlagen | 7 |
| 2.1 | Sicherheitshinweis für den Benutzer..... | 7 |
| 2.2 | Hinweise zur Projektierung..... | 8 |
| 2.3 | Allgemeine Daten..... | 11 |
| 2.3.1 | Einsatz unter erschwerten Betriebsbedingungen..... | 12 |
| 3 | Montage und Aufbaurichtlinien | 13 |
| 3.1 | Übersicht..... | 13 |
| 3.2 | Einbaumaße..... | 14 |
| 3.3 | Montage SPEED-Bus..... | 15 |
| 3.4 | Aufbaurichtlinien..... | 19 |
| 4 | Hardwarebeschreibung | 22 |
| 4.1 | Leistungsmerkmale..... | 22 |
| 4.2 | Aufbau..... | 23 |
| 4.3 | Technische Daten..... | 27 |
| 5 | Einsatz | 29 |
| 5.1 | Grundlagen CAN..... | 29 |
| 5.2 | Adressierung am SPEED-Bus..... | 31 |
| 5.3 | Projektierung Schnelleinstieg..... | 32 |
| 5.4 | Projektierung..... | 33 |
| 5.5 | Betriebsarten..... | 41 |
| 5.6 | Prozessabbild..... | 42 |
| 5.7 | Telegrammaufbau..... | 43 |
| 5.8 | Objekt-Verzeichnis..... | 45 |
| 5.9 | Diagnose..... | 67 |
| 5.9.1 | Aufbau der Diagnosedaten..... | 67 |
| 5.10 | SZL auslesen..... | 73 |
| 5.10.1 | SFC 51 - RDSYSST - Auslesen der Informationen der SZL..... | 73 |
| 5.10.2 | SZL-Listen des CAN-Masters..... | 74 |
| 5.11 | Station (de-)aktivieren..... | 76 |

1 Allgemeines

1.1 Copyright © YASKAWA Europe GmbH

All Rights Reserved

Dieses Dokument enthält geschützte Informationen von Yaskawa und darf außer in Übereinstimmung mit anwendbaren Vereinbarungen weder offengelegt noch benutzt werden.

Dieses Material ist durch Urheberrechtsgesetze geschützt. Ohne schriftliches Einverständnis von Yaskawa und dem Besitzer dieses Materials darf dieses Material weder reproduziert, verteilt, noch in keiner Form von keiner Einheit (sowohl Yaskawa-intern als auch -extern) geändert werden, es sei denn in Übereinstimmung mit anwendbaren Vereinbarungen, Verträgen oder Lizenzen.

Zur Genehmigung von Vervielfältigung oder Verteilung wenden Sie sich bitte an:
YASKAWA Europe GmbH, European Headquarters, Hauptstraße 185, 65760 Eschborn, Germany

Tel.: +49 6196 569 300

Fax.: +49 6196 569 398

E-Mail: info@yaskawa.eu.com

Internet: www.yaskawa.eu.com



Es wurden alle Anstrengungen unternommen, um sicherzustellen, dass die in diesem Dokument enthaltenen Informationen zum Zeitpunkt der Veröffentlichung vollständig und richtig sind. Das Recht auf Änderungen der Informationen bleibt jedoch vorbehalten.

Die vorliegende Kundendokumentation beschreibt alle heute bekannten Hardware-Einheiten und Funktionen. Es ist möglich, dass Einheiten beschrieben sind, die beim Kunden nicht vorhanden sind. Der genaue Lieferumfang ist im jeweiligen Kaufvertrag beschrieben.

EG-Konformitätserklärung

Hiermit erklärt YASKAWA Europe GmbH, dass die Produkte und Systeme mit den grundlegenden Anforderungen und den anderen relevanten Vorschriften übereinstimmen. Die Übereinstimmung ist durch CE-Zeichen gekennzeichnet.

Informationen zur Konformitätserklärung

Für weitere Informationen zur CE-Kennzeichnung und Konformitätserklärung wenden Sie sich bitte an Ihre Landesvertretung der YASKAWA Europe GmbH.

Warenzeichen

VIPA, SLIO, System 100V, System 200V, System 300V, System 300S, System 400V, System 500S und Commander Compact sind eingetragene Warenzeichen der YASKAWA Europe GmbH.

SPEED7 ist ein eingetragenes Warenzeichen der YASKAWA Europe GmbH.

SIMATIC, STEP, SINEC, TIA Portal, S7-300, S7-400 und S7-1500 sind eingetragene Warenzeichen der Siemens AG.

Microsoft und Windows sind eingetragene Warenzeichen von Microsoft Inc., USA.

Portable Document Format (PDF) und Postscript sind eingetragene Warenzeichen von Adobe Systems, Inc.

Alle anderen erwähnten Firmennamen und Logos sowie Marken- oder Produktnamen sind Warenzeichen oder eingetragene Warenzeichen ihrer jeweiligen Eigentümer.

Dokument-Support

Wenden Sie sich an Ihre Landesvertretung der YASKAWA Europe GmbH, wenn Sie Fehler anzeigen oder inhaltliche Fragen zu diesem Dokument stellen möchten. Sie können YASKAWA Europe GmbH über folgenden Kontakt erreichen:

E-Mail: Documentation.HER@yaskawa.eu.com

Technischer Support

Wenden Sie sich an Ihre Landesvertretung der YASKAWA Europe GmbH, wenn Sie Probleme mit dem Produkt haben oder Fragen zum Produkt stellen möchten. Ist eine solche Stelle nicht erreichbar, können Sie den Yaskawa Kundenservice über folgenden Kontakt erreichen:

YASKAWA Europe GmbH,
European Headquarters, Hauptstraße 185, 65760 Eschborn, Germany
Tel.: +49 6196 569 500 (Hotline)
E-Mail: support@yaskawa.eu.com

1.2 Über dieses Handbuch

Zielsetzung und Inhalt

Das Handbuch beschreibt den CP 342-1CA70 aus dem System 300S von Yaskawa. Beschrieben wird Aufbau, Projektierung und Anwendung.

| Produkt | Best.-Nr. | ab Stand: | |
|-------------|-----------|-----------|--------|
| | | CP-HW | CP-FW |
| CP 342S-CAN | 342-1CA70 | 1 | V1.2.5 |

Zielgruppe

Das Handbuch ist geschrieben für Anwender mit Grundkenntnissen in der Automatisierungstechnik.

Aufbau des Handbuchs

Das Handbuch ist in Kapitel gegliedert. Jedes Kapitel beschreibt eine abgeschlossene Thematik.

Orientierung im Dokument

Als Orientierungshilfe stehen im Handbuch zur Verfügung:

- Gesamt-Inhaltsverzeichnis am Anfang des Handbuchs
- Verweise mit Seitenangabe

Verfügbarkeit

Das Handbuch ist verfügbar in:

- gedruckter Form auf Papier
- in elektronischer Form als PDF-Datei (Adobe Acrobat Reader)

Piktogramme Signalwörter

Wichtige Textteile sind mit folgenden Piktogrammen und Signalworten hervorgehoben:

**GEFAHR!**

Unmittelbare oder drohende Gefahr. Personenschäden sind möglich.

**VORSICHT!**

Bei Nichtbefolgen sind Sachschäden möglich.



Zusätzliche Informationen und nützliche Tipps.

1.3 Sicherheitshinweise

Bestimmungsgemäße Verwendung

Das System ist konstruiert und gefertigt für:

- Kommunikation und Prozesskontrolle
- Allgemeine Steuerungs- und Automatisierungsaufgaben
- den industriellen Einsatz
- den Betrieb innerhalb der in den technischen Daten spezifizierten Umgebungsbedingungen
- den Einbau in einen Schaltschrank



GEFAHR!

Das Gerät ist nicht zugelassen für den Einsatz

- in explosionsgefährdeten Umgebungen (EX-Zone)

Dokumentation

Handbuch zugänglich machen für alle Mitarbeiter in

- Projektierung
- Installation
- Inbetriebnahme
- Betrieb



VORSICHT!

Vor Inbetriebnahme und Betrieb der in diesem Handbuch beschriebenen Komponenten unbedingt beachten:

- Änderungen am Automatisierungssystem nur im spannungslosen Zustand vornehmen!
- Anschluss und Änderung nur durch ausgebildetes Elektro-Fachpersonal
- Nationale Vorschriften und Richtlinien im jeweiligen Verwenderland beachten und einhalten (Installation, Schutzmaßnahmen, EMV ...)

Entsorgung

Zur Entsorgung des Geräts nationale Vorschriften beachten!

2 Grundlagen

2.1 Sicherheitshinweis für den Benutzer

Handhabung elektrostatisch gefährdeter Baugruppen

Die Baugruppen sind mit hochintegrierten Bauelementen in MOS-Technik bestückt. Diese Bauelemente sind hoch empfindlich gegenüber Überspannungen, die z.B. bei elektrostatischer Entladung entstehen. Zur Kennzeichnung dieser gefährdeten Baugruppen wird nachfolgendes Symbol verwendet:



Das Symbol befindet sich auf Baugruppen, Baugruppenträgern oder auf Verpackungen und weist so auf elektrostatisch gefährdete Baugruppen hin. Elektrostatisch gefährdete Baugruppen können durch Energien und Spannungen zerstört werden, die weit unterhalb der Wahrnehmungsgrenze des Menschen liegen. Hantiert eine Person, die nicht elektrisch entladen ist, mit elektrostatisch gefährdeten Baugruppen, können Spannungen auftreten und zur Beschädigung von Bauelementen führen und so die Funktionsweise der Baugruppen beeinträchtigen oder die Baugruppe unbrauchbar machen. Auf diese Weise beschädigte Baugruppen werden in den wenigsten Fällen sofort als fehlerhaft erkannt. Der Fehler kann sich erst nach längerem Betrieb einstellen. Durch statische Entladung beschädigte Bauelemente können bei Temperaturänderungen, Erschütterungen oder Lastwechseln zeitweilige Fehler zeigen. Nur durch konsequente Anwendung von Schutzmaßnahmen und verantwortungsbewusste Beachtung der Handlungsregeln lassen sich Funktionsstörungen und Ausfälle an elektrostatisch gefährdeten Baugruppen wirksam vermeiden.

Versenden von Baugruppen

Verwenden Sie für den Versand immer die Originalverpackung.

Messen und Ändern von elektrostatisch gefährdeten Baugruppen

Bei Messungen an elektrostatisch gefährdeten Baugruppen sind folgende Dinge zu beachten:

- Potenzialfreie Messgeräte sind kurzzeitig zu entladen.
- Verwendete Messgeräte sind zu erden.

Bei Änderungen an elektrostatisch gefährdeten Baugruppen ist darauf zu achten, dass ein geerdeter Lötkolben verwendet wird.



VORSICHT!

Bei Arbeiten mit und an elektrostatisch gefährdeten Baugruppen ist auf ausreichende Erdung des Menschen und der Arbeitsmittel zu achten.

2.2 Hinweise zur Projektierung

Übersicht

Die Projektierung eines SPEED7-Systems sollte nach folgender Vorgehensweise erfolgen:

- Projektierung der SPEED7-CPU und des internen DP-Masters (falls vorhanden)
- Projektierung der reell gesteckten Module am Standard-Bus
- Projektierung des internen Ethernet-PG/OP-Kanals nach den reell gesteckten Modulen als virtueller CP 343-1 (Angabe von IP-Adresse, Subnetz-Maske und Gateway für Online-Projektierung)
- Projektierung eines internen CP 343 (falls vorhanden) als 2. CP 343-1
- Projektierung und Vernetzung aller SPEED-Bus-CPs bzw. -DP-Master als CP 343-1 (343-1EX11) bzw. CP 342-5 (342-5DA02 V5.0)
- Projektierung aller SPEED-Bus-Module als einzelne DP-Slaves in einem virtuellen DP-Master-Modul (SPEEDBUS.GSD erforderlich)



Bitte verwenden Sie zur Projektierung einer CPU 31xS von Yaskawa immer die entsprechende Siemens CPU aus dem Hardware-Katalog. Zur Projektierung werden fundierte Kenntnisse im Umgang mit dem Siemens SIMATIC Manager und dem Hardware-Konfigurator von Siemens vorausgesetzt!

Voraussetzung

Der Hardware-Konfigurator ist Bestandteil des Siemens SIMATIC Managers. Er dient der Projektierung. Die Module, die hier projiziert werden können, entnehmen Sie dem Hardware-Katalog. Für den Einsatz der System 300S Module am SPEED-Bus ist die Einbindung der System 300S Module über die GSD-Datei SPEEDBUS.GSD von Yaskawa im Hardwarekatalog erforderlich.

Vorgehensweise

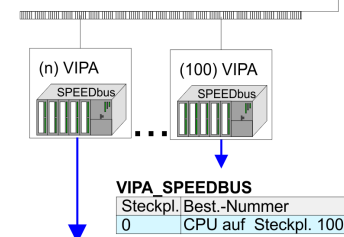
Standard-Bus

| Steckpl. | Modul |
|----------|----------------|
| 1 | |
| 2 | CPU ... |
| X... | ... |
| X... | ... |
| 3 | |

reelle Module am Standard-Bus

| |
|---|
| 343-1EX11 (PG/OP) |
| 343-1EX11 (nur CPU 31xSN) |
| CP bzw. DP-Master am SPEED-Bus als 343-1EX11 bzw. 342-5DA02 |
| 342-5DA02 V5.0 |

virtueller DP-Master für CPU und alle SPEED-Bus-Module



| Steckpl. | Best.-Nummer |
|----------|----------------------|
| 0 | Module v. Steckpl. n |

Die Projektierung einer SPEED7-CPU besteht aus folgenden Komponenten. Um kompatibel mit dem Siemens SIMATIC Manager zu sein, sind folgende Schritte durchzuführen:

1. Vorbereitung

Starten Sie den Hardware-Konfigurator von Siemens und binden Sie die SPEEDBUS.GSD für den SPEED-Bus von Yaskawa ein.

2. Projektierung der CPU

Projektieren Sie die entsprechende CPU. Sofern Ihre SPEED7-CPU einen DP-Master besitzt, können Sie diesen jetzt mit PROFIBUS vernetzen und Ihre DP-Slaves anbinden.

3. Projektierung der reell gesteckten Module am Standard-Bus

Platzieren Sie ab Steckplatz 4 die Module, die sich auf dem Standard-Bus rechts der CPU befinden.

4. Projektierung der integrierten CPs

Für den internen Ethernet-PG/OP-Kanal ist immer als 1. Modul unter den reell gesteckten Modulen ein CP 343-1 (343-1EX11) zu platzieren. Hat Ihre SPEED7-CPU zusätzlich einen CP 343 integriert, so ist dieser ebenfalls als CP 343-1 aber immer unterhalb des zuvor platzierten CP 343-1 zu projektieren.

5. Projektierung aller SPEED-Bus-CPs und -DP-Master

Platzieren und vernetzen Sie unter den zuvor projektieren internen CPU-Komponenten alle CPs als 343-1EX11 und DP-Master als 342-5DA02 V5.0, die sich am SPEED-Bus befinden.

i Bitte beachten Sie, dass die Reihenfolge innerhalb einer Funktionsgruppe (CP bzw. DP-Master) der Reihenfolge am SPEED-Bus von rechts nach links entspricht.

6. Projektierung der CPU und aller SPEED-Bus-Module in einem virtuellen Master-System

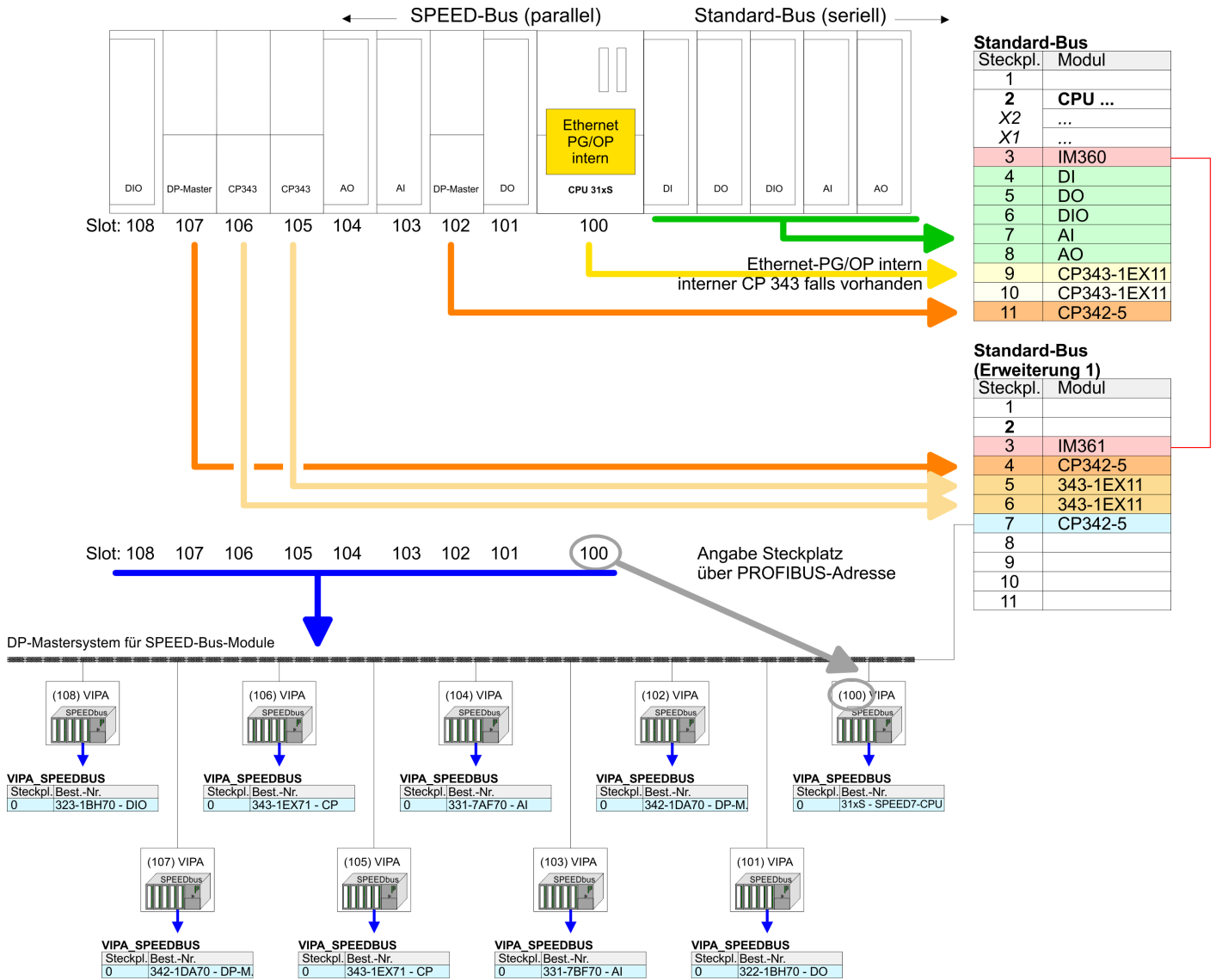
Die Steckplatzzuordnung der SPEED-Bus-Module und die Parametrierung der Ein-/Ausgabe-Peripherie hat über ein virtuelles PROFIBUS-DP-Master-System zu erfolgen. Platzieren Sie hierzu als letztes Modul einen DP-Master (342-5DA02 V5.0) mit Mastersystem. Die PROFIBUS Adresse muss hierbei < 100 sein! Binden Sie nun für die CPU und jedes Modul am SPEED-Bus den Slave "VIPA_SPEEDBUS" an. Nach der Installation der SPEEDBUS.GSD finden Sie diesen unter Profibus-DP / Weitere Feldgeräte / I/O / VIPA_SPEEDBUS. Stellen Sie als PROFIBUS Adresse die Steckplatz-Nr. (100...110) des Moduls ein und platzieren Sie auf dem einzigen Steckplatz 0 des Slave-Systems das entsprechende Modul.

Buserweiterung mit IM 360 und IM 361

Zur Buserweiterung können Sie die IM 360 von Siemens einsetzen, an die Sie bis zu 3 Erweiterungs-Racks über die IM 361 anbinden können. Buserweiterungen dürfen immer nur auf Steckplatz 3 platziert werden. Näheres hierzu finden im Teil "Einsatz CPU 31xS" unter "Adressierung".

Zusammenfassung

In der nachfolgenden Abbildung sind alle Projektierschritte nochmals zusammengefasst:



Das entsprechende Modul ist aus dem HW-Katalog von VIPA_SPEEDBUS auf Steckplatz 0 zu übernehmen



Die Reihenfolge der DPM- und CP-Funktionsgruppen ist unerheblich. Es ist lediglich darauf zu achten, dass innerhalb einer Funktionsgruppe die Reihenfolge (DP1, DP2 ... bzw. CP1, CP2 ...) eingehalten wird.



Hinweis gültig für alle SPEED-Bus-Module!

Für den SPEED-Bus ist immer als letztes Modul der Siemens DP-Master CP 342-5 (342-5DA02 V5.0) einzubinden, zu vernetzen und in die Betriebsart DP-Master zu parametrieren. An dieses Mastersystem ist jedes einzelne SPEED-Bus-Modul als VIPA_SPEED-Bus-Slave anzubinden. Durch Angabe der SPEED-Bus-Steckplatz-Nr. über die PROFIBUS-Adresse und durch Einbinden des entsprechenden SPEED-Bus-Moduls auf dem einzigen Steckplatz 0 erhält der Siemens SIMATIC Manager so Informationen über die am SPEED-Bus befindlichen Module.

2.3 Allgemeine Daten

Konformität und Approbation

| | | |
|-------------|------------|---|
| Konformität | | |
| CE | 2014/35/EU | Niederspannungsrichtlinie |
| | 2014/30/EU | EMV-Richtlinie |
| Approbation | | |
| UL | | Siehe Technische Daten |
| Sonstiges | | |
| RoHS | 2011/65/EU | Richtlinie zur Beschränkung der Verwendung bestimmter gefährlicher Stoffe in Elektro- und Elektronikgeräten |

Personenschutz und Geräteschutz

| | | |
|-------------------------------------|---|---------------------------------------|
| Schutzart | - | IP20 |
| Potenzialtrennung | | |
| Zum Feldbus | - | Galvanisch entkoppelt |
| Zur Prozessebene | - | Galvanisch entkoppelt |
| Isolationsfestigkeit | | - |
| Isolationsspannung gegen Bezugserde | | |
| Eingänge / Ausgänge | - | AC / DC 50V, bei Prüfspannung AC 500V |
| Schutzmaßnahmen | - | gegen Kurzschluss |

Umgebungsbedingungen gemäß EN 61131-2

| | | |
|-----------------------------|---------------|--|
| Klimatisch | | |
| Lagerung /Transport | EN 60068-2-14 | -25...+70°C |
| Betrieb | | |
| Horizontaler Einbau hängend | EN 61131-2 | 0...+60°C |
| Horizontaler Einbau liegend | EN 61131-2 | 0...+55°C |
| Vertikaler Einbau | EN 61131-2 | 0...+50°C |
| Luftfeuchtigkeit | EN 60068-2-30 | RH1 (ohne Betauung, relative Feuchte 10 ... 95%) |
| Verschmutzung | EN 61131-2 | Verschmutzungsgrad 2 |
| Aufstellhöhe max. | - | 2000m |
| Mechanisch | | |
| Schwingung | EN 60068-2-6 | 1g, 9Hz ... 150Hz |
| Schock | EN 60068-2-27 | 15g, 11ms |

Allgemeine Daten > Einsatz unter erschwerten Betriebsbedingungen

| Montagebedingungen | | |
|--------------------|---|-------------------------|
| Einbauort | - | Im Schaltschrank |
| Einbaulage | - | Horizontal und vertikal |

| EMV | Norm | Bemerkungen | |
|--------------------------|--------------|----------------------------|--|
| Störaussendung | EN 61000-6-4 | Class A (Industriebereich) | |
| Störfestigkeit Zone B | EN 61000-6-2 | Industriebereich | |
| | | EN 61000-4-2 | ESD 8kV bei Luftentladung (Schärfegrad 3), 4kV bei Kontaktentladung (Schärfegrad 2) |
| | | EN 61000-4-3 | HF-Einstrahlung (Gehäuse) 80MHz ... 1000MHz, 10V/m, 80% AM (1kHz) 1,4GHz ... 2,0GHz, 3V/m, 80% AM (1kHz) 2GHz ... 2,7GHz, 1V/m, 80% AM (1kHz) |
| | | EN 61000-4-6 | HF-Leitungsgeführt 150kHz ... 80MHz, 10V, 80% AM (1kHz) |
| | | EN 61000-4-4 | Burst, Schärfegrad 3 |
| | | EN 61000-4-5 | Surge, Schärfegrad 3 * |

*) Aufgrund der energiereichen Einzelimpulse ist bei Surge eine angemessene externe Beschaltung mit Blitzschutzelementen wie z.B. Blitzstromableitern und Überspannungsableitern erforderlich.

2.3.1 Einsatz unter erschwerten Betriebsbedingungen



Ohne zusätzlich schützende Maßnahmen dürfen die Produkte nicht an Orten mit erschwerten Betriebsbedingungen; z.B. durch:

- Staubentwicklung
- chemisch aktive Substanzen (ätzende Dämpfe oder Gase)
- starke elektrische oder magnetische Felder

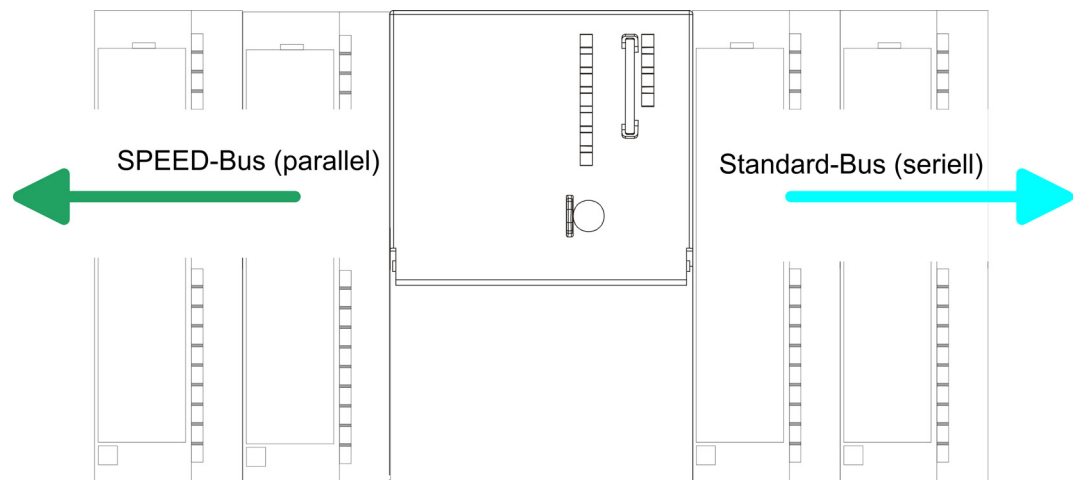
eingesetzt werden!

3 Montage und Aufbaurichtlinien

3.1 Übersicht

SPEED-Bus

- Der SPEED-Bus ist ein von Yaskawa entwickelter 32Bit Parallel-Bus.
- Über SPEED-Bus haben Sie die Möglichkeit bis zu 10 SPEED-Bus-Module an Ihre CPU zu koppeln.
- Im Gegensatz zum "Standard"-Rückwandbus, bei dem die Module rechts von der CPU über Einzel-Busverbinder gesteckt werden, erfolgt beim SPEED-Bus die Ankopplung über eine spezielle SPEED-Bus-Schiene links von der CPU.
- Von Yaskawa erhalten Sie Profilschienen mit integriertem SPEED-Bus für 2, 6 oder 10 SPEED-Bus-Peripherie-Module in unterschiedlichen Längen.
- Jede SPEED-Bus-Schiene besitzt eine Steckmöglichkeit für eine externe Spannungsversorgung. Hiermit können Sie den maximalen Strom am Rückwandbus erhöhen. Nur auf "SLOT1 DCDC" können Sie entweder ein SPEED-Bus-Modul oder eine Zusatzspannungsversorgung (307-1FB70) stecken.



SPEED-Bus-Peripherie-Module

Die SPEED-Bus-Peripherie-Module können ausschließlich auf den hierfür vorgesehenen SPEED-Bus-Steckplätzen links von der CPU eingesetzt werden. Für den SPEED-Bus sind folgende Module verfügbar:

- Schnelle Feldbus-Module, wie PROFIBUS DP-, Interbus-, CANopen-Master und CANopen-Slave
- Schneller CP 343 (CP 343 Kommunikationsprozessor für Ethernet)
- Schneller CP 341 mit 2-facher RS 422/485-Schnittstelle
- Schnelle digitale Ein-/Ausgabe-Module (Fast Digital IN/OUT)

Serieller Standard-Bus

Die einzelnen Module werden direkt auf eine Profilschiene montiert und über den Rückwandbus-Verbinder verbunden. Vor der Montage ist der Rückwandbus-Verbinder von hinten an das Modul zu stecken. Die Rückwandbusverbinder sind im Lieferumfang der Peripherie-Module enthalten.

Paralleler SPEED-Bus

Bei SPEED-Bus erfolgt die Busanbindung über eine in die Profilschiene integrierte SPEED-Bus-Steckleiste links von der CPU. Aufgrund des parallelen SPEED-Bus müssen nicht alle Steckplätze hintereinander belegt sein.

SLOT 1 für Zusatzspannungsversorgung

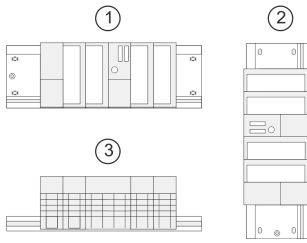
Auf Steckplatz 1 (SLOT 1 DCDC) können Sie entweder ein SPEED-Bus-Modul oder eine Zusatz-Spannungsversorgung stecken.

Einbaumaße

Montagemöglichkeiten

Sie haben die Möglichkeit das System 300 waagrecht, senkrecht oder liegend aufzubauen. Beachten Sie bitte die hierbei zulässigen Umgebungstemperaturen:

- 1 waagrechter Aufbau: von 0 bis 60°C
- 2 senkrechter Aufbau: von 0 bis 50°C
- 3 liegender Aufbau: von 0 bis 55°C

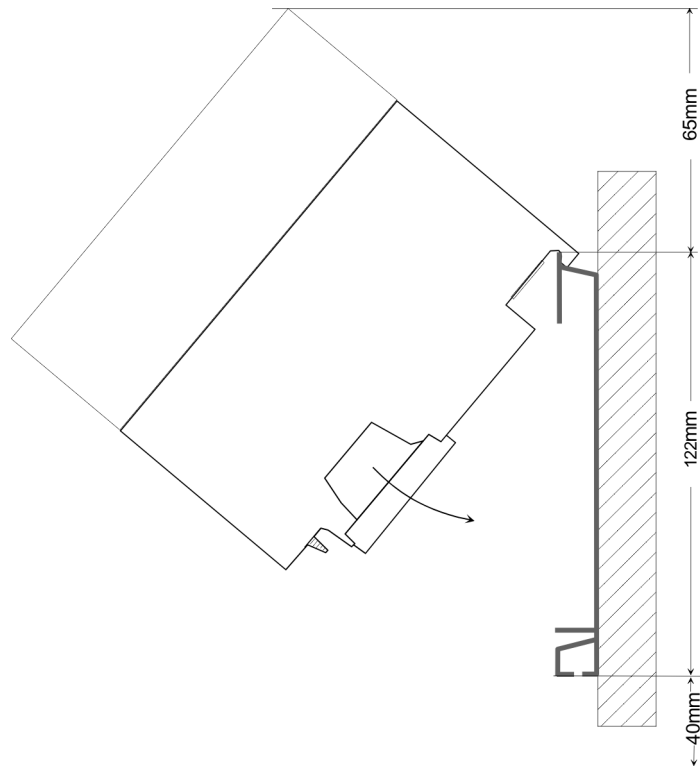


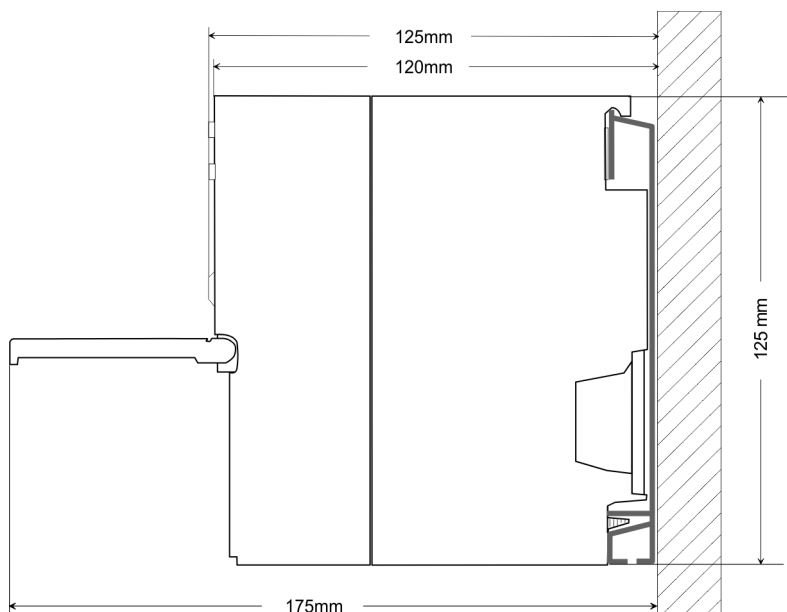
3.2 Einbaumaße

Maße Grundgehäuse

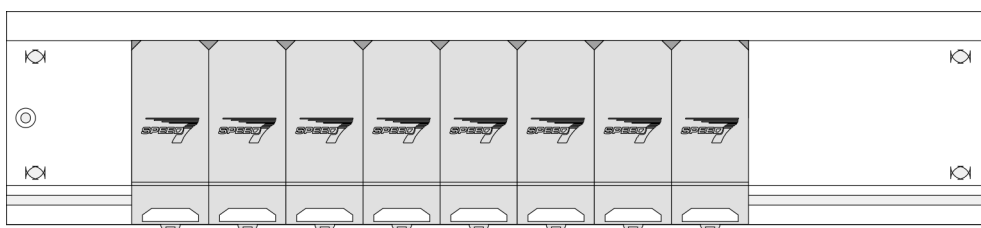
1fach breit (BxHxT) in mm: 40 x 125 x 120

Montagemaße



Maße montiert**3.3 Montage SPEED-Bus****Vorkonfektionierte
SPEED-Bus-Profil-Schiene**

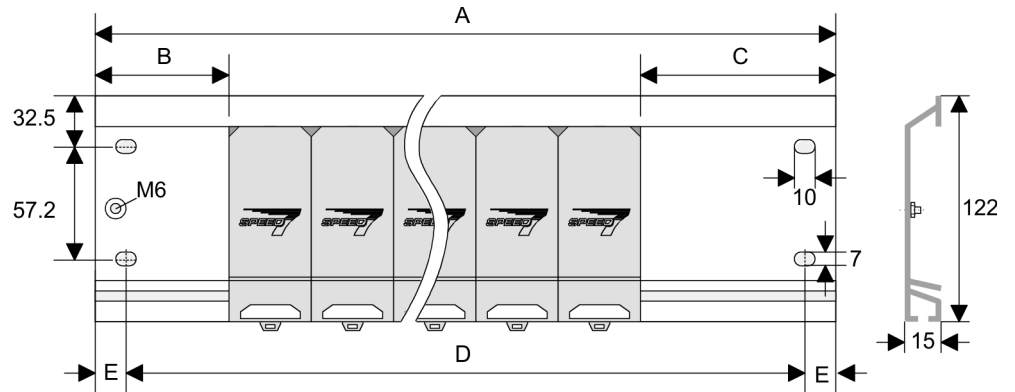
Für den Einsatz von SPEED-Bus-Modulen ist eine vorkonfektionierte SPEED-Bus-Steckleiste erforderlich. Diese erhalten Sie schon montiert auf einer Profilschiene mit 2, 6 oder 10 Steckplätzen.

**Maße**

| Bestellnummer | Anzahl Module SPEED-Bus/ Standard-Bus | A | B | C | D | E |
|---------------|--|-----|-----|-----|-----|----|
| 391-1AF10 | 2/6 | 530 | 100 | 268 | 510 | 10 |
| 391-1AF30 | 6/2 | 530 | 100 | 105 | 510 | 10 |
| 391-1AF50 | 10/0 | 530 | 20 | 20 | 510 | 10 |
| 391-1AJ10 | 2/15 | 830 | 22 | 645 | 800 | 15 |
| 391-1AJ30 | 6/11 | 830 | 22 | 480 | 800 | 15 |
| 391-1AJ50 | 10/7 | 830 | 22 | 320 | 800 | 15 |

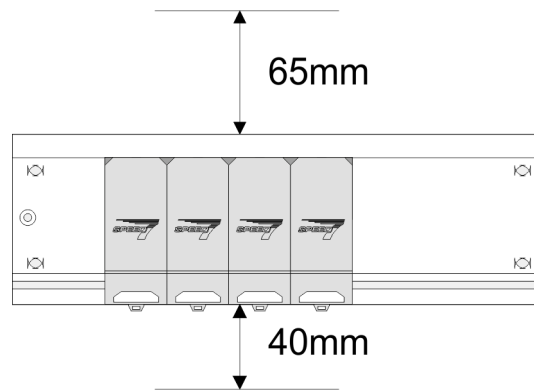
Maße in mm

Montage SPEED-Bus

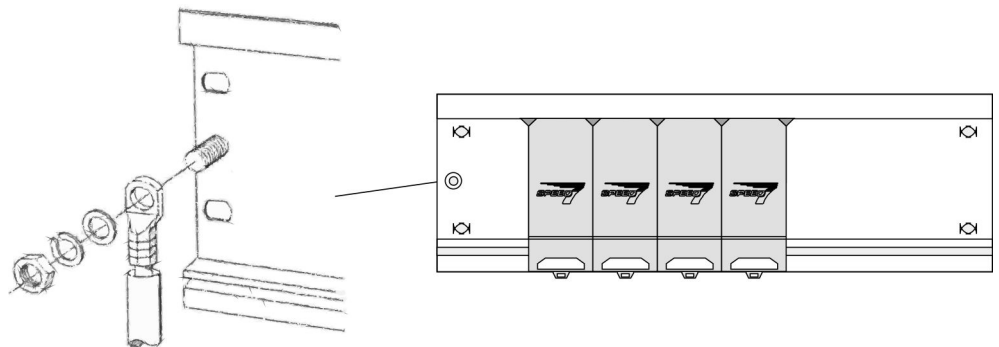


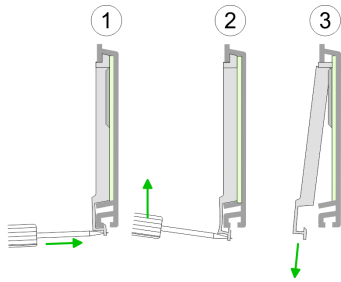
Montage der Profilschiene

1. Verschrauben Sie die Profilschiene mit dem Untergrund (Schraubengröße: M6) so, dass mindestens 65mm Raum oberhalb und 40mm unterhalb der Profilschiene bleibt. Achten Sie immer auf eine niederohmige Verbindung zwischen Profilschiene und Untergrund.



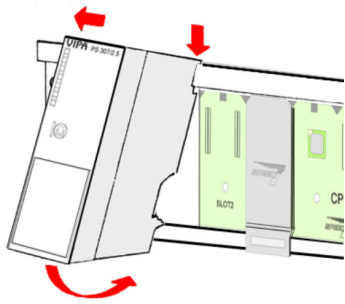
2. Verbinden Sie die Profilschiene über den Stehbolzen mit Ihrem Schutzleiter. Der Mindestquerschnitt der Leitung zum Schutzleiter beträgt hierbei 10mm².



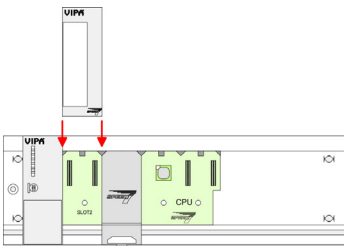
Montage SPEED-Bus-Module

1. ➔ Entfernen Sie mit einem geeigneten Schraubendreher die entsprechenden Schutzabdeckungen über den SPEED-Bus-Steckplätzen, indem Sie diese entriegeln und nach unten abziehen.

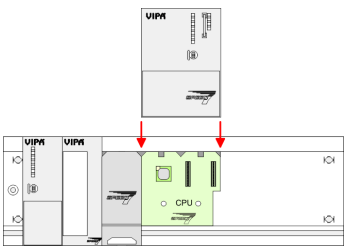
Da es sich bei SPEED-Bus um einen parallelen Bus handelt, müssen nicht alle SPEED-Bus-Steckplätze hintereinander belegt sein. Lassen Sie bei einem nicht benutzten SPEED-Bus-Steckplatz die Abdeckung gesteckt.



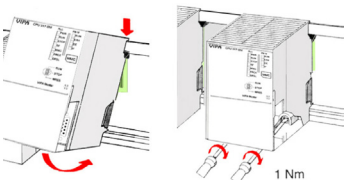
2. ➔ Bei Einsatz einer DC 24V-Spannungsversorgung hängen Sie diese an der gezeigten Position links vom SPEED-Bus auf der Profilschiene ein und schieben Sie diese nach links bis ca. 5mm vor den Erdungsbolzen der Profilschiene.
3. ➔ Schrauben Sie die Spannungsversorgung fest.



4. ➔ Zur Montage von SPEED-Bus-Modulen setzen Sie diese zwischen den dreieckigen Positionierhilfen an einem mit "SLOT ..." bezeichneten Steckplatz an und klappen sie diese nach unten.
5. ➔ Nur auf "SLOT1 DCDC" können Sie entweder ein SPEED-Bus-Modul oder eine Zusatzspannungsversorgung stecken.
6. ➔ Schrauben Sie die CPU fest.

Montage CPU ohne Standard-Bus-Module

1. ➔ Soll die SPEED7-CPU ausschließlich am SPEED-Bus betrieben werden, setzen Sie diese wie gezeigt zwischen den beiden Positionierhilfen an dem mit "CPU SPEED7" bezeichneten Steckplatz an und klappen sie diese nach unten.



2. ➔ Schrauben Sie die CPU fest.

3.4 Aufbaurichtlinien

Allgemeines

Die Aufbaurichtlinien enthalten Informationen über den störsicheren Aufbau eines SPS-Systems. Es werden die Wege beschrieben, wie Störungen in Ihre Steuerung gelangen können, wie die elektromagnetische Verträglichkeit (EMV) sicher gestellt werden kann und wie bei der Schirmung vorzugehen ist.

Was bedeutet EMV?

Unter Elektromagnetischer Verträglichkeit (EMV) versteht man die Fähigkeit eines elektrischen Gerätes, in einer vorgegebenen elektromagnetischen Umgebung fehlerfrei zu funktionieren, ohne vom Umfeld beeinflusst zu werden bzw. das Umfeld in unzulässiger Weise zu beeinflussen.

Die Komponenten von Yaskawa sind für den Einsatz in Industrieumgebungen entwickelt und erfüllen hohe Anforderungen an die EMV. Trotzdem sollten Sie vor der Installation der Komponenten eine EMV-Planung durchführen und mögliche Störquellen in die Betrachtung einbeziehen.

Mögliche Störeinträge

Elektromagnetische Störungen können sich auf unterschiedlichen Pfaden in Ihre Steuerung einkoppeln:

- Elektromagnetische Felder (HF-Einkopplung)
- Magnetische Felder mit energietechnischer Frequenz
- Bus-System
- Stromversorgung
- Schutzleiter

Je nach Ausbreitungsmedium (leitungsgebunden oder -ungebunden) und Entfernung zur Störquelle gelangen Störungen über unterschiedliche Kopplungsmechanismen in Ihre Steuerung.

Man unterscheidet:

- galvanische Kopplung
- kapazitive Kopplung
- induktive Kopplung
- Strahlungskopplung

Grundregeln zur Sicherstellung der EMV

Häufig genügt zur Sicherstellung der EMV das Einhalten einiger elementarer Regeln. Beachten Sie beim Aufbau der Steuerung deshalb die folgenden Grundregeln.

- Achten Sie bei der Montage Ihrer Komponenten auf eine gut ausgeführte flächenhafte Massung der inaktiven Metallteile.
 - Stellen Sie eine zentrale Verbindung zwischen der Masse und dem Erde/Schutzleitersystem her.
 - Verbinden Sie alle inaktiven Metallteile großflächig und impedanzarm.
 - Verwenden Sie nach Möglichkeit keine Aluminiumteile. Aluminium oxidiert leicht und ist für die Massung deshalb weniger gut geeignet.
- Achten Sie bei der Verdrahtung auf eine ordnungsgemäße Leitungsführung.
 - Teilen Sie die Verkabelung in Leitungsgruppen ein. (Starkstrom, Stromversorgungs-, Signal- und Datenleitungen).
 - Verlegen Sie Starkstromleitungen und Signal- bzw. Datenleitungen immer in getrennten Kanälen oder Bündeln.
 - Führen Sie Signal- und Datenleitungen möglichst eng an Masseflächen (z.B. Tragholme, Metallschienen, Schrankbleche).

- Achten Sie auf die einwandfreie Befestigung der Leitungsschirme.
 - Datenleitungen sind geschirmt zu verlegen.
 - Analogleitungen sind geschirmt zu verlegen. Bei der Übertragung von Signalen mit kleinen Amplituden kann das einseitige Auflegen des Schirms vorteilhaft sein.
 - Leitungen für Frequenzumrichter, Servo- und Schrittmotore sind geschirmt zu verlegen.
 - Legen Sie die Leitungsschirme direkt nach dem Schrankeintritt großflächig auf eine Schirm-/Schutzleiterschiene auf, und befestigen Sie die Schirme mit Kabelschellen.
 - Achten Sie darauf, dass die Schirm-/Schutzleiterschiene impedanzarm mit dem Schrank verbunden ist.
 - Verwenden Sie für geschirmte Datenleitungen metallische oder metallisierte Steckergehäuse.
- Setzen Sie in besonderen Anwendungsfällen spezielle EMV-Maßnahmen ein.
 - Erwägen Sie bei Induktivitäten den Einsatz von Löschgliedern.
 - Beachten Sie, dass bei Einsatz von Leuchtstofflampen sich diese negativ auf Signalleitungen auswirken können.
- Schaffen Sie ein einheitliches Bezugspotenzial und erden Sie nach Möglichkeit alle elektrischen Betriebsmittel.
 - Achten Sie auf den gezielten Einsatz der Erdungsmaßnahmen. Das Erden der Steuerung dient als Schutz- und Funktionsmaßnahme.
 - Verbinden Sie Anlagenteile und Schränke mit Ihrer SPS sternförmig mit dem Erde/Schutzleitersystem. Sie vermeiden so die Bildung von Erdschleifen.
 - Verlegen Sie bei Potenzialdifferenzen zwischen Anlagenteilen und Schränken ausreichend dimensionierte Potenzialausgleichsleitungen.

Schirmung von Leitungen

Elektrische, magnetische oder elektromagnetische Störfelder werden durch eine Schirmung geschwächt; man spricht hier von einer Dämpfung. Über die mit dem Gehäuse leitend verbundene Schirmschiene werden Störströme auf Kabelschirme zur Erde hin abgeleitet. Hierbei ist darauf zu achten, dass die Verbindung zum Schutzleiter impedanzarm ist, da sonst die Störströme selbst zur Störquelle werden.

Bei der Schirmung von Leitungen ist folgendes zu beachten:

- Verwenden Sie möglichst nur Leitungen mit Schirmgeflecht.
- Die Deckungsdichte des Schirmes sollte mehr als 80% betragen.
- In der Regel sollten Sie die Schirme von Leitungen immer beidseitig auflegen. Nur durch den beidseitigen Anschluss der Schirme erreichen Sie eine gute Störunterdrückung im höheren Frequenzbereich. Nur im Ausnahmefall kann der Schirm auch einseitig aufgelegt werden. Dann erreichen Sie jedoch nur eine Dämpfung der niedrigen Frequenzen. Eine einseitige Schirmanbindung kann günstiger sein, wenn:
 - die Verlegung einer Potenzialausgleichsleitung nicht durchgeführt werden kann.
 - Analogsignale (einige mV bzw. μA) übertragen werden.
 - Folienschirme (statische Schirme) verwendet werden.
- Benutzen Sie bei Datenleitungen für serielle Kopplungen immer metallische oder metallisierte Stecker. Befestigen Sie den Schirm der Datenleitung am Steckergehäuse. Schirm nicht auf den PIN 1 der Steckerleiste auflegen!
- Bei stationärem Betrieb ist es empfehlenswert, das geschirmte Kabel unterbrechungsfrei abzuisolieren und auf die Schirm-/Schutzleiterschiene aufzulegen.
- Benutzen Sie zur Befestigung der Schirmgeflechte Kabelschellen aus Metall. Die Schellen müssen den Schirm großflächig umschließen und guten Kontakt ausüben.
- Legen Sie den Schirm direkt nach Eintritt der Leitung in den Schrank auf eine Schirmschiene auf. Führen Sie den Schirm bis zu Ihrer SPS weiter, legen Sie ihn dort jedoch nicht erneut auf!

**VORSICHT!****Bitte bei der Montage beachten!**

Bei Potenzialdifferenzen zwischen den Erdungspunkten kann über den beidseitig angeschlossenen Schirm ein Ausgleichsstrom fließen.

Abhilfe: Potenzialausgleichsleitung.

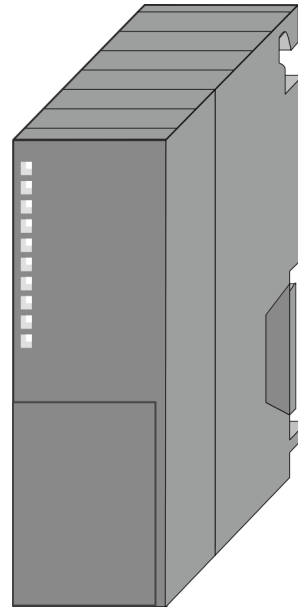
4 Hardwarebeschreibung

4.1 Leistungsmerkmale

CP 342-1CA70

Der CP darf ausschließlich auf dem SPEED-Bus eingesetzt werden.

- CANopen-Master für SPEED-Bus
- 125 CAN-Slaves an einen CANopen-Master ankoppelbar
- Projektierung unter WinCoCT von VIPA
- Diagnosefähig
- 40 Transmit PDOs
- 40 Receive PDOs
- PDO-Linking
- PDO-Mapping
- 1 SDO als Server, 127 SDO als Client
- Emergency Object
- NMT Object
- Node Guarding, Heartbeat
- Ein-/Ausgabe-Bereich 0x6xxx je maximal 320Bytes
- Ein-/Ausgabe-Bereich 0xAxxx je maximal 320Bytes

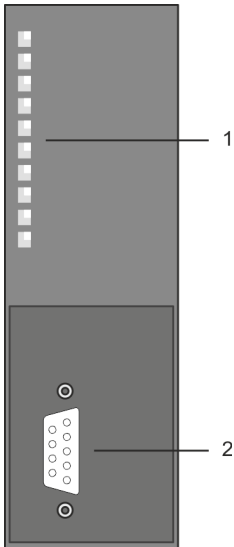


Bestelldaten

| Typ | Bestellnummer | Beschreibung |
|-------------|---------------|------------------------------|
| CP 342S-CAN | 342-1CA70 | CANopen-Master für SPEED-Bus |

4.2 Aufbau

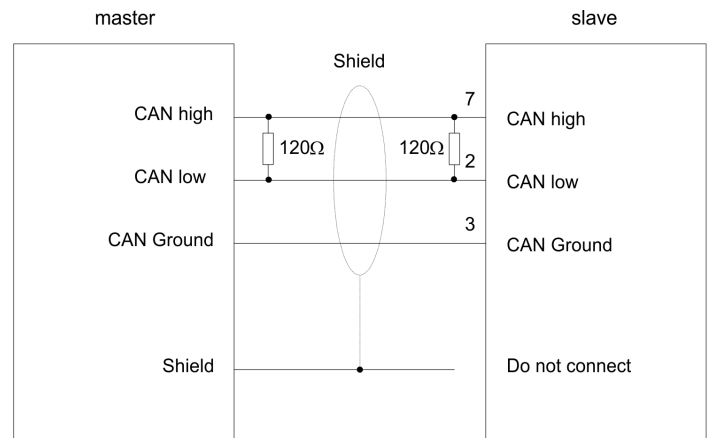
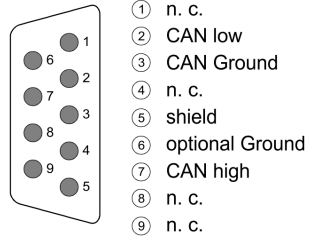
CP 342-1CA70



- 1 LED Statusanzeigen
Folgende Komponente befindet sich unter der Frontklappe:
- 2 CAN-Schnittstelle

CAN-Schnittstelle

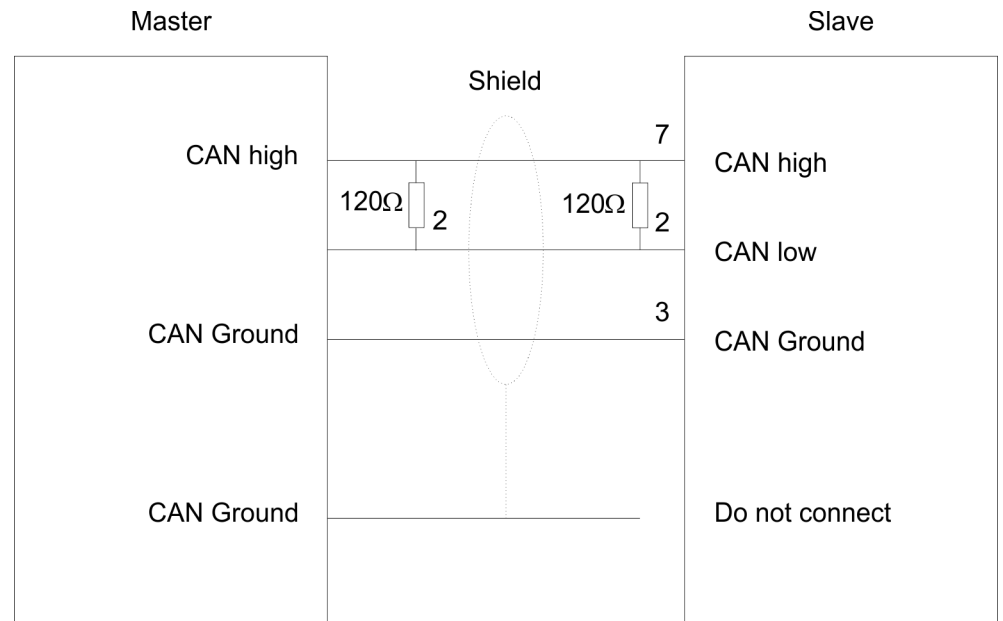
CANopen master
X2



Bus-Anbindung

9-poliger CAN Stecker:

CAN-Bus verwendet als Übertragungsmedium eine abgeschirmte Dreidrahtleitung. In Systemen mit mehr als zwei Stationen werden alle Teilnehmer parallel verdrahtet. Hierzu ist das Bus-Kabel unterbrechungsfrei durchzuschleifen.



An den Leitungsenden muss das Bus-Kabel immer mit einem Abschlusswiderstand von 120Ω abgeschlossen werden, um Reflexionen und damit Übertragungsprobleme zu vermeiden!

LEDs Der CP 342-1CA70 besitzt verschiedene LEDs, die der Busdiagnose dienen und den eigenen Betriebszustand anzeigen. Abhängig von der Betriebsart geben diese nach folgendem Schema Auskunft über den Betriebszustand des CP:








Master-Betrieb

| RUN ■ grün | ERR ■ rot | BA ■ gelb | IF ■ rot | Bedeutung |
|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|---|
| <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | Master hat keine Projektierung, d.h. die Schnittstelle ist deaktiviert. |
| <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | Blinkend 1Hz: Master wartet auf gültige Parameter von der CPU. |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | CPU befindet sich im RUN. Master befindet sich im Zustand "operational", d.h. er tauscht Daten mit den Slaves aus. Eingänge können gelesen und Ausgänge angesprochen werden. |
| <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | CPU befindet sich im RUN. Master befindet sich im Zustand "operational", es fehlt mindestens 1 Slave. |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | CPU befindet sich im RUN. Blinkend 1Hz: Master befindet sich im Zustand "pre-operational". Die Eingänge sind undefiniert und die Ausgänge sind gesperrt. |
| <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | CPU befindet sich im RUN. Blinkend 1Hz: Master befindet sich im Zustand "pre-operational", es fehlt mindestens 1 Slave. |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | CPU befindet sich im RUN. Blinkend 10Hz: Master befindet sich im Zustand "prepared". |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | CPU befindet sich im STOP. Es fehlt mindestens 1 Slave. |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | CPU befindet sich im STOP. Master zeigt Initialisierungsfehler bei fehlerhafter Parametrierung. |

Slave-Betrieb

| RUN ■ grün | ERR ■ rot | BA ■ gelb | IF ■ rot | Bedeutung |
|-------------------------------------|-------------------------------------|-------------------------------------|--------------------------|--|
| <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | Slave hat keine Projektierung, d.h. die Schnittstelle ist deaktiviert. |
| <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | Blinkend 1Hz: Slave wartet auf gültige Parameter von der CPU. |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | CPU befindet sich im RUN. Slave befindet sich im Zustand "operational". Er tauscht Daten mit dem Master aus. Eingänge können gelesen und Ausgänge angesprochen werden. |

Aufbau

| RUN  grün | ERR  rot | BA  gelb | IF  rot | Bedeutung |
|---|--|--|---|--|
|  | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | CPU befindet sich im RUN. Blinkend 1Hz: Slave befindet sich im Zustand "pre-operational". Die Eingänge sind undefiniert und die Ausgänge sind gesperrt. Falls parametrierung, Anzeige dass der Master ausgefallen ist. |
| <input type="checkbox"/> |  | <input type="checkbox"/> |  | CPU befindet sich im STOP. Slave zeigt Initialisierungsfehler bei fehlerhafter Parametrierung. |

Spannungsversorgung

Der CP 342-1CA70 bezieht seine Spannungsversorgung über den SPEED-Bus. ↪ *Kap. 4.3 "Technische Daten" Seite 27*

Firmwareupdate

Sie haben die Möglichkeit mittels einer Speicherkarte über die SPEED7-CPU ein Firmwareupdate unter anderem auch für den CP 342-1CA70 durchzuführen. Damit eine Firmwaredatei beim Hochlauf erkannt und zugeordnet werden kann, ist für jede updatefähige Komponente und jeden Hardware-Ausgabestand ein pkg-Dateiname reserviert, der mit "px" beginnt und sich in einer 6-stelligen Ziffer unterscheidet. Den pkg-Dateinamen finden Sie unter der Frontklappe auf einem Aufkleber auf der rechten Seite des Moduls.

Node-ID über Yaskawa WinCoCT einstellen

Die Zuteilung der Node-ID (Knotenadresse) erfolgt unter der Konfiguration mit WinCoCT. Hierbei kann die Adresse zwischen 1 ... 126 liegen, wobei jede Adresse nur einmal in diesem Bus-System vergeben sein darf. Bitte beachten Sie, dass Sie unter Einsatz von WinCoCT die Node-ID nachträglich nicht mehr ändern können.

Ein-/Ausgangs-Daten

Der CP kann max. 320Byte Eingangs- und 320Byte Ausgangsdaten, d.h. max. 40 PDOs verarbeiten.

Einsatz

Über einen CANopen-Master können bis zu 126 CANopen-Slaves an die CPU angekoppelt werden. Der CANopen-Master kommuniziert mit den CANopen-Slaves und blendet die Datenbereiche im Adressbereich der CPU ein. Bei jedem NETZ EIN bzw. nach dem URLÖSCHEN holt sich die CPU vom Master die I/O-Mapping-Daten. Hat der CP keine Parameter, so sind die zugehörigen LEDs aus und die CANopen-Schnittstelle ist deaktiviert.

4.3 Technische Daten

| Artikelnr. | 342-1CA70 |
|--|--|
| Bezeichnung | CP 342S CAN - CANopen-Master - SPEED-Bus |
| SPEED-Bus | ✓ |
| Stromaufnahme/Verlustleistung | |
| Stromaufnahme aus Rückwandbus | 550 mA |
| Verlustleistung | 2,75 W |
| Status, Alarm, Diagnosen | |
| Statusanzeige | ja |
| Alarmer | nein |
| Prozessalarm | nein |
| Diagnosealarm | nein |
| Diagnosefunktion | nein |
| Diagnoseinformation auslesbar | möglich |
| Versorgungsspannungsanzeige | keine |
| Sammelfehleranzeige | ja |
| Kanalfehleranzeige | keine |
| Funktionalität Sub-D Schnittstellen | |
| Bezeichnung | CAN |
| Physik | CAN |
| Anschluss | 9poliger SubD Stecker |
| Potenzialgetrennt | ✓ |
| MPI | - |
| MP ² I (MPI/RS232) | - |
| Punkt-zu-Punkt-Kopplung | - |
| 5V DC Spannungsversorgung | - |
| 24V DC Spannungsversorgung | - |
| | |
| Bezeichnung | - |
| Physik | - |
| Anschluss | - |
| Potenzialgetrennt | - |
| MPI | - |
| MP ² I (MPI/RS232) | - |
| Punkt-zu-Punkt-Kopplung | - |
| 5V DC Spannungsversorgung | - |
| 24V DC Spannungsversorgung | - |

Technische Daten

| | |
|---|-------------------------|
| Artikelnr. | 342-1CA70 |
| Funktionalität RJ45 Schnittstellen | |
| Bezeichnung | - |
| Physik | - |
| Anschluss | - |
| Potenzialgetrennt | - |
| PG/OP Kommunikation | - |
| max. Anzahl Verbindungen | - |
| Produktiv Verbindungen | - |
| Feldbus | - |
| | |
| Bezeichnung | - |
| Physik | - |
| Anschluss | - |
| Potenzialgetrennt | - |
| PG/OP Kommunikation | - |
| max. Anzahl Verbindungen | - |
| Produktiv Verbindungen | - |
| Feldbus | - |
| Gehäuse | |
| Material | PPE |
| Befestigung | Profilschiene SPEED-Bus |
| Mechanische Daten | |
| Abmessungen (BxHxT) | 40 mm x 125 mm x 120 mm |
| Gewicht Netto | 210 g |
| Gewicht inklusive Zubehör | - |
| Gewicht Brutto | - |
| Umgebungsbedingungen | |
| Betriebstemperatur | 0 °C bis 60 °C |
| Lagertemperatur | -25 °C bis 70 °C |
| Zertifizierungen | |
| Zertifizierung nach UL | ja |
| Zertifizierung nach KC | - |

5 Einsatz

5.1 Grundlagen CAN

Allgemeines

- Der CAN-Bus (Control Area Network) ist ein international offener Feldbus-Standard für Gebäude-, Fertigungs- und Prozessautomatisierung und wurde ursprünglich für die Automobiltechnik entwickelt.
- Aufgrund der umfassenden Fehlererkennungs-Maßnahmen gilt der CAN-Bus als das sicherste Bus-System mit einer Restfehlerwahrscheinlichkeit von weniger als $4,7 \times 10^{-11}$. Fehlerhafte Meldungen werden signalisiert und automatisch neu übertragen.
- Im Gegensatz zu PROFIBUS und INTERBUS sind beim CAN-Bus auch verschiedene Schicht-7-Anwenderprofile unter dem CAL-Schicht-7-Protokoll definiert (CAL=CAN application layer). Ein solches Anwenderprofil ist CANopen, dessen Standardisierung der CiA (CAN in Automation) e.V. übernimmt.

CANopen

- CANopen ist das Anwenderprofil für den Bereich industrieller Echtzeitsysteme und wird zur Zeit von vielen Herstellern implementiert. CANopen wurde als Profil DS-301 von der CAN-Nutzerorganisation (C.i.A) veröffentlicht. Das Kommunikationsprofil DS-301 dient zur Standardisierung der Geräte. Somit werden die Produkte verschiedener Hersteller austauschbar. Weiter sind zur Gewährleistung der Austauschbarkeit in dem Geräteprofil DS-401 die gerätespezifischen Daten und die Prozessdaten standardisiert. DS-401 standardisiert die digitalen und analogen Ein-/Ausgabe-Module.
- CANopen besteht aus dem Kommunikationsprofil (communication profile), das festlegt, welche Objekte für die Übertragung bestimmter Daten zu verwenden sind, und den Geräteprofilen (device profiles), die die Art der Daten spezifizieren, die mit den Objekten übertragen werden.
- Das CANopen-Kommunikationsprofil basiert auf einem Objektverzeichnis ähnlich dem des PROFIBUS. Im Kommunikationsprofil DS-301 sind zwei Objektarten sowie einige Spezialobjekte definiert:
 - Prozessdatenobjekte (PDO)
 - PDOs dienen der Übertragung von Echtzeitdaten
 - Servicedatenobjekte (SDO)
 - SDOs ermöglichen den lesenden und schreibenden Zugriff auf das Objektverzeichnis

Übertragungsmedium

- CAN basiert auf einer linienförmigen Topologie. Sie haben die Möglichkeit, mittels Routerknoten eine Netzstruktur aufzubauen. Die Anzahl der Teilnehmer pro Netz wird nur durch die Leistungsfähigkeit des eingesetzten Bus-Treiberbausteins begrenzt.
- Die maximale Netzausdehnung ist durch Signallaufzeiten begrenzt. Bei 1Mbit/s ist z.B. eine Netzausdehnung von 40m und bei 80kbit/s von 1000m möglich.
- CAN-Bus verwendet als Übertragungsmedium eine abgeschirmte Dreidrahtleitung (Fünfdraht optional). Der CAN-Bus arbeitet mit Spannungsdifferenzen. Er ist daher unempfindlicher gegenüber Störeinflüssen als eine Spannungs- oder Stromschnittstelle. Das Netz sollte als Linie konfiguriert sein, mit einem 120Ω Abschlusswiderstand am Ende.
- Auf dem CP befindet sich ein 9poliger Stecker. Über diesen Stecker koppeln Sie den CAN-Bus-Koppler als Slave direkt in das CAN-Bus-Netz ein.
- Alle Teilnehmer im Netz kommunizieren mit der gleichen Übertragungsrates. Die Bus-Struktur erlaubt das rückwirkungsfreie Ein- und Auskoppeln von Stationen oder die schrittweise Inbetriebnahme des Systems. Spätere Erweiterungen haben keinen Einfluss auf Stationen, die bereits in Betrieb sind. Es wird automatisch erkannt, ob ein Teilnehmer ausgefallen oder neu am Netz ist.

Bus-Zugriffsverfahren

- Man unterscheidet bei Bus-Zugriffsverfahren generell zwischen kontrolliertem (deterministischem) und unkontrolliertem (zufälligen) Bus-Zugriff.
- CAN arbeitet nach dem Verfahren Carrier-Sense Multiple Access (CSMA), d.h. jeder Teilnehmer ist bezüglich des Bus-Zugriffs gleichberechtigt und kann auf den Bus zugreifen, sobald dieser frei ist (zufälliger Bus-Zugriff).
- Der Nachrichtenaustausch ist nachrichtenbezogen und nicht teilnehmerbezogen. Jede Nachricht ist mit einem priorisierenden Identifier eindeutig gekennzeichnet. Es kann immer nur ein Teilnehmer für seine Nachricht den Bus belegen.
- Die Bus-Zugriffssteuerung bei CAN geschieht mit Hilfe der zerstörungsfreien, bitweisen Arbitrierung. Hierbei bedeutet zerstörungsfrei, dass der Gewinner der Arbitrierung sein Telegramm nicht erneut senden muss. Beim gleichzeitigen Mehrfachzugriff von Teilnehmern auf den Bus wird automatisch der wichtigste Teilnehmer ausgewählt. Erkennt ein sendebereiter Teilnehmer, dass der Bus belegt ist, so wird sein Sendewunsch bis zum Ende der aktuellen Übertragung verzögert.

5.2 Adressierung am SPEED-Bus

Übersicht

Damit die gesteckten Peripheriemodule am SPEED-Bus gezielt angesprochen werden können, müssen ihnen bestimmte Adressen in der CPU zugeordnet werden. Sofern keine Hardware-Konfiguration vorliegt, vergibt die CPU beim Hochlauf steckplatzabhängig automatisch E/A-Peripherieadressen unter anderem auch für gesteckte Module am SPEED-Bus.

Maximale Anzahl steckbarer Module

Im Hardware-Konfigurator von Siemens können Sie maximal 8 Module pro Zeile parametrieren. Bei Einsatz der SPEED7-CPU's können Sie bis zu 32 Module am Standard-Bus und zusätzlich 10 Module am SPEED-Bus ansteuern. Hier gehen CPs und DP-Master, da diese zusätzlich virtuell am Standard-Bus zu projektieren sind, in die Summe von 32 Modulen am Standard-Bus mit ein. Für die Projektierung von Modulen, die über die Anzahl von 8 hinausgehen, können virtuell Zeilenanschlaltungen verwendet werden. Hierbei setzen Sie im Hardware-Konfigurator auf Ihre 1. Profilschiene auf Steckplatz 3 die Anschaltung IM 360 aus dem Hardware-Katalog. Nun können Sie Ihr System um bis zu 3 Profilschienen ergänzen, indem Sie jede auf Steckplatz 3 mit einer IM 361 von Siemens beginnen.

Über Hardware-Konfiguration Adressen definieren

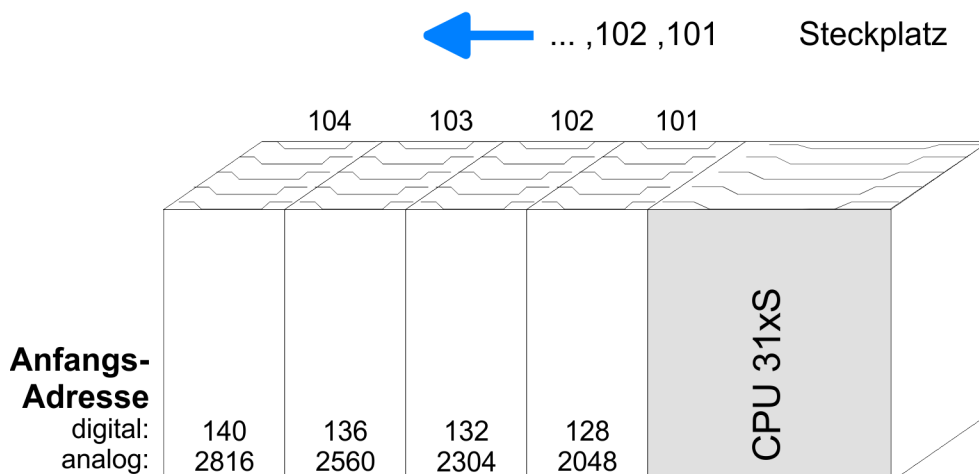
Über Lese- bzw. Schreibzugriffe auf die Peripheriebytes oder auf das Prozessabbild können Sie die Module ansprechen. Mit einer Hardware-Konfiguration können Sie über ein virtuelles PROFIBUS-System durch Einbindung der SPEEDBUS.GSD Adressen definieren. Klicken Sie hierzu auf die Eigenschaften des entsprechenden Moduls und stellen Sie die gewünschte Adresse ein.

Automatische Adressierung

Falls Sie keine Hardware-Konfiguration verwenden möchten, tritt eine automatische Adressierung in Kraft. Bei der automatischen Adressierung werden steckplatzabhängig DI0s in einem Abstand von 4Byte und AIOs, FMs, CPs in einem Abstand von 256Byte abgelegt.

Nach folgenden Formeln wird steckplatzabhängig die Anfangsadresse ermittelt, ab der das entsprechende Modul im Adressbereich abgelegt wird:

- DI0s: Anfangsadresse = $4 \times (\text{Steckplatz} - 101) + 128$
- AIOs, FMs, CPs: Anfangsadresse = $256 \times (\text{Steckplatz} - 101) + 2048$



5.3 Projektierung Schnelleinstieg

Übersicht

Die Projektierung des CANopen-Masters am SPEED-Bus erfolgt unter WinCoCT (Windows CANopen Configuration Tool) von Yaskawa. Aus WinCoCT exportieren Sie Ihr Projekt als wld-Datei. Die wld-Datei können Sie in Ihren Hardware-Konfigurator von Siemens importieren. Zur Einbindung des CAN-Master-Moduls in Ihre SPEED7-CPU ist aus dem SPEED-Bus-Hardwarekatalog das CAN-Master-Modul als VIPA_SPEEDBUS DP-Slave an einem virtuellen DP-Master zu projektieren.

Schnelleinstieg

Für den Einsatz von System 300S Modulen und des CAN-Masters am SPEED-Bus ist die Einbindung der System 300S Module über die GSD-Datei von Yaskawa im Hardwarekatalog erforderlich. Zur Projektierung im Hardware-Konfigurator sind folgende Schritte durchzuführen:

1. ➤ WinCoCT starten und CANopen-Netzwerk projektieren
2. ➤ Hierzu mit  eine "Master"-Gruppe anlegen und mit  einen CANopen-Master für SPEED-Bus einfügen. Bitte beachten Sie, dass Sie nachträglich eine Node-ID nicht mehr ändern können.
3. ➤ Über **Node** > CANopen Manager mit "Device is NMT Master" die Master-Funktion aktivieren und mit [Close] bestätigen.
4. ➤ Mit **Node** > PLC Parameters Parameter vorgeben wie Diagnose-Verhalten und CPU-Adress-Bereiche.
5. ➤ Eine "Slave"-Gruppe mit  anlegen und mit  Ihre CANopen-Slaves hinzufügen.
6. ➤ Den Slaves sofern konfigurierbar über "Module" Module hinzufügen und ggf. parametrieren
7. ➤ Unter "Connections" Prozessdatenverbindungen in der Matrix einstellen. ggf. Eingabe im Prozessabbild des Master überprüfen.
8. ➤ Projekt speichern und mit **File** > Export als wld-Datei exportieren.
9. ➤ In den Siemens SIMATIC Manager wechseln und Datenbaustein von CAN-wld-Datei in Bausteine-Verzeichnis kopieren.
10. ➤ Hardware-Konfigurator von Siemens starten und SPEEDBUS.GSD für SPEED7 von Yaskawa einbinden.
11. ➤ Entsprechende Siemens CPU projektieren.
12. ➤ Beginnend mit Steckplatz 4, die System 300 Module am Standard-Bus in gesteckter Reihenfolge platzieren.
13. ➤ Für den SPEED-Bus immer als letztes Modul den DP-Master CP 342-5 von Siemens (342-5DA02 V5.0) einbinden, vernetzen und in die *Betriebsart* DP-Master parametrieren. An dieses Mastersystem jedes einzelne SPEED-Bus-Modul als VIPA_SPEEDBUS-Slave anbinden. Hierbei entspricht die PROFIBUS-Adresse der Steckplatz-Nr. beginnend mit 100 für die CPU. Auf dem Steckplatz 0 jedes Slaves das ihm zugeordnete Modul platzieren und ggf. Parameter ändern.

5.4 Projektierung

Voraussetzung

Der Hardware-Konfigurator ist Bestandteil des Siemens SIMATIC Managers und er dient der Projektierung. Die Module, die hier projiziert werden können, entnehmen Sie dem Hardware-Katalog. Für den Einsatz der System 300S Module am SPEED-Bus ist die Einbindung der System 300S Module über die GSD-Datei SPEEDBUS.GSD von Yaskawa im Hardwarekatalog erforderlich.



Für die Projektierung werden fundierte Kenntnisse im Umgang mit dem Siemens SIMATIC Manager und dem Hardware-Konfigurator vorausgesetzt!

SPEEDBUS.GSD installieren

Die GSD (Geräte-Stamm-Datei) ist in folgenden Sprachversionen online verfügbar. Weitere Sprachen erhalten Sie auf Anfrage:

| Name | Sprache |
|--------------|-------------------|
| SPEEDBUS.GSD | deutsch (default) |
| SPEEDBUS.GSG | deutsch |
| SPEEDBUS.GSE | englisch |

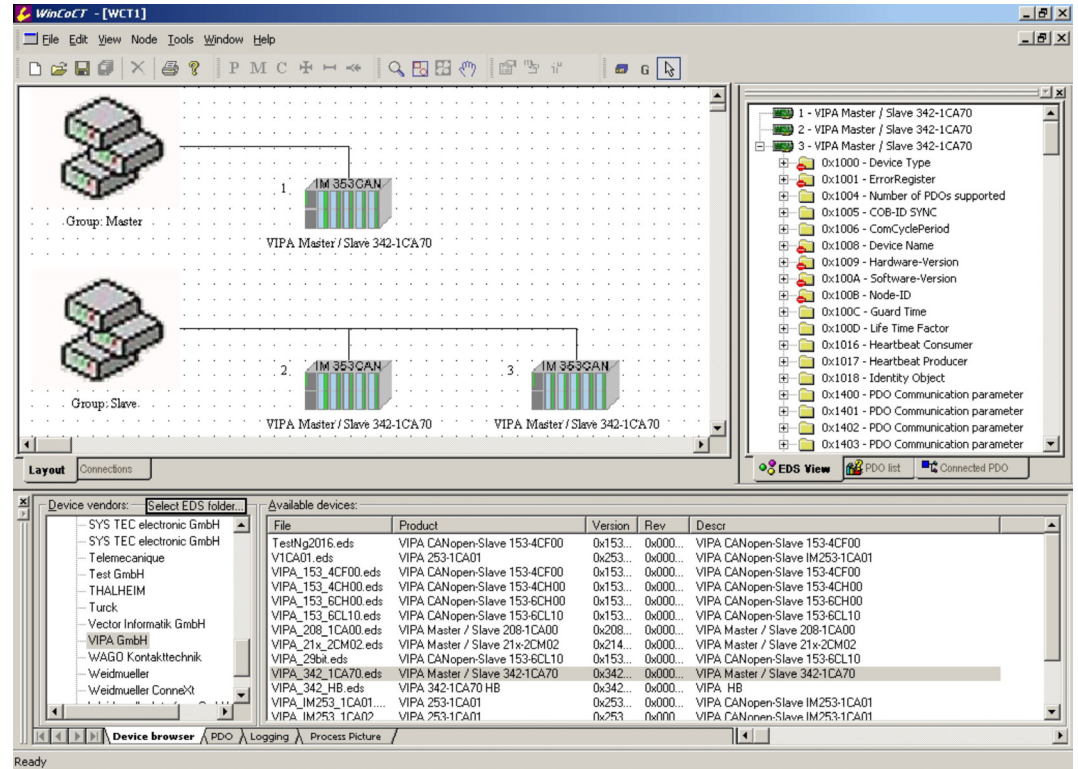
Die GSD-Dateien finden Sie auf www.yaskawa.eu.com im Service-Bereich.

Die Einbindung der SPEEDBUS.GSD erfolgt nach folgender Vorgehensweise:

1. ➤ Gehen Sie in den Service-Bereich von www.yaskawa.eu.com.
2. ➤ Laden Sie aus dem Downloadbereich unter "*Config Dateien* ➔ *PROFIBUS*" die entsprechende Datei für Ihr System 300S.
3. ➤ Extrahieren Sie die Datei in Ihr Arbeitsverzeichnis.
4. ➤ Starten Sie den Hardware-Konfigurator von Siemens.
5. ➤ Schließen Sie alle Projekte.
6. ➤ Gehen Sie auf "*Extras* ➔ *Neue GSD-Datei installieren*".
7. ➤ Navigieren Sie in das Verzeichnis `VIPA_System_300S` und geben Sie **SPEEDBUS.GSD** an.
 - ⇒ Alle SPEED7-CPU's und -Module des System 300S von Yaskawa sind jetzt im Hardwarekatalog unter Profibus-DP / Weitere Feldgeräte / I/O / VIPA_SPEEDBUS enthalten.

WinCoCT

WinCoCT (**Windows CANopen Configuration Tool**) ist ein von VIPA entwickeltes Konfigurations-Tool zur komfortablen Projektierung von CANopen-Netzwerken. WinCoCT stellt auf einer grafischen Benutzeroberfläche die Topologie Ihres CANopen-Netzwerks dar. Hier können Sie Feldgeräte und Steuerungen platzieren, parametrieren, gruppieren und Verbindungen projektieren. Die Auswahl der Geräte erfolgt über eine Liste, die jederzeit über eine EDS-Datei (**E**lectronic **D**ata **S**heet) beliebig erweitert werden kann. Durch Klick mit der rechten Maustaste auf ein Gerät, erscheint für dieses Gerät ein Menü, das zu einem Teil aus statischen und zum anderen Teil aus dynamischen Komponenten besteht. Zur Konfiguration des Prozessdatenaustauschs werden alle PDOs in Form einer Matrix dargestellt, wobei TxPDOs als Zeile und RxPDOs als Spalte ausgegeben werden.

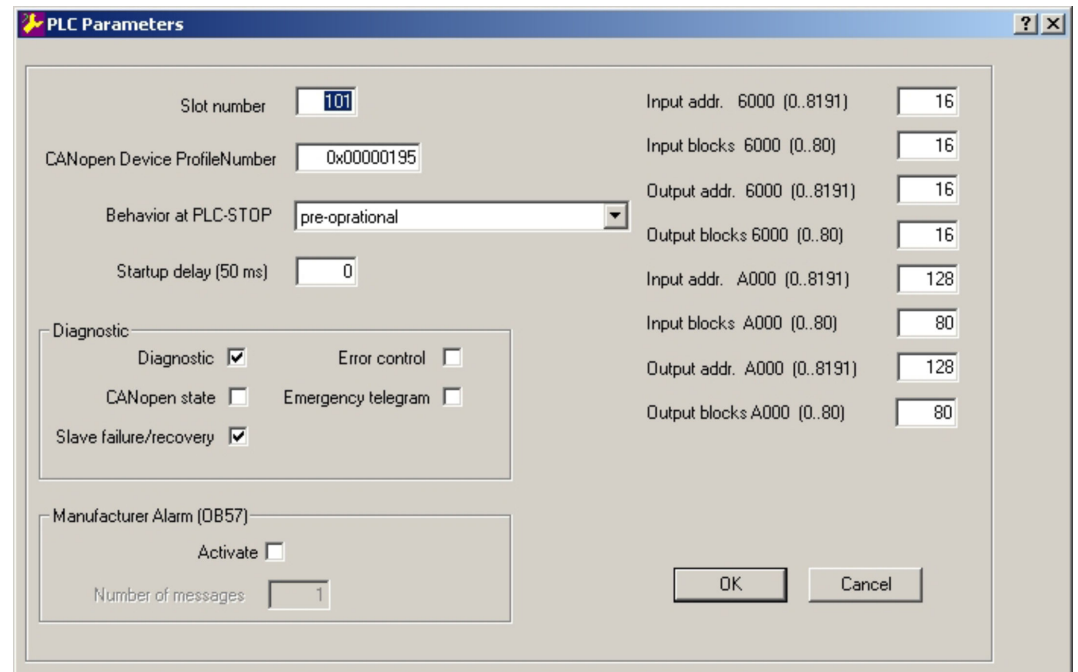


Projektparameter ein- stellen

Über **Tools** > *Project options* können Sie CAN-spezifische Parameter, wie Baud-Rate, Auswahl des Masters usw. vorgeben. Näheres hierzu finden Sie in der Beschreibung von WinCoCT.

Parameter SPEED-Bus CAN-Master

Sie haben die Möglichkeit über WinCoCT VIPA-spezifische Parameter für den CAN-Master vorzugeben, indem Sie mit der rechten Maustaste auf den Master klicken und mit Set PLC-Parameters den nachfolgenden Dialog aufrufen:



| | |
|-------------------------------------|---|
| Slot number | Steckplatz-Nr. auf dem Bus 101 ... 110: Zur Adressierung am SPEED-Bus, wobei Slot number 101 dem SLOT 1 am SPEED-Bus entspricht |
| CANopen DeviceProfile-Number | Fest eingestellt auf 0x195 |
| Behavior at PLC-STOP | Hier können Sie das Verhalten der Ausgabe-Kanäle einstellen, sobald die CPU in STOP geht. Folgende Werte stehen zur Auswahl: <ul style="list-style-type: none"> ■ <i>Switch substitute value 0:</i> Schaltet alle Ausgänge auf 0. Der Slave bleibt im Zustand operational. ■ <i>Keep last value:</i> Friert den aktuellen Zustand der Ausgänge ein. Der Slave bleibt im Zustand operational. ■ <i>Pre-operational:</i> Alle projektierten Slaves werden in den Zustand pre-operational gesetzt. Beim Übergang von CPU-STOP nach CPU-RUN werden alle Slaves in den Zustand operational gesetzt. ■ <i>Pre-operational + switch substitute value:</i> Schaltet alle Ausgänge auf 0. Danach werden alle projektierten Slaves in den Zustand pre-operational gesetzt. Beim Übergang von CPU-STOP nach CPU-RUN werden alle Slaves in den Zustand operational gesetzt. |
| Diagnostic | In diesem Bereich können Sie das Diagnose-Verhalten des CAN-Masters einstellen. Mit dem SFC 13 können Sie die Diagnose auslesen. <ul style="list-style-type: none"> ■ <i>Diagnostic:</i> Aktiviert die Diagnosefunktion |
| NMT-Slave | <ul style="list-style-type: none"> ■ <i>CANopen state:</i> Im aktivierten Zustand sendet der CAN-Master seinen Status "pre-operational" oder "operational" an die CPU. Den Status können Sie über SFC 13 abrufen. |
| NMT-Master | <ul style="list-style-type: none"> ■ <i>Slave failure/recovery:</i> Wenn Sie diese Option aktiviert haben wird bei Slave-Ausfall und -Wiederkehr der OB 86 in der CPU aufgerufen. ■ <i>Error control:</i> Ist diese Option angewählt, so sendet der NMT-Master alle Guarding-Fehler als Diagnose an die CPU, die den OB 82 aufruft. ■ <i>Emergency Telegram:</i> Bei aktivierter Option sendet der NMT-Master alle Emergency-Telegramme als Diagnose an die CPU, die den OB 82 aufruft. |
| Adressbereich in der CPU | Hier können Sie die Adressbereiche vorgeben, die der CAN-Master für seine E/A-Bereiche in der CPU belegt. Jeder Block besteht aus 4Byte. <ul style="list-style-type: none"> ■ <i>Input addr. 6000, Input blocks</i> PAE-Basis-Adresse in der CPU, die von 0x6000-CAN-Eingangsdaten belegt werden. Für Input blocks können max. 80 (320Byte) eingetragen werden. ■ <i>Output addr. 6000, Output blocks</i> PAA-Basis-Adresse in der CPU, die von 0x6000-CAN-Ausgangsdaten belegt werden. Für Output blocks können max. 80 (320Byte) eingetragen werden. |

- *Input addr. A000, Input blocks*
PAE-Basis-Adresse in der CPU, die von 0xA000-CAN-Eingangs-Netzwerk-Variablen belegt werden. Für Input blocks können max. 80 (320Byte) eingetragen werden.
- *Output addr. A000, Output blocks*
PAA-Basis-Adresse in der CPU, die von 0xA000-CAN-Ausgangs-Netzwerk-Variablen belegt werden. Für Output blocks können max. 80 (320Byte) eingetragen werden.

Manufacturer Alarm (OB 57)

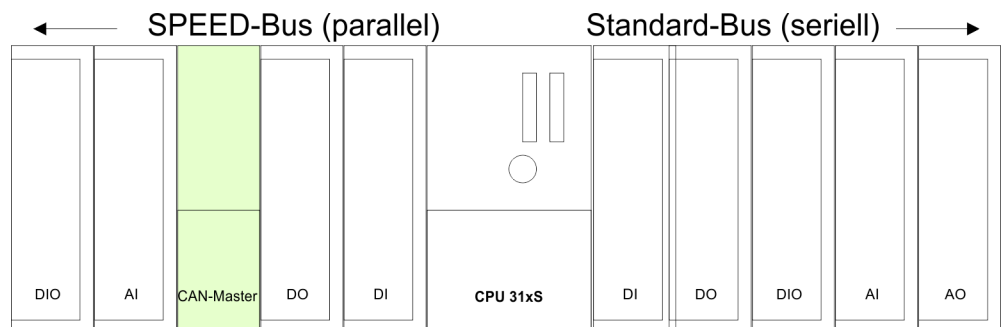
- *Activate*: Aktiviert den hersteller-spezifischen Alarm OB 57.
- *Number of Messages*: Gibt die Anzahl der Telegramme an, die empfangen werden müssen, damit der OB 57 ausgelöst wird. Zusätzlich muss der Index 2000h im CANopen-Objekt-Verzeichnis initialisiert werden.

Schritte der Projektierung

Nachfolgend wird die Vorgehensweise der Projektierung an einem abstrakten Beispiel gezeigt: Die Projektierung gliedert sich in folgende 4 Teile:

1. ➤ CAN-Master-Projektierung in WinCoCT und Export als wld-Datei
2. ➤ CAN-Master-Projektierung importieren
3. ➤ Projektierung der Module am Standard-Bus
4. ➤ Projektierung aller SPEED-Bus-Module als virtuelles PROFIBUS-Netzwerk.
Hierzu ist die SPEEDBUS.GSD erforderlich.

Hardwareaufbau

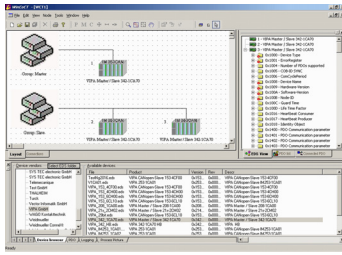



Voraussetzungen

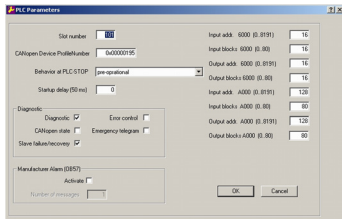
- Zur Projektierung eines CANopen-Systems ist die aktuellste EDS-Datei in das EDS-Verzeichnis von WinCoCT zu übertragen.
- Für den Einsatz der System 300S Module am SPEED-Bus ist die Einbindung der System 300S Module über die GSD-Datei SPEEDBUS.GSD von VIPA im Hardware-katalog erforderlich.

CAN-Master-Projektierung unter WinCoCT

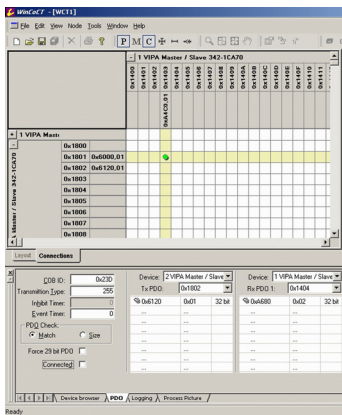
1. ➤ Kopieren Sie die erforderlichen EDS-Dateien in das EDS-Verzeichnis und starten Sie WinCoCT.
2. ➤ Legen Sie mit eine "Master"-Gruppe an und fügen Sie mit einen CANopen-Master für SPEED-Bus ein.



3. ➤ Legen Sie mit **G** eine "Slave"-Gruppe an und fügen Sie mit  Ihre CANopen-Slaves hinzu.
4. ➤ Klicken mit der rechten Maustaste auf den entsprechenden Slave und fügen Sie über "Module" die entsprechenden Module hinzu.
5. ➤ Parametrieren Sie Ihre Module mit [Parameter] bzw. über das entsprechende Objekt-Verzeichnis



6. ➤ Klicken Sie mit der rechten Maustaste auf den Master und öffnen Sie den VIPA-spezifischen Dialog "Set PLC Parameters". Hier können Sie das Diagnose-Verhalten einstellen und die Adress-Bereiche vorgeben, die vom Master in der CPU belegt werden. Unter "Slot number" geben Sie Ihre SPEED-Bus-Slot-Nr. addiert mit 100 an (101 ... 110), auf der Ihr CAN-Master gesteckt ist. Hieraus generiert WinCoCT beim Export die entsprechende DB-Nr. + 2000.



7. ➤ Wechseln Sie im Hauptfenster in das Register "Connections". Hier werden die Prozessdaten als Eingänge (1. Spalte) und als Ausgänge (1. Zeile) in einer Matrix dargestellt. Zur Anzeige der Prozessdaten eines Geräts, dem ein "+" vorangestellt ist, klicken Sie auf das entsprechende Gerät.
8. ➤ Zu Ihrer Hilfe können Sie immer nur dann eine Verbindung definieren, wenn das Fadenkreuz grün erscheint. Bereits projektierte PDOs werden über ein gelbes Fadenkreuz bzw. über eine grüne Markierung dargestellt. Stellen Sie mit der Maus in Zeile und Spalte der Matrix die entsprechende Zelle ein und klicken Sie mit der linken Maustaste

→ jetzt können Sie im zugehörigen PDO-Fenster Ihre Verbindung projektieren. Sie können die projektierte Verbindung überprüfen, indem Sie wieder in "Layout" wechseln, auf den Master klicken und über "Process Picture" das Prozessabbild des Masters ausgeben.

9. ➤ Speichern Sie Ihr Projekt.
10. ➤ Über **File > Export** wird Ihr CANopen-Projekt in eine wld-Datei exportiert. Der Name setzt sich zusammen aus Projektname + Knotenadresse + Kennung **Master/Slave**.
11. ➤ Aus diesen wld-Dateien ist der entsprechende Datenbaustein in das zugehörige SPS-Programm zu importieren. Die Vorgehensweise hierzu finden Sie auf der Folgeseite.
⇒ Hiermit ist die CANopen-Projektierung unter WinCoCT abgeschlossen.

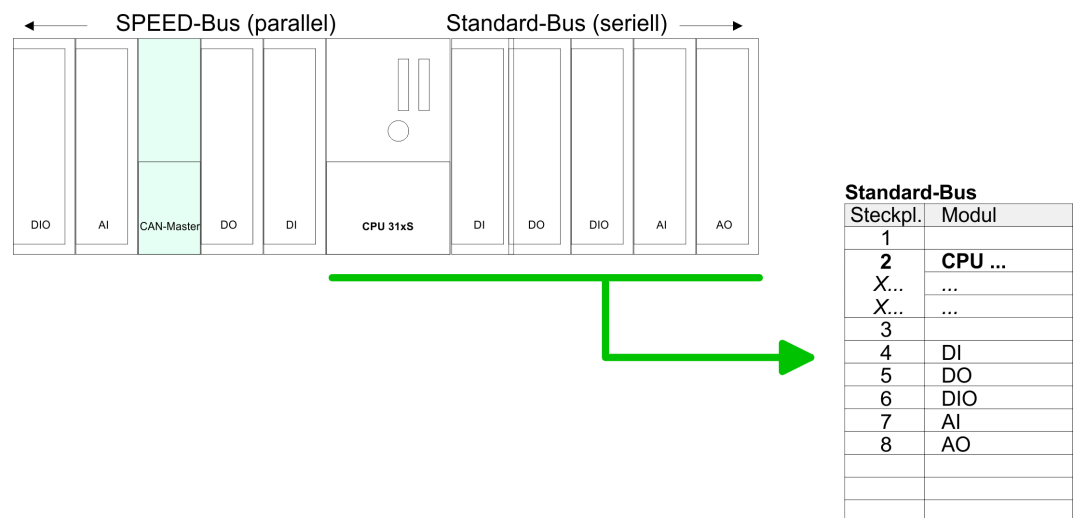
Import in SPS-Programm

1. ➤ Starten Sie den Siemens SIMATIC Manager mit einem neuen Projekt. Öffnen Sie den Hardware-Konfigurator und fügen Sie aus dem Hardware-Katalog eine Profilschiene ein.
2. ➤ Platzieren Sie auf Steckplatz 2 die entsprechende Siemens CPU.
3. ➤ Öffnen Sie die wld-Datei mit **Datei > Memory Card Datei > öffnen**
4. ➤ Kopieren Sie den DB 2xxx in Ihr Bausteine-Verzeichnis
⇒ Sobald Sie diesen Baustein an Ihre SPEED7-CPU übertragen, wird dieser von der CPU erkannt und die entsprechenden Parameter werden an den gewünschten CAN-Master übertragen. Dies ist aber nur möglich, wenn Ihr CAN-Master-Modul in der Hardware-Konfiguration am SPEED-Bus eingebunden wird. Die Vorgehensweise hierzu sehen Sie auf den Folgeseiten.

Projektierung der Module am Standard-Bus

Die am Standard-Bus rechts der CPU befindlichen Module sind nach folgenden Vorgehensweisen zu projektieren:

1. Starten Sie den Hardware-Konfigurator von Siemens mit einem neuen Projekt und fügen Sie aus dem Hardware-Katalog eine Profilschiene ein.
2. Platzieren Sie auf Steckplatz 2 die entsprechende Siemens CPU.
3. Binden Sie unterhalb der zuvor projektierten CPU beginnend mit Steckplatz 4 Ihre System 300V Module auf dem Standard-Bus in der gesteckten Reihenfolge ein.
4. Parametrieren Sie ggf. CPU bzw. die Module. Das Parameterfenster wird geöffnet, sobald Sie auf das entsprechende Modul doppelklicken.
5. Zur Buserweiterung können Sie die IM 360 von Siemens einsetzen, an die Sie bis zu 3 Erweiterungs-Racks über die IM 361 anbinden können. Buserweiterungen dürfen immer nur auf Steckplatz 3 platziert werden.
6. Sichern Sie Ihr Projekt.



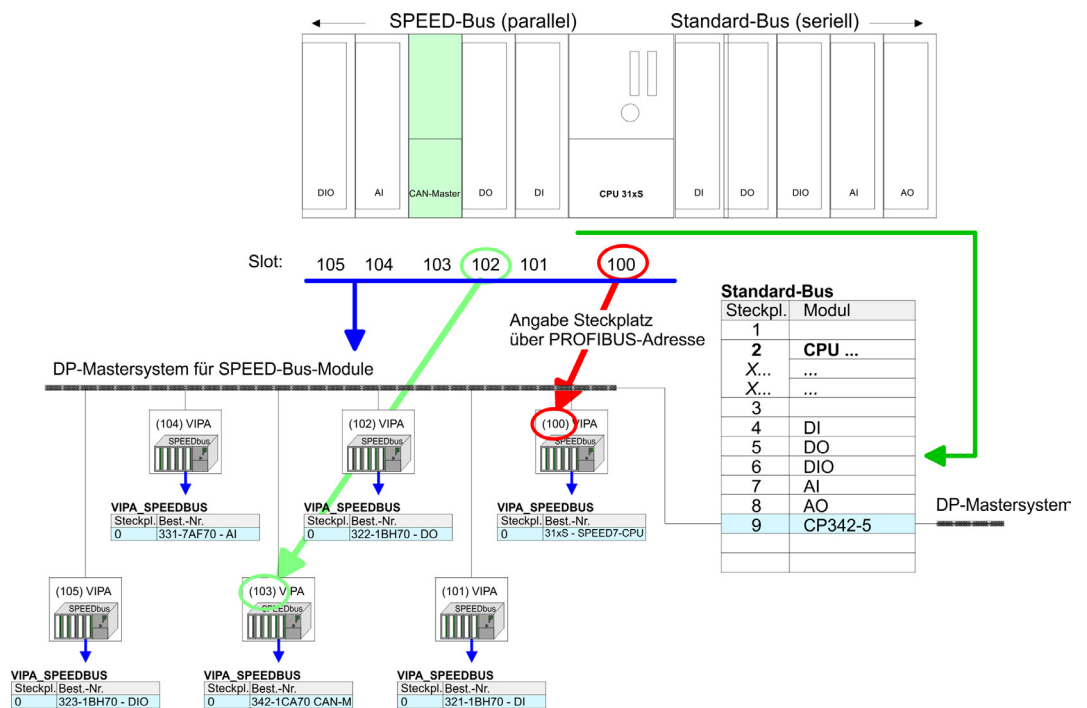
Projektierung aller SPEED-Bus-Module in einem virtuellen Master-System

Die Steckplatzzuordnung der CPU mit ihren SPEED-Bus-Modulen und die Parametrierung der Ein-/Ausgabe-Peripherie hat über ein virtuelles PROFIBUS-DP-Master-System zu erfolgen. Platzieren Sie hierzu immer als letztes Modul einen DP-Master (342-5DA02 V5.0) mit Mastersystem. Für den Einsatz der System 300S Module am SPEED-Bus ist die Einbindung der System 300S Module über die GSD-Datei SPEEDBUS.GSD von VIPA im Hardwarekatalog erforderlich. Nach der Installation der SPEEDBUS.GSD finden Sie unter *Profibus-DP / Weitere Feldgeräte / I/O / VIPA_SPEEDBUS* das DP-Slave-System *VIPA_SPEEDBUS*. Binden Sie nun für die CPU und jedes Modul am SPEED-Bus ein Slave-System "VIPA_SPEEDBUS" an. Stellen Sie als PROFIBUS-Adresse die Steckplatz-Nr. (100...110) des Moduls ein und platzieren Sie auf Steckplatz 0 des Slave-Systems das entsprechende Modul aus dem Hardwarekatalog von *VIPA_SPEEDBUS*.



VORSICHT!

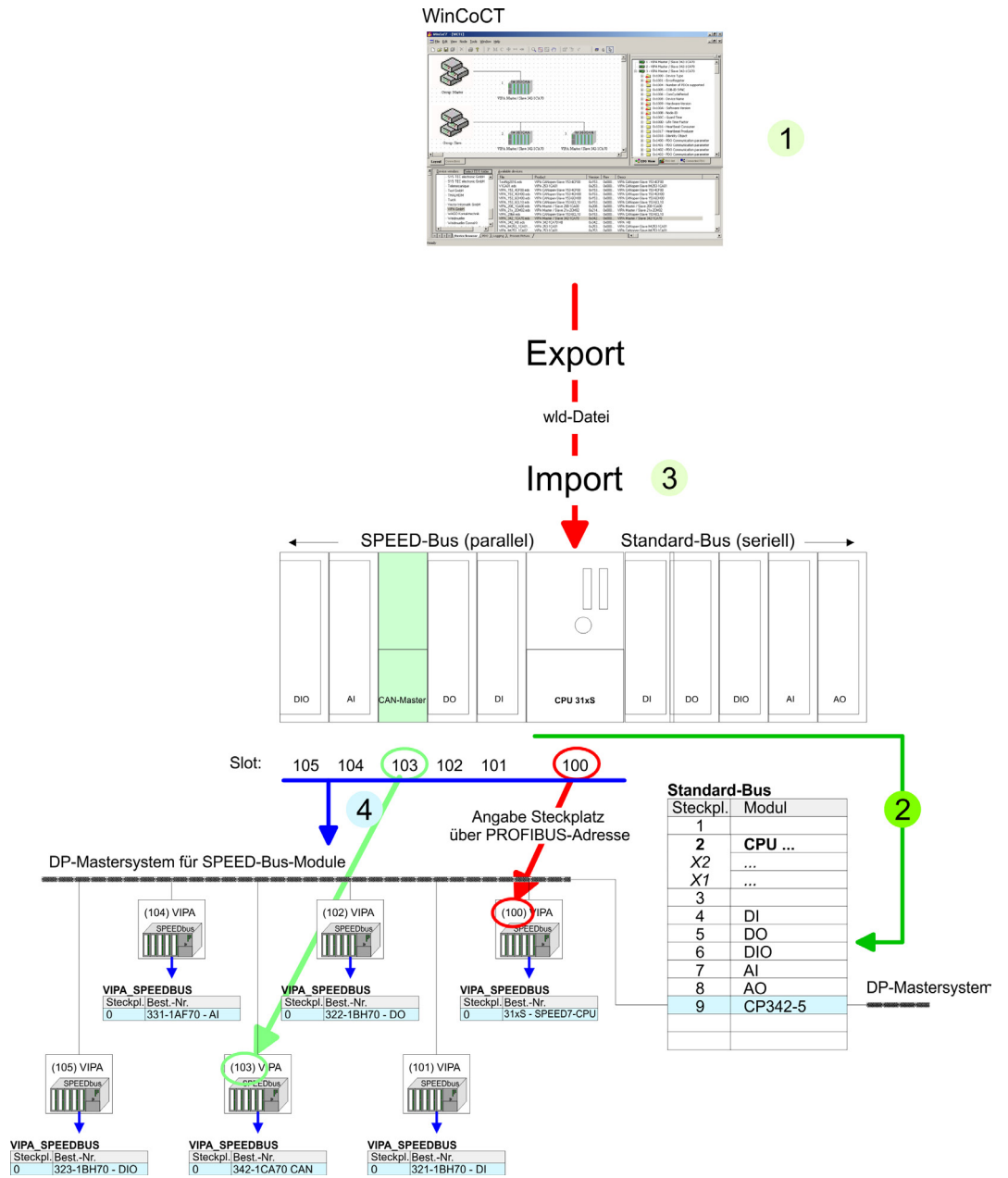
Bitte beachten Sie, dass Sie bei Anbindungen über externe PROFIBUS-DP-Master - zur Projektierung eines SPEED-Bus-Systemes erforderlich - keine Adressdoppelbelegung projektieren! Der Siemens Hardware-Konfigurator führt bei externen DP-Master-Systemen keine Adressüberprüfung durch!



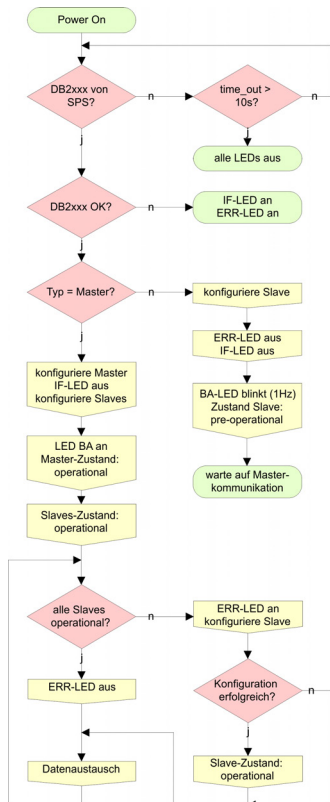
- Das entsprechende Modul ist aus dem HW-Katalog von *VIPA_SPEEDBUS* auf Steckplatz 0 zu übernehmen.
- Zusammen mit Ihrer Hardware-Konfiguration können Sie Ihre DP-Master-Projektierung in die CPU übertragen. Diese reicht die Projektierung an den CAN-Master weiter.

Zusammenfassung

In der nachfolgenden Abbildung sind alle Projektierschritte nochmals zusammengefasst:



5.5 Betriebsarten



■ STOP → RUN (automatisch)

- Nach NETZ EIN und bei gültigen Projektierdaten in der CPU geht der Master automatisch in RUN über. Auf einen Betriebsarten-Schalter für den Master wurde verzichtet.
- Nach einem NETZ EIN werden automatisch die Projektierdaten von der CPU an den CAN-Master geschickt. Dieser baut eine Kommunikation zu den CAN-Slaves auf.
- Bei erfolgter Kommunikation und gültigen Bus-Parametern, geht der CAN-Master in den Zustand "operational" über. Die LEDs RUN und BA leuchten.
- Bei fehlerhaften Parametern bleibt der CAN-Master in STOP und zeigt über die IF-LED einen Parametrierfehler an.

■ RUN

- Im RUN leuchten die RUN- und BA-LEDs. Jetzt können Daten ausgetauscht werden.
- Im Fehlerfall wie z.B. Slave-Ausfall, wird dies am CAN-Master über die ERR-LED angezeigt und ein Alarm an die CPU abgesetzt.

5.6 Prozessabbild

Das Prozessabbild setzt sich aus folgenden Teilen zusammen:

- Prozessabbild für Eingangs-Daten (PAE) für RPDOs
- Prozessabbild für Ausgangsdaten (PAA) für TPDOs

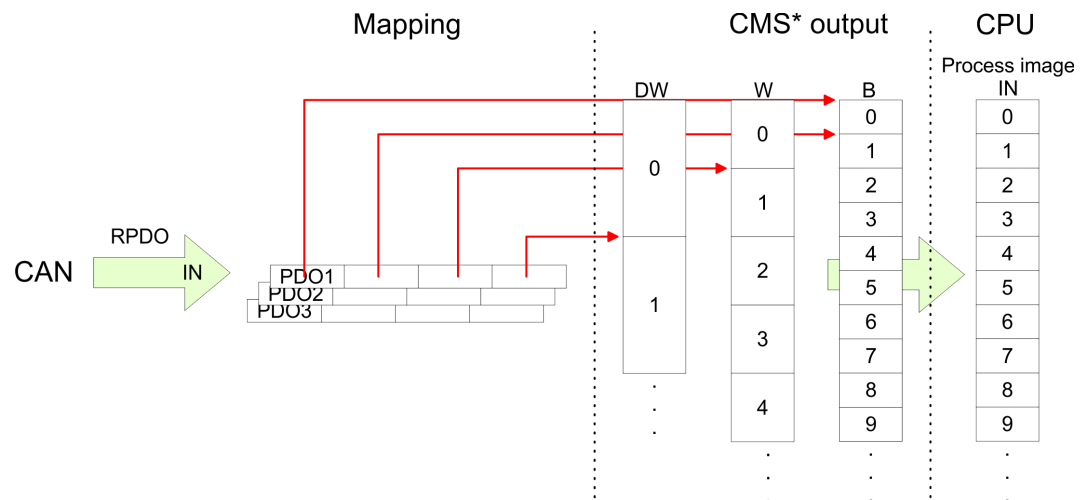
Hiervon besteht jeder Teil aus einem 320Byte "Digital-Data"- und 320Byte "Network Variables"-Bereich.

Prozesseingangsabbild

CANopen-Eingangs-Objekte:

- 8 Bit Digitale Eingabe (Objekt 0x6000)
- 16 Bit Digitale Eingabe (Objekt 0x6100)
- 32 Bit Digitale Eingabe (Objekt 0x6120)
- 8 Bit Eingangs-Netzwerk-Variablen (Objekt 0xA040)
- 16 Bit Eingangs-Netzwerk-Variablen (Objekt 0xA100)
- 32 Bit Eingangs-Netzwerk-Variablen (Objekt 0xA200)
- 64 Bit Eingangs-Netzwerk-Variablen (Objekt 0xA440)

Wie in der nachfolgenden Abbildung zu erkennen ist, wird für die verschiedenen CANopen-Objekte der gleiche Speicherbereich in der CPU verwendet. Beispielsweise würde ein Zugriff auf Index 0x6000 mit Subindex 2 einem Zugriff auf Index 0x6100 mit Subindex 1 entsprechen. Beide Objekte belegen die gleiche Speicherzelle in der CPU. Bitte beachten Sie, dass auch die Eingangs-Netzwerk-Variablen den gleichen Speicherbereich benutzen.



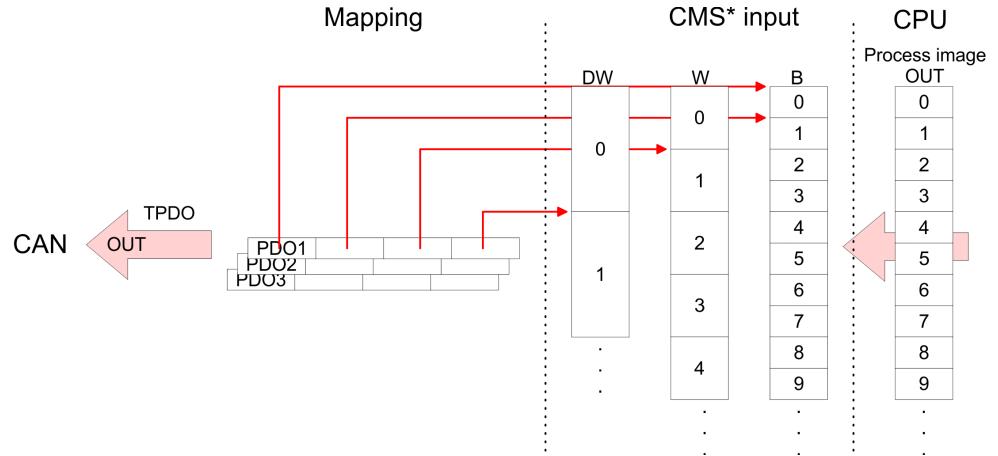
*) CMS = **C**ANopen **M**aster/**S**lave

Prozessausgangsabbild

CANopen-Ausgangs-Objekte:

- 8 Bit Digitale Ausgabe (Objekt 0x6200)
- 16 Bit Digitale Ausgabe (Objekt 0x6300)
- 32 Bit Digitale Ausgabe (Objekt 0x6320)
- 8 Bit Ausgangs-Netzwerk-Variablen (Objekt 0xA400)
- 16 Bit Ausgangs-Netzwerk-Variablen (Objekt 0xA580)
- 32 Bit Ausgangs-Netzwerk-Variablen (Objekt 0xA680)
- 64 Bit Ausgangs-Netzwerk-Variablen (Objekt 0xA8C0)

Wie in der nachfolgenden Abbildung zu erkennen ist, wird für die verschiedenen CANopen-Objekte der gleiche Speicherbereich in der CPU verwendet. Beispielsweise würde ein Zugriff auf Index 0x6200 mit Subindex 2 einem Zugriff auf Index 0x6300 mit Subindex 1 entsprechen. Beide Objekte belegen die gleiche Speicherzelle in der CPU. Bitte beachten Sie, dass auch die Ausgangs-Netzwerk-Variablen den gleichen Speicherbereich benutzen.



*) CMS = CANopen Master/Slave

5.7 Telegrammaufbau

Identifier

Alle CANopen Telegramme besitzen nach CiA DS-301 folgenden Aufbau:

| Byte | Bit 7 ... Bit 0 |
|------|--|
| 1 | <ul style="list-style-type: none"> ■ Bit 3 ... Bit 0: Höchstwertige 4 Bits der Modul-ID ■ Bit 7 ... Bit 4: CANopen Funktionscode |
| 2 | <ul style="list-style-type: none"> ■ Bit 3 ... Bit 0: Datenlänge (DLC) ■ Bit 4: RTR-Bit: <ul style="list-style-type: none"> – 0: keine Daten (Anforderungstelegramm) – 1: Daten vorhanden ■ Bit 7 ... Bit 5: Niederwertige 3 Bits der Modul-ID |

Data

| Byte | Bit 7 ... Bit 0 |
|----------|-----------------|
| 3 ... 10 | Daten |

Der Unterschied zu einem Schicht-2-Telegramm besteht in einer zusätzlichen Unterteilung des 2 Byte Identifiers in einen Funktionsteil und eine Modul-ID. Im Funktionsteil wird die Art des Telegramms (Objekt) festgelegt und mit der Modul-ID wird der Empfänger adressiert. Der Datenaustausch bei CANopen-Geräten erfolgt in Form von Objekten. Im CANopen-Kommunikationsprofil sind zwei Objektarten sowie einige Spezialobjekte definiert.

Der Yaskawa CAN-Master für SPEED-Bus unterstützt folgende Objekte:

- 40 Transmit PDOs (PDO Linking, PDO Mapping)
- 40 Receive PDOs (PDO Linking, PDO Mapping)
- 2 Standard SDOs (1 Server, 127 Clients)
- 1 Emergency Objekt
- 1 Netzwerkmanagement Objekt NMT

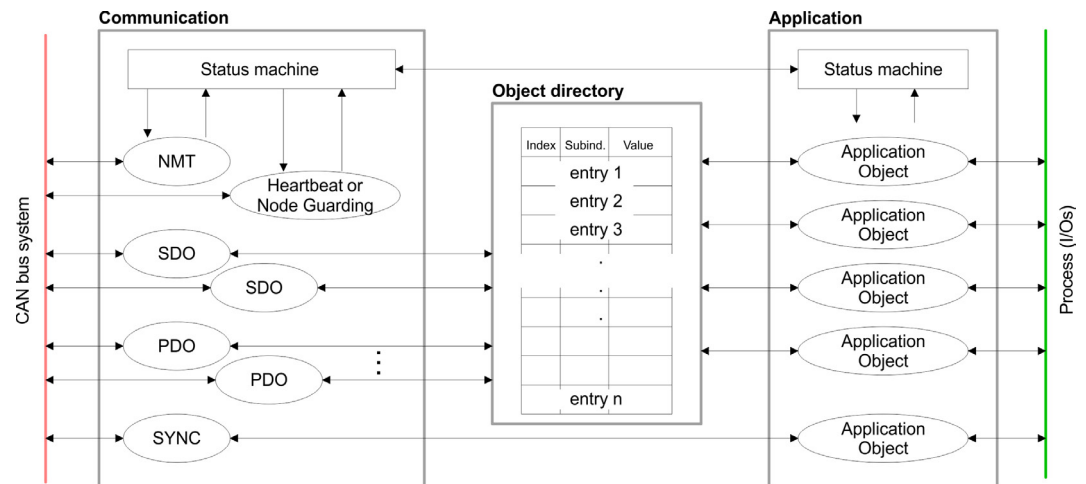
- Node Guarding
- Heartbeat



Der genaue Aufbau und Dateninhalt aller Objekte ist in den CiA-Profilen DS-301, DS-302, DS-401 und DS-405 beschrieben.

Struktur des Gerätemodells

Ein CANopen Gerät kann wie folgt strukturiert werden:



- Communication** - Stellt die Kommunikationsdatenobjekte und die zugehörige Funktionalität zum Datenaustausch über das CANopen Netzwerk zur Verfügung.
- Application** - Die Applikationsdatenobjekte enthalten z.B. Ein- und Ausgangsdaten. Eine Applikationsstatusmaschine überführt die Ausgänge im Fehlerfall in einen sicheren Zustand.
- Object directory** - Das Objektverzeichnis ist wie eine zweidimensionale Tabelle organisiert. Die Daten werden über Index und Subindex adressiert. Es enthält alle Datenobjekte (Applikationsdaten + Parameter), die von außen zugänglich sind und die das Verhalten von Kommunikation, Applikation und Statusmaschinen beeinflussen.

PDO

Bei vielen Feldbussystemen wird ständig das gesamte Prozessabbild übertragen - meist mehr oder weniger zyklisch. CANopen ist nicht auf dieses Kommunikationsprinzip beschränkt, da CAN durch die Multi-Master Buszugriffsregelung andere Möglichkeiten bietet. Bei CANopen werden die Prozessdaten in Segmente zu maximal 8Byte aufgeteilt. Diese Segmente heißen **Prozessdaten-Objekte (PDOs)**. Die PDOs entsprechen jeweils einem CAN-Telegramm und werden über dessen spezifischen CAN-Identifizier zugeordnet und in ihrer Priorität bestimmt. Für den Prozessdatenaustausch stehen beim CAN-Master für Ein- und Ausgabe jeweils 40Tx Transmit-PDOs bzw. 40Rx Receive-PDOs zur Verfügung. Jedes PDO besteht dabei aus maximal 8 Datenbytes. PDOs werden unbestätigt übertragen, da das CAN-Protokoll die Übertragung sicherstellt.

Die PDOs werden aus Sicht des CAN-Masters bezeichnet:

- Receive-PDOs (RxPDOs) werden vom CAN-Master empfangen und enthalten Eingangsdaten, die im PAE (**P**rozessabbild der **E**ingänge) abgelegt werden.
- Transmit-PDOs (TxPDOs) werden vom CAN-Master gesendet und enthalten Ausgangsdaten, die im PAA (**P**rozessabbild der **A**usgänge) liegen.

Die Belegung dieser PDOs mit Ein- bzw. Ausgangsdaten erfolgt unter WinCoCT automatisch.

SDO

Für Zugriffe auf das Objektverzeichnis wird das **Service-Daten-Objekt** (SDO) verwendet. Mit dem SDO können Sie lesend oder schreibend auf das Objektverzeichnis zugreifen. Im CAL-Schicht-7-Protokoll finden Sie die Spezifikation des Multiplexed-Domain-Transfer-Protocols, das von den SDOs genutzt wird. Mit diesem Protokoll können Sie Daten beliebiger Länge übertragen. Hierbei werden Nachrichten gegebenenfalls auf mehrere CAN-Nachrichten mit gleichem Identifier aufgeteilt (Segmentierung). Ein SDO wird bestätigt übertragen, d.h. jeder Empfang einer Nachricht wird quittiert.



Eine nähere Beschreibung der SDO-Telegramme finden sie in der vom CiA verfassten DS-301 Norm. Nachfolgend sollen lediglich die Fehlermeldungen aufgeführt werden, die im Falle einer fehlerhaften Parameterkommunikation erzeugt werden.

FC/SFC 219 CAN_TLGR

Jede SPEED7-CPU hat den FC/SFC 219 integriert. Hiermit können Sie von Ihrem SPS-Programm auf Ihrem CAN-Master einen SDO Lese- oder Schreibzugriff auslösen. Hierbei adressieren Sie den Master über die Steckplatz-Nr. und den Ziel-Slave über seine CAN-Adresse. Die Prozessdaten bestimmen Sie durch Angabe von INDEX und SUBINDEX. Über SDO kann pro Zugriff maximal ein Datenwort Prozessdaten übertragen werden.



Näheres zum Einsatz dieser Bausteine finden Sie im Handbuch "SPEED7 Operationsliste" von Yaskawa.

5.8 Objekt-Verzeichnis**Struktur**

- Im CANopen-Objektverzeichnis werden alle für den CP relevanten CANopen Objekte eingetragen. Jeder Eintrag im Objektverzeichnis ist durch einen 16Bit-Index gekennzeichnet.
- Falls ein Objekt aus mehreren Komponenten besteht (z.B. Objekttyp Array oder Record), sind die Komponenten über einen 8Bit-Subindex gekennzeichnet.
- Der Objektname beschreibt die Funktion eines Objekts. Das Datentyp-Attribut spezifiziert den Datentyp des Eintrags.
- Über das Zugriffsattribut ist spezifiziert, ob ein Eintrag nur gelesen werden kann, nur geschrieben werden oder gelesen und geschrieben werden darf.

Das Objektverzeichnis ist in folgende 3 Bereiche aufgeteilt:

- **Kommunikationsspezifischer Profilbereich (0x1000 – 0x1FFF)**
 - Dieser Bereich beinhaltet die Beschreibung aller spezifischen Parameter für die Kommunikation.

| | |
|-----------------|--|
| 0x1000 – 0x1018 | allgemeine kommunikationsspezifische Parameter (z.B. der Gerätename) |
| 0x1400 – 0x1427 | Kommunikationsparameter (z.B. Identifier) der Receive-PDOs |
| 0x1600 – 0x1627 | Mappingparameter der Receive-PDOs Die Mappingparameter enthalten die Querverweise auf die Applikationsobjekte, die in die PDOs gemappt sind und die Datenbreite des entsprechenden Objektes |
| 0x1800 – 0x1827 | Kommunikations- und Mappingparameter der Transmitt-PDOs |
| 0x1A00 – 0x1A27 | |

- **Herstellerspezifischer Profilbereich (0x2000 – 0x5FFF)**
 - Hier finden Sie die herstellerspezifischen Einträge wie z.B. PDO-Control, CAN-Übertragungsrate (Übertragungsrate nach RESET) usw.
- **Standardisierter Geräteprofilbereich (0x6000 – 0x9FFF)**
 - In diesem Bereich liegen die Objekte für das Geräteprofil nach DS-401.



Da die CiA Normen ausschließlich in englischer Sprache vorliegen, wurden die Tabelleneinträge der Objekte zum eindeutigen Verständnis in englischer Sprache übernommen. Eine nähere Beschreibung der Tabelleneinträge in Deutsch finden Sie jeweils unterhalb der Tabellen.

Objektverzeichnis Übersicht

| Index | Content of Object |
|-----------------|--------------------------------------|
| 0x1000 | Device type |
| 0x1001 | Error register |
| 0x1005 | COB-ID SYNC |
| 0x1006 | Communication Cycle Period |
| 0x1007 | Synchronous Window Length |
| 0x1008 | Manufacturer Hardware Version |
| 0x1009 | Hardware version |
| 0x100A | Software version |
| 0x100C | Guard time |
| 0x100D | Life time factor |
| 0x1016 | Consumer Heartbeat Time |
| 0x1017 | Producer Heartbeat Time |
| 0x1018 | Identity Object |
| 0x1400 - 0x1427 | Receive PDO Communication Parameter |
| 0x1600 - 0x1627 | Receive PDO Mapping Parameter |
| 0x1800 - 0x1827 | Transmit PDO Communication Parameter |
| 0x1A00 - 0x1A27 | Transmit PDO Mapping Parameter |
| 0x1F22 | Concise DCF |
| 0x1F25 | Post Configuration |
| 0x1F80 | NMT StartUp |
| 0x1F81 | Slave Assignment |
| 0x1F82 | Request NMT |
| 0x1F83 | Request Guarding |
| 0x2000 | Initialize Rx-COB-ID for OB57 |
| 0x2001 | Node-ID - PLC-STOP |
| 0x2002 | Node-ID - PLC-RUN |
| 0x2003 | Start address RxPDO-Counter |
| 0x2004 | Start address NG/HB- ToggleBit |

| Index | Content of Object |
|--------|--|
| 0x2005 | Start address L2-Message-Area |
| 0x2016 | Lenze NodeGuarding |
| 0x2100 | Message PLC-RUN |
| 0x2101 | Message PLC-STOP |
| 0x2200 | J1939: PGN for Multipaket Transfer |
| 0x3000 | Special settings for CAN |
| 0x6000 | Digital-Input-8-Bit Array (see DS 401) |
| 0x6100 | Digital-Input-16-Bit Array (see DS 401) |
| 0x6120 | Digital-Input-32Bit Array (see DS 401) |
| 0x6200 | Digital-Output-8-Bit Array (see DS 401) |
| 0x6300 | Digital-Output-16-Bit Array (see DS 401) |
| 0x6320 | Digital-Output-32-Bit Array (see DS 401) |
| 0xA040 | Dynamic Unsigned8 Input |
| 0xA100 | Dynamic Unsigned16 Input |
| 0xA200 | Dynamic Unsigned32 Input |
| 0xA440 | Dynamic Unsigned64 Input |
| 0xA4C0 | Dynamic Unsigned8 Output |
| 0xA580 | Dynamic Unsigned16 Output |
| 0xA680 | Dynamic Unsigned32 Output |
| 0xA8C0 | Dynamic Unsigned64 Output |

Device Type

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|------------|------------|-------|------|---------------|--------------------------|
| 0x1000 | 0 | DeviceType | Unsigned32 | ro | N | 0x00050191 | Statement of device type |

Der 32Bit-Wert ist in zwei 16Bit-Felder unterteilt:

| High word | Low word |
|--------------------------------------|-----------------------|
| Additional information Device | profile number |
| 0000 0000 0000 wxyz (bit) | 405dec=0x0195 |

Die "Additional Information" enthält Angaben über die Signalarten des I/O-Gerätes:

- z=1 → digitale Eingänge
- y=1 → digitale Ausgänge
- x=1 → analoge Eingänge
- w=1 → analoge Ausgänge

Error register

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|----------------|-----------|-------|------|---------------|----------------|
| 0x1001 | 0 | Error Register | Unsigned8 | ro | Y | 0x00 | Error register |

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---------|----------|----------|-------|----------|----------|----------|---------|
| ManSpec | reserved | reserved | Comm. | reserved | reserved | reserved | Generic |

- *ManSpec*:
 - Herstellerspezifischer Fehler, wird in Objekt 0x1003 genauer spezifiziert.
- *Comm.*:
 - Kommunikationsfehler (Overrun CAN)
- *Generic*:
 - Ein nicht näher spezifizierter Fehler ist aufgetreten (Flag ist bei jeder Fehlermeldung gesetzt)

SYNC identifier

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|--------------------|------------|-------|------|---------------|--------------------------------|
| 0x1005 | 0 | COB-Id syncmessage | Unsigned32 | ro | N | 0x00000080 | Identifier of the SYNC message |

Die unteren 11Bit des 32Bit Wertes enthalten den Identifier (0x80 = 128dez).

Bit 30 = 0: Slave arbeitet als Sync Consumer (0x00000080)

Bit 30 = 1: Slave arbeitet als Sync Producer (0x40000080)

SYNC interval

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|----------------------------|------------|-------|------|---------------|---|
| 0x1006 | 0 | Communication cycle period | Unsigned32 | rw | N | 0x00000000 | Maximum length of the SYNC interval in μ s. |

Wenn hier ein Wert ungleich Null eingetragen wird, so geht der Koppler in den Fehlerzustand, wenn beim synchronen PDO-Betrieb innerhalb der "Watchdog-Zeit" kein SYNC-Telegramm empfangen wurde.

Synchronous Window Length

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|---------------------------|------------|-------|------|---------------|---|
| 0x1007 | 0 | Synchronous window length | Unsigned32 | rw | N | 0x00000000 | Contains the length of time window for synchronous PDOs in μ s. |

Device name

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|--------------------------|----------------|-------|------|---------------|-----------------------|
| 0x1008 | 0 | Manufacturer device name | Visible string | ro | N | | Device name of the CP |

- Yaskawa 342-1CA70 = Yaskawa CANopen Master/Slave 342-1CA70
- Da der zurückgelieferte Wert größer als 4Byte ist, wird das segmentierte SDO-Protokoll zur Übertragung verwendet.

Hardware version

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|-------------------------------|----------------|-------|------|---------------|-------------------------------|
| 0x1009 | 0 | Manufacturer Hardware version | Visible string | ro | N | | Hardware version number of CP |

- Yaskawa 342-1CA70 = 1.00
- Da der zurückgelieferte Wert größer als 4Byte ist, wird das segmentierte SDO-Protokoll zur Übertragung verwendet.

Software version

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|-------------------------------|----------------|-------|------|---------------|--|
| 0x100A | 0 | Manufacturer Software version | Visible string | ro | N | | Software version number CANopen software |

- Yaskawa 342-1CA70 = 1.07
- Da der zurückgelieferte Wert größer als 4Byte ist, wird das segmentierte SDO-Protokoll zur Übertragung verwendet.

Guard time

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|----------------|------------|-------|------|---------------|---|
| 0x100C | 0 | Guard time[ms] | Unsigned16 | rw | N | 0x0000 | Interval between two guard telegrams. Is set by the NMT master or configuration tool. |

Life time factor

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|------------------|-----------|-------|------|---------------|---|
| 0x100D | 0 | Life time factor | Unsigned8 | rw | N | 0x00 | Life time factor x guard time = life time (watchdog for lifeguarding) |

Wenn innerhalb der Life Time kein Guarding-Telegramm empfangen wurde, geht der Knoten in den Fehlerzustand. Wenn "Life Time Factor" und/oder "Guard Time" = 0 sind, so führt der Knoten kein Lifeguarding durch, kann aber dennoch vom Master überwacht werden (Node Guarding).

Consumer Heartbeat Time

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|--------------------------|------------|-------|------|---------------|-------------------------|
| 0x1016 | 0 | Consumer heart-beat time | Unsigned8 | ro | N | 0x05 | Number of entries |
| | 1...127 | | Unsigned32 | rw | N | 0x00000000 | Consumer heartbeat time |

Struktur des "Consumer Heartbeat Time" Eintrags:

| | | | |
|------------|-----------|-----------|----------------|
| Bits | 31-24 | 23-16 | 15-0 |
| Value | Reserved | Node-ID | Heartbeat time |
| Encoded as | Unsigned8 | Unsigned8 | Unsigned16 |

Sobald Sie versuchen, für die gleiche Node-ID eine "consumer heartbeat time" ungleich 0 zu konfigurieren, bricht der Knoten den SDO-Download ab und bringt den Fehlercode 0604 0043h.

Producer Heartbeat Time

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|--------------------------|------------|-------|------|---------------|---|
| 0x1017 | 0 | Producer heart-beat time | Unsigned16 | rw | N | 0x0000 | Defines the cycle time of heartbeat in ms |

Identity Object

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|-----------------|------------|-------|------|---------------|---|
| 0x1018 | 0 | Identity Object | Unsigned8 | ro | N | 0x04 | Contains general information about the device (number of entries) |
| | 1 | Vendor ID | Unsigned32 | ro | N | 0xAFFEAF | Vendor ID |
| | 2 | Product Code | Unsigned32 | ro | N | 0x3421CA70 | Product Code |
| | 3 | Revision Number | Unsigned32 | ro | N | | Revision Number |
| | 4 | Serial Number | Unsigned32 | ro | N | | Serial Number |

**Communication parameter
RxPDO1**

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|---------------------|-----------|--------------------|------------|-------|------|-------------------------|---|
| 0x1400... 0x1427 | 0 | Number of Elements | Unsigned8 | ro | N | 0x02 | Communication parameter for the first receive PDOs, sub-index 0: number of following parameters |
| | 1 | COB-ID | Unsigned32 | rw | N | 0xC0000200 + NODE_ID | COB-ID RxPDO1 |
| | 2 | Transmission type | Unsigned8 | rw | N | 0xFF | Transmission type of the PDO |

Subindex 1 (COB-ID): Die unteren 11Bit des 32Bit-Wertes (Bits 0-10) enthalten den CAN-Identifizier, das MSBit (Bit 31) gibt Auskunft, ob das PDO aktiv ist (0) oder nicht (1), Bit 30 teilt mit, ob ein RTR-Zugriff auf dieses PDO zulässig ist (0) oder nicht (1).

Der Subindex 2 enthält die Übertragungsart.

Mapping RxPDO1

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|--------------------|------------|-------|------|---------------|---|
| 0x1600 | 0 | Number of Elements | Unsigned8 | rw | N | 0x01 | Mapping parameter of the first receive PDO; sub-index 0: number of mapped objects |
| | 1 | 1. mapped object | Unsigned32 | rw | N | 0x62000108 | (2 byte index, 1 byte sub-index, 1 byte bit-width) |
| | 2 | 2. mapped object | Unsigned32 | rw | N | 0x62000208 | (2 byte index, 1 byte sub-index, 1 byte bit-width) |
| | ... | ... | ... | ... | ... | ... | ... |
| | 8 | 8. mapped | Unsigned32 | rw | N | 0x62000808 | (2 byte index, 1 byte sub-index, 1 byte bit-width) |

Das 1. Empfangs-PDO (RxPDO1) ist per Default für digitale Ausgänge vorgesehen. Je nach Anzahl der bestückten Ausgänge wird automatisch die erforderliche Länge des PDOs bestimmt und die entsprechenden Objekte gemappt.

Da die digitalen Ausgänge byteweise organisiert sind, kann die Länge des PDOs in Bytes direkt dem Subindex 0 entnommen werden. Wenn das Mapping verändert wird, so muss der Eintrag in Subindex 0 entsprechend angepasst werden.

Communication parameter TxPDO1

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|--------------------|------------|-------|------|----------------------|--|
| 0x1800 | 0 | Number of Elements | Unsigned8 | ro | N | 0x05 | Communication parameter of the first transmit PDO, sub-index 0: number of following parameters |
| | 1 | COB-ID | Unsigned32 | rw | N | 0x80000180 + NODE_ID | COB-ID TxPDO1 |
| | 2 | Transmission type | Unsigned8 | rw | N | 0xFF | Transmission type of the PDO |
| | 3 | Inhibit time | Unsigned16 | rw | N | 0x0000 | Repetition delay [value x 100 µs] |
| | 5 | Event time | Unsigned16 | rw | N | 0x0000 | Event timer [value x 1 ms] |

Subindex 1 (COB-ID): Die unteren 11Bit des 32Bit Wertes (Bits 0-10) enthalten den CAN-Identifizier, das MSBit (Bit 31) gibt Auskunft, ob das PDO aktiv ist (0) oder nicht (1), Bit 30 teilt mit, ob ein RTR-Zugriff auf dieses PDO zulässig ist (0) oder nicht (1). Der Subindex 2 enthält die Übertragungsart, Subindex 3 die Wiederholungsverzögerung zwischen zwei gleichen PDOs. Wenn ein "Event Timer" mit einem Wert ungleich 0 existiert, wird nach Ablauf dieses Timers das PDO übertragen. Existiert ein "Inhibit Timer", wird das Ereignis um diese Zeit verzögert.

Mapping TxPDO1

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|--------------------|------------|-------|------|------------------------------------|---|
| 0x1A00 | 0 | Number of Elements | Unsigned8 | rw | N | depending on the components fitted | Mapping parameter of the first transmit PDO; sub-index 0: number of mapped objects |
| | 1 | 1. mapped object | Unsigned32 | rw | N | 0x60000108 | (2 byte index, 1 byte sub-index, 1 byte bit-width) |
| | 2 | 2. mapped object | Unsigned32 | rw | N | 0x60000208 | (2 byte index, 1 byte sub-index, 1 byte bit-width) |
| | ... | ... | ... | ... | ... | ... | ... |
| | 8 | 8. mapped object | Unsigned32 | rw | N | 0x60000808 | (2 byte index, 1 byte sub-index, 1 byte bit-width) |

Das 1. Sende-PDO (TxPDO1) ist per Default für digitale Eingänge vorgesehen. Je nach Anzahl der bestückten Eingänge wird automatisch die erforderliche Länge des PDOs bestimmt und die entsprechenden Objekte gemappt. Da die digitalen Eingänge byteweise organisiert sind, kann die Länge des PDOs in Bytes direkt dem Subindex 0 entnommen werden. Wenn das Mapping verändert wird, muss der Eintrag in Subindex 0 entsprechend angepasst werden.

Concise DCF

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|-------------|--------|-------|------|---------------|---------|
| 0x1F22 | Array | Concise DCF | Domain | rw | N | | |

Dieses Objekt ist für den Configuration Manager erforderlich. Das Concise-DCF ist eine Kurzfassung des DCF (**D**evice **C**onfiguration **F**ile).

Post Configuration

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|----------------|------------|-------|------|---------------|---------|
| 0x1F25 | Array | ConfigureSlave | Unsigned32 | rw | N | 0x00000000 | |

Der Configuration Manager kann über diesen Eintrag angewiesen werden, eine gespeicherte Konfiguration in das Netz zu übertragen. Die Konfiguration kann zu jeder Zeit über Index 0x1F25 für einen bestimmten Knoten ausgelöst werden.

- Subindex 0 hat den Wert 128.
- Subindex x (mit x = 1..127):
Löst Rekonfiguration für Knoten mit der Node-ID x aus.
- Subindex 128: Rekonfiguration aller Knoten.
Soll z.B. für den Knoten 2 die Konfiguration ausgelöst werden und sind für diesen Knoten Konfigurationsdaten vorhanden, so ist der Wert 0x666E6F63 (ASCII = "conf") auf das Objekt 1F25h Subindex 2 zu schreiben.

NMT Start-up

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|------------|------------|-------|------|---------------|---------|
| 0x1F80 | 0x00 | NMTStartup | Unsigned32 | rw | N | 0x00000000 | |

Hier geben Sie an, ob das Gerät der NMT-Master ist.

| Bit | Meaning |
|-----------|---|
| Bit 0 | <ul style="list-style-type: none"> ■ 0: Device is NOT the NMT Master. All other bits have to be ignored. The objects of the Network List have to be ignored. ■ 1: Device is the NMT Master. |
| Bit 1 | <ul style="list-style-type: none"> ■ 0: Start only explicitly assigned slaves. ■ 1: After boot-up perform the service NMT Start Remote Node All Nodes. |
| Bit 2..31 | Reserved by CiA, always 0 |

Slave Assignment

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|-----------------|------------|-------|------|---------------|---------|
| 0x1F81 | 0x00 | SlaveAssignment | Unsigned32 | rw | N | 0x00000000 | |

- Hier erfolgt ein Eintrag der Knoten, die vom Master überwacht, kontrolliert und gesteuert werden sollen.
- Für jeden zugeordneten Knoten ist hier ein Eintrag vorzunehmen.
- Subindex 0 hat den Wert 127.
- Jeder andere Subindex korrespondiert mit der Node-ID des Knotens.

| Byte | Bit | Meaning |
|-----------|-----------|--|
| Byte 0 | Bit 0 | <ul style="list-style-type: none"> ■ 0: Node with this ID is not a slave ■ 1: Node with this ID is a slave. After configuration (with Configuration Manager) the Node will be set to state operational. |
| | Bit 1 | <ul style="list-style-type: none"> ■ 0: On Error Control Event or other detection of a booting slave inform the application. ■ 1: On Error Control Event or other detection of a booting slave inform the application and automatically start Error Control service. |
| | Bit 2 | <ul style="list-style-type: none"> ■ 0: On Error Control Event or other detection of a booting slave do NOT automatically configure and start the slave. ■ 1: On Error Control Event or other detection of a booting slave do start the process Start Boot Slave. |
| | Bit 7.. 3 | Reserved by CiA, always 0 |
| Byte 1 | | 8 Bit Value for the RetryFactor |
| Byte 2, 3 | | 16 Bit Value for the GuardTime |

Request NMT

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|------------|-----------|-------|------|---------------|---------|
| 0x1F82 | 0x00 | RequestNMT | Unsigned8 | rw | N | 0x00000000 | |

Ist ein vollständig autonomer Start des Stacks nicht gewünscht, so können die Funktionalitäten:

- Statusumschaltung
- Starten des Guardings
- Konfiguration über CMT

auch für jeden Knoten einzeln auf Anfrage durchgeführt werden. Die Anfrage erfolgt immer über Objekte im Objektverzeichnis. Die Umschaltung des Kommunikationsstatus aller im Netz vorhandenen Knoten (einschließlich des lokalen Slaves) wird dabei über den Eintrag 1F82h im lokalen Objektverzeichnis bewerkstelligt:

- Subindex 0 hat den Wert 128
- Subindex x (with x = 1...127):
 - Löst NMT-Service für Knoten mit der Node-ID x aus.
- Subindex 128:
 - Löst NMT-Service für alle Knoten aus.

Beim Schreiben wird der gewünschte Status als Wert angegeben:

| State | Value |
|--------------------|-------|
| Prepared | 4 |
| Operational | 5 |
| ResetNode | 6 |
| ResetCommunication | 7 |
| Pre-operational | 127 |

Request Guarding

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|-----------------|------------|-------|------|---------------|---------|
| 0x1F83 | 0x00 | RequestGuarding | Unsigned32 | rw | N | 0x00000000 | |

- Subindex 0 hat den Wert 128.
- Subindex x (with x=1..127):
 - Löst Guarding für den Slave mit Node-ID x aus.

| Value | Write Access | Read Access |
|-------|----------------|-------------------------------|
| 1 | Start Guarding | Slave actually is guarded |
| 0 | Stop Guarding | Slave actually is not guarded |

- Subindex 128:
 - Request Start/Stop Guarding für alle Knoten.

Initialize Rx-COB-ID for OB 57

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|--------------------|------------|-------|------|---------------|------------------------------|
| 0x2000 | 0 | Number of elements | Unsigned8 | ro | N | 8 | Number of available entries. |
| | 1 | 1. COB-ID | Unsigned32 | rw | N | 0 | COB-ID which generates OB 57 |
| | 2 | 2. COB-ID | Unsigned32 | rw | N | 0 | COB-ID which generates OB 57 |
| | ... | ... | ... | ... | ... | ... | ... |
| | 8 | 8. COB-ID | Unsigned32 | rw | N | 0 | COB-ID which generates OB 57 |

Structur COB-ID

| | UNSIGNED32 | | | 28-11 | LSB |
|-----------|------------|-----|----|----------------------|-------------------|
| | MSB | | | | |
| Bits | 31 | 30 | 29 | 28-11 | 10-0 |
| 11-bit-ID | 0/1 | 0/1 | 0 | 00000000000000000000 | 11-bit Identifier |
| 29-bit-ID | 0/1 | 0/1 | 1 | 29-bit Identifier | |

| Bit number | Value | Meaning |
|------------|-------|--|
| 31 (MSB) | 0 | PDO exists / is valid |
| | 1 | PDO does not exist / is not valid |
| 30 | 0 | |
| | 1 | no RTR allowed on this PDO |
| 29 | 0 | 11-bit ID (CAN 2.0A) |
| | 1 | 29-bit ID (CAN 2.0B) |
| 28-11 | 0 | if bit 29=0 |
| | X | if bit 29=1: bits 28-11 of 29-bit-COB-ID |
| 10-0 (LSB) | X | bits 10-0 of COB-ID |

Node-ID - PLC-STOP

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|--------------------------|------------|-------|------|---------------|----------------------------------|
| 0x2001 | 0x00 | Number of elements | Unsigned8 | ro | N | 0 | Number of available entries. |
| | 0x01 | 1. Node-ID for PLC-STOP | Unsigned8 | rw | N | 0 | Node-ID (value range:1...127) |
| | ... | ... | ... | ... | ... | ... | ... |
| | 0x10 | 16. Node-ID for PLC-STOP | Unsigned32 | rw | N | 0 | COB-ID which generates OB 57 |

Beim Übergang von PLC-RUN → PLC-STOP werden die hier eingetragenen CAN-Geräte über das NMT-Kommando *Preoperational* in den Zustand Preoperational gesetzt.

Node-ID - PLC-Run

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|--------------------------|-----------|-------|------|---------------|-----------------------------------|
| 0x2002 | 0x00 | Number of elements | Unsigned8 | ro | N | 0 | Number of available entries. |
| | 0x01 | 1. Node-ID for PLC-STOP | Unsigned8 | rw | N | 0 | Node-ID (value range: 1...127) |
| | ... | ... | ... | ... | ... | ... | ... |
| | 0x10 | 16. Node-ID for PLC-STOP | Unsigned8 | rw | N | 0 | Node-ID (value range: 1...127) |

Beim Übergang von PLC-STOP → PLC-RUN werden die hier eingetragenen CAN-Geräte über das NMT-Kommando *Operational* in den Zustand Operational gesetzt.

Start address RxPDO-Counter

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|-----------------------------|-----------|-------|------|---------------|-----------------------------|
| 0x2003 | 0x00 | Start address RxPDO-Counter | Unsigned8 | rw | N | 0 | Start address RxPDO-Counter |

Über diesen Index kann in der SPS eine Startadresse im Prozesseingangsabbild für die RxPDO-Zähler definiert werden.

1. ➤ Für jedes RxPDO wird ein Zähler zur Verfügung gestellt.
2. ➤ Mit Empfang eines PDOs wird der entsprechende Zähler + 1 gezählt.
3. ➤ Beim Übergang von 255 → 0 springt der Zähler automatisch auf 1.
4. ➤ Im Defaultzustand und beim Übergang der SPS in den Zustand STOP wird der Zähler auf 0 gesetzt.

| PAE-Adresse | Bedeutung |
|-------------|-------------------------|
| X | Zähler für RxPDO 1 |
| X+1 | Zähler für RxPDO 2 |
| X+2 | Zähler für RxPDO 3 |
| X+3 | Zähler für RxPDO 4 |
| X+4 | Zähler für RxPDO 5 |
| X+5 | Zähler für RxPDO 6 |
| | |
| X+35 | Zähler für RxPDO 36 |
| X+36 | Zähler für RxPDO 37 |
| X+37 | Zähler für RxPDO 38 |
| X+38 | Zähler für RxPDO 39 |
| X+39 | Zähler für RxPDO 40 |
| X+40 | Zähler für SYNC-Message |

Start address NG/HB-ToggleBit

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|-----------------------------|-----------|-------|------|---------------|-----------------------------|
| 0x2003 | 0x00 | Start address RxPDO-Counter | Unsigned8 | rw | N | 0 | Start address RxPDO-Counter |

- Über diesen Index kann in der SPS eine Startadresse im Prozesseingangsabbild (PAE) für NG/HB definiert werden.
- Für jede NodeGuarding/Heartbeat COB-ID ist ein Bit reserviert.
- Mit jeder empfangenen NG/HB-COB-ID wird ein Bit getoggelt.
- Im Defaultzustand und beim Übergang der SPS in den Zustand STOP wird das Togglebit auf 0 gesetzt.

| PAE-Adresse | Bedeutung |
|-------------|--------------------------------------|
| X | Toggle Bit für COB-ID 0x701 .. 0x708 |
| X+1 | Toggle Bit für COB-ID 0x709 .. 0x710 |
| X+2 | Toggle Bit für COB-ID 0x711 .. 0x718 |
| X+3 | Toggle Bit für COB-ID 0x719 .. 0x720 |
| X+4 | Toggle Bit für COB-ID 0x721 .. 0x728 |
| X+5 | Toggle Bit für COB-ID 0x729 .. 0x730 |
| X+6 | Toggle Bit für COB-ID 0x731 .. 0x738 |
| X+7 | Toggle Bit für COB-ID 0x739 .. 0x740 |
| X+8 | Toggle Bit für COB-ID 0x741 .. 0x748 |
| X+9 | Toggle Bit für COB-ID 0x749 .. 0x750 |
| X+10 | Toggle Bit für COB-ID 0x751 .. 0x758 |
| X+11 | Toggle Bit für COB-ID 0x759 .. 0x760 |
| X+12 | Toggle Bit für COB-ID 0x761 .. 0x768 |
| X+13 | Toggle Bit für COB-ID 0x769 .. 0x770 |
| X+14 | Toggle Bit für COB-ID 0x771 .. 0x778 |
| X+15 | Toggle Bit für COB-ID 0x779 .. 0x77F |

Start address L2-Message-Area

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|-------------------------------|-----------|-------|------|---------------|-----------------|
| 0x2005 | 0x00 | Start address L2-Message-Area | Unsigned8 | rw | N | 0 | L2-Message-Area |

- Über diesen Index kann in der SPS eine Startadresse im Prozesseingangsabbild für die L2-Message-Area definiert werden.
- Über die Message-Area kann der Endanwender über das SPS-Programm CAN-Telegramme versenden.
- Es stehen dabei 5 verschiedene Messagepuffer zur Verfügung.

- Über das Status-Byte wird das Senden der CAN-Telegramme gesteuert.
- Im Defaultzustand und beim Übergang der SPS in den Zustand STOP wird die komplette Datenstruktur mit 0 initialisiert.

| PAA Adr. | Kanal | Typ | Bedeutung | | PAE Adr. | Kanal | Typ | Bedeutung |
|----------|-------|-----|-------------|--|----------|-------|-----|-------------|
| X | 0 | B | Status-Byte | | X | 0 | B | Status byte |
| X+1 | | B | Datenlänge | | X+1 | 1 | B | Status byte |
| X+2 | | DW | COB-ID | | X+2 | 2 | B | Status byte |
| X+3 | | | | | X+3 | 3 | B | Status byte |
| X+4 | | | | | X+5 | 4 | B | Status byte |
| X+5 | | | | | | | | |
| X+6 | | B | Datenbyte 0 | | | | | |
| X+7 | | B | Datenbyte 1 | | | | | |
| X+8 | | B | Datenbyte 2 | | | | | |
| X+9 | | B | Datenbyte 3 | | | | | |
| X+10 | | B | Datenbyte 4 | | | | | |
| X+11 | | B | Datenbyte 5 | | | | | |
| X+12 | | B | Datenbyte 6 | | | | | |
| X+13 | | B | Datenbyte 7 | | | | | |
| X+14 | 1 | B | Status-Byte | | | | | |
| X+15 | | B | Datenlänge | | | | | |
| X+16 | | DW | COB-ID | | | | | |
| X+17 | | | | | | | | |
| X+18 | | | | | | | | |
| X+19 | | | | | | | | |
| X+20 | | B | Datenbyte 0 | | | | | |
| X+21 | | B | Datenbyte 1 | | | | | |
| X+22 | | B | Datenbyte 2 | | | | | |
| X+23 | | B | Datenbyte 3 | | | | | |
| X+24 | | B | Datenbyte 4 | | | | | |
| X+25 | | B | Datenbyte 5 | | | | | |
| X+26 | | B | Datenbyte 6 | | | | | |
| X+27 | | B | Datenbyte 7 | | | | | |
| X+28 | 2 | B | Status-Byte | | | | | |
| X+29 | | B | Datenlänge | | | | | |
| X+30 | | DW | COB-ID | | | | | |
| X+31 | | | | | | | | |
| X+32 | | | | | | | | |
| X+33 | | | | | | | | |
| X+34 | | B | Datenbyte 0 | | | | | |
| X+35 | | B | Datenbyte 1 | | | | | |
| X+36 | | B | Datenbyte 2 | | | | | |

Objekt-Verzeichnis

| PAA Adr. | Kanal | Typ | Bedeutung | | PAE Adr. | Kanal | Typ | Bedeutung |
|----------|-------|-------------|-------------|-------------|-------------|-------|-----|-----------|
| X+37 | 3 | B | Datenbyte 3 | | | | | |
| X+38 | | B | Datenbyte 4 | | | | | |
| X+39 | | B | Datenbyte 5 | | | | | |
| X+40 | | B | Datenbyte 6 | | | | | |
| X+41 | | B | Datenbyte 7 | | | | | |
| X+42 | | B | Status-Byte | | | | | |
| X+43 | | B | Datenlänge | | | | | |
| X+44 | | DW | COB-ID | | | | | |
| X+45 | | | | | | | | |
| X+46 | | | | | | | | |
| X+47 | | | | | | | | |
| X+48 | | | | B | Datenbyte 0 | | | |
| X+49 | | B | Datenbyte 1 | | | | | |
| X+50 | | B | Datenbyte 2 | | | | | |
| X+51 | | B | Datenbyte 3 | | | | | |
| X+52 | | B | Datenbyte 4 | | | | | |
| X+53 | | B | Datenbyte 5 | | | | | |
| X+54 | | B | Datenbyte 6 | | | | | |
| X+55 | | B | Datenbyte 7 | | | | | |
| X+56 | | 4 | B | Status-Byte | | | | |
| X+57 | B | | Datenlänge | | | | | |
| X+58 | DW | | COB-ID | | | | | |
| X+59 | | | | | | | | |
| X+60 | | | | | | | | |
| X+61 | | | | | | | | |
| X+62 | | | | B | Datenbyte 0 | | | |
| X+63 | B | | Datenbyte 1 | | | | | |
| X+64 | B | | Datenbyte 2 | | | | | |
| X+65 | B | | Datenbyte 3 | | | | | |
| X+66 | B | | Datenbyte 4 | | | | | |
| X+67 | B | | Datenbyte 5 | | | | | |
| X+68 | B | | Datenbyte 6 | | | | | |
| X+69 | BV | Datenbyte 7 | | | | | | |

| CANmaster | SPS |
|--|---|
| Initialisierung/ SPS-Stop | |
| <ul style="list-style-type: none"> ■ L2-Message-Area: <ul style="list-style-type: none"> – Datenstruktur wird mit 0 initialisiert | <ul style="list-style-type: none"> ■ OB 100: <ul style="list-style-type: none"> – PAA-Bereich der L2-Message-Area mit 0 initialisieren |
| Telegramm senden | |
| PAE-Status ungleich PAA-Status? → Telegramm in Sendequelle eintragen → setze PAE-Status = PAA-Status | PAE-Status gleich PAA-Status? → COB-ID + Daten schreiben → PAA-Status incrementieren |

Lenze NodeGuarding

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|--------------------|------------|-------|------|---------------|--------------------------------|
| 0x2016 | 0x00 | Number of elements | Unsigned8 | ro | N | 127 | Number of available entries. |
| | 0x01 | 1. entry | Unsigned32 | rw | N | 0 | NodeGuarding für Node-ID = 1 |
| | 0x02 | 2. entry | Unsigned32 | rw | N | 0 | NodeGuarding für Node-ID = 2 |
| | 0x03 | 3. entry | Unsigned32 | rw | N | | NodeGuarding für Node-ID = 3 |
| | ... | ... | ... | ... | ... | ... | ... |
| | 0x7F | 127. entry | Unsigned32 | rw | N | 0 | NodeGuarding für Node-ID = 127 |

Dieser Index ist speziell für Lenze Frequenzumrichter eingerichtet worden. LENZE unterstützt kein Nodeguarding/Heartbeat nach CANopen Spezifikation DS301. Über diesen Index besteht nun die Möglichkeit einen SDO-Transfer einzurichten. Der CAN-Master versendet im zeitlichen Abstand (TimeOut Value * 100ms) ein SDO.request an den Lenze Frequenzumrichter. Ist nach einem Timeout von 1sec keine SDO.response vom Lenze Frequenzumrichter eingetroffen so meldet der CAN-Master Slaveausfall an die SPS (OB86-Aufruf).

Struktur des Lenze-Nodeguarding-Eintrags

| | | | |
|------------|------------|-----------|-----------------------|
| Bits | 31-16 | 15-8 | 7-0 |
| Value | Index | SubIndex | TimeOut-value * 100ms |
| Encoded as | Unsigned16 | Unsigned8 | Unsigned8 |

- Beispiel für Lenze:
 - 0x5E980005 // Index 0x5E98 entspricht der Lenze-Codestelle C0359, SubIndex 0, Timeout 5 == 500ms.

Message PLC-RUN

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|--------------------|------------|-------|------|---------------|------------------------------|
| 0x2100 | 0x00 | Number of elements | Unsigned8 | ro | N | 10 | Number of available entries. |
| | 0x01 | COB-ID | Unsigned32 | rw | N | 0 | COB-ID |
| | 0x02 | Data length | Unsigned8 | rw | N | 0 | Data length |
| | 0x03 | Data 1 | Unsigned8 | rw | N | 0 | Data 1 |
| | 0x04 | Data 2 | Unsigned8 | rw | N | 0 | Data 1 |
| | 0x05 | Data 3 | Unsigned8 | rw | N | 0 | Data 1 |

Über diesen Index kann ein CAN-Telegramm definiert werden, welches beim Übergang von SPS-STOP → SPS-RUN ausgeführt werden soll.

Message PLC-STOP

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|--------------------|------------|-------|------|---------------|-----------------------------|
| 0x2101 | 0x00 | Number of elements | Unsigned8 | ro | N | 10 | Number of available entries |
| | 0x01 | COB-ID | Unsigned32 | rw | N | 0 | COB-ID |
| | 0x02 | Data length | Unsigned8 | rw | N | 0 | Data length |
| | 0x03 | Data 1 | Unsigned8 | rw | N | 0 | Data 1 |
| | 0x04 | Data 2 | Unsigned8 | rw | N | 0 | Data 1 |
| | 0x05 | Data 3 | Unsigned8 | rw | N | 0 | Data 1 |
| | 0x06 | Data 4 | Unsigned8 | rw | N | 0 | Data 1 |
| | 0x07 | Data 5 | Unsigned8 | rw | N | 0 | Data 1 |
| | 0x08 | Data 6 | Unsigned8 | rw | N | 0 | Data 1 |
| | 0x09 | Data 7 | Unsigned8 | rw | N | 0 | Data 1 |
| | 0x0A | Data 8 | Unsigned8 | rw | N | 0 | Data 1 |

Über diesen Index kann ein CAN-Telegramm definiert werden, welches beim Übergang von SPS-RUN → SPS-STOP ausgeführt werden soll.

J1939: PGN for Multipaket Transfer

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|--------------------|-----------|-------|------|---------------|------------------------------|
| 0x2200 | 0x00 | Number of elements | Unsigned8 | ro | N | 16 | Number of available entries. |
| | 0x01 | 1. PGN | Unsigned8 | rw | N | 0 | PGN |
| | ... | ... | ... | ... | ... | ... | ... |
| | 0x10 | 16. PGN | Unsigned8 | rw | N | 0 | PGN |

Es handelt sich hierbei um einen Index für das J1939 Protokoll. Im J1939 Protokoll werden größere Datenmengen über das Multipaket-Protokoll übertragen. Es werden dabei folgende COB-IDs verwendet: 20ECFF00h, 20EBFF00h Über die COB-ID 20ECFF00h wird die PGN-Nummer und die Länge der Daten übertragen. Über die COB-

ID 20EBFF00h werden die Daten segmentiert übertragen. Um die Daten korrekt bearbeiten zu können ist es zwingend erforderlich die Funktionalität Herstellerspezifischer-Alarm (Manufacturer Alarm OB 57) im Konfigurationstool WinCoCT/PLC-Parameter zu aktivieren und die Anzahl der Telegramme (Number of messages) auf 1 einzustellen. Ferner sind im Index 2000h die COB-IDs 20ECFF00h, 20EBFF00h einzutragen. Mit dem Index 0x2200 kann nun die Anzahl der OB57-Aufrufe in der SPS eingeschränkt werden. Nur die Datentelegramme der hier eingestellten PGN-Nummern erzeugen einen OB57-Aufruf in der SPS.

Special Settings for CAN

| Index | Sub-index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|--------------------------|-----------|-------|------|---------------|--------------------------|
| 0x3000 | 0x00 | Special Settings for CAN | Unsigned8 | rw | N | 0 | Special Settings for CAN |

Über diesen Index werden Spezialfunktionen der CAN-Firmware eingestellt.

- **Bit 0:** Die RxPDO-Längenüberprüfung kann abgeschaltet werden
 - Bit = 0: Die Längenüberprüfung ist aktiviert
 - Bit = 1: Die Längenüberprüfung ist deaktiviert
- **Bit 6...1:** reserviert
- **Bit 7:** spezielles Bit für J1939
 - Bit = 0: Die Priorität der J1939 COB-IDs wird überprüft
 - Bit = 1: Die Priorität der J1939 COB-IDs wird nicht überprüft

8bit Digital inputs

| Index | Sub-Index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|--------------------------|-----------|-------|------|---------------|---|
| 0x6000 | 0x00 | 8bit digital input block | Unsigned8 | ro | N | 0x01 | Number of available digital 8bit input blocks |
| | 0x01 | 1. input block | Unsigned8 | ro | Y | | 1. digital input block |
| | ... | ... | ... | ... | ... | ... | ... |
| | 0x40 | 64. input block | Unsigned8 | ro | Y | | 64. digital input block |

16bit Digital inputs

| Index | Sub-Index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|---------------------------|------------|-------|------|------------------------------------|--|
| 0x6100 | 0x00 | 16bit digital input block | Unsigned8 | ro | N | depending on the fitted components | Number of available digital 16bit input blocks |
| | 0x01 | 1. input block | Unsigned16 | ro | N | | 1. digital input block |
| | ... | ... | ... | ... | ... | ... | ... |
| | 0x40 | 160. input block | Unsigned16 | ro | N | | 32. digital input block |

32bit Digital inputs

| Index | Sub-Index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|---------------------------|------------|-------|------|------------------------------------|--|
| 0x6120 | 0x00 | 32bit digital input block | Unsigned8 | ro | N | depending on the fitted components | Number of available digital 32bit input blocks |
| | 0x01 | 1. input block | Unsigned32 | ro | N | | 1. digital input block |
| | ... | ... | ... | ... | ... | ... | ... |
| | 0x50 | 80. input block | Unsigned32 | ro | N | | 16. digital input block |

8bit Digital outputs

| Index | Sub-Index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|---------------------------|-----------|-------|------|---------------|--|
| 0x6200 | 0x00 | 8bit digital output block | Unsigned8 | ro | N | 0x01 | Number of available digital 8bit output blocks |
| | 0x01 | 1. output block | Unsigned8 | rw | Y | | 1. digital output block |
| | ... | ... | ... | ... | ... | ... | ... |
| | 0x40 | 64. output block | Unsigned8 | rw | Y | | 64. digital output block |

16bit Digital outputs

| Index | Sub-Index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|---------------------------|------------|-------|------|------------------------------------|---|
| 0x6300 | 0x00 | 16bit digital input block | Unsigned8 | ro | N | Depending on the components fitted | Number of available digital 16bit output blocks |
| | 0x01 | 1. output block | Unsigned16 | rw | N | | 1. digital output block |
| | ... | ... | ... | ... | ... | ... | ... |
| | 0x0A | 160. output block | Unsigned16 | rw | N | | 32. digital output block |

32bit Digital outputs

| Index | Sub-Index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|---------------------------|------------|-------|------|------------------------------------|---|
| 0x6320 | 0x00 | 32bit digital input block | Unsigned8 | ro | N | Depending on the components fitted | Number of available digital 32bit output blocks |
| | 0x01 | 1. output block | Unsigned32 | rw | N | | 1. digital output block |
| | ... | ... | ... | ... | ... | ... | ... |
| | 0x50 | 80. output block | Unsigned32 | rw | N | | 16. digital output block |

8bit Network input variables

| Index | Sub-Index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|--------------------------|-----------|-------|------|---------------|---|
| 0xA040 | 0x00 | 8bit digital input block | Unsigned8 | ro | N | 0x01 | Number of available digital 8bit input blocks |
| | 0x01 | 1. input block | Unsigned8 | ro | Y | | 1. digital input block |
| | ... | ... | ... | ... | ... | ... | ... |
| | 0xFE | 254. input block | Unsigned8 | ro | Y | | 320. digital input block |

16bit Network input variables

| Index | Sub-Index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|---------------------------|------------|-------|------|------------------------------------|--|
| 0xA100 | 0x00 | 16bit digital input block | Unsigned8 | ro | N | depending on the fitted components | Number of available digital 16bit input blocks |
| | 0x01 | 1. input block | Unsigned16 | ro | N | | 1. digital input block |
| | ... | ... | ... | ... | ... | ... | ... |
| | 0xA0 | 160. input block | Unsigned16 | ro | N | | 160. digital input block |

32bit Network input variables

| Index | Sub-Index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|---------------------------|------------|-------|------|------------------------------------|--|
| 0xA200 | 0x00 | 32bit digital input block | Unsigned8 | ro | N | Depending on the components fitted | Number of available digital 32bit input blocks |
| | 0x01 | 1. input block | Unsigned32 | ro | N | | 1. digital input block |
| | ... | ... | ... | ... | ... | ... | ... |
| | 0x50 | 80. input block | Unsigned32 | ro | N | | 80. digital input block |

64bit Network input variables

| Index | Sub-Index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|---------------------------|------------|-------|------|------------------------------------|--|
| 0xA440 | 0x00 | 64bit digital input block | Unsigned8 | ro | N | depending on the fitted components | Number of available digital 64bit input blocks |
| | 0x01 | 1. input block | Unsigned32 | ro | N | | 1. digital input block |
| | ... | ... | ... | ... | ... | ... | ... |
| | 0x28 | 40. input block | Unsigned32 | ro | N | | 40. digital input block |

8bit Network output variables

| Index | Sub-Index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|--------------------------|-----------|-------|------|---------------|--|
| 0xA400 | 0x00 | 8bit digital input block | Unsigned8 | ro | N | 0x01 | Number of available digital 8bit output blocks |
| | 0x01 | 1. output block | Unsigned8 | rw | Y | | 1. digital output block |
| | ... | ... | ... | ... | ... | ... | ... |
| | 0xFE | 254. output block | Unsigned8 | rw | Y | | 320. digital output block |

16bit Network output variables

| Index | Sub-Index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|---------------------------|------------|-------|------|------------------------------------|---|
| 0xA580 | 0x00 | 16bit digital input block | Unsigned8 | ro | N | Depending on the components fitted | Number of available digital 16bit output blocks |
| | 0x01 | 1. output block | Unsigned16 | rw | N | | 1. digital output block |
| | ... | ... | ... | ... | ... | ... | ... |
| | 0xA0 | 160. output block | Unsigned16 | rw | N | | 160. digital output block |

32bit Network output variables

| Index | Sub-Index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|---------------------------|------------|-------|------|------------------------------------|---|
| 0xA680 | 0x00 | 32bit digital input block | Unsigned8 | ro | N | Depending on the components fitted | Number of available digital 32bit output blocks |
| | 0x01 | 1. output block | Unsigned32 | rw | N | | 1. digital output block |
| | ... | ... | ... | ... | ... | ... | ... |
| | 0x50 | 80. output block | Unsigned32 | rw | N | | 80. digital output block |

64bit Network output variables

| Index | Sub-Index | Name | Type | Attr. | Map. | Default value | Meaning |
|--------|-----------|---------------------------|------------|-------|------|------------------------------------|---|
| 0xA8C0 | 0x00 | 64bit digital input block | Unsigned8 | ro | N | Depending on the components fitted | Number of available digital 64bit output blocks |
| | 0x01 | 1. output block | Unsigned32 | rw | N | | 1. digital output block |
| | ... | ... | ... | ... | ... | ... | ... |
| | 0x50 | 40. output block | Unsigned32 | rw | N | | 40. digital output block |

5.9 Diagnose

Übersicht

Sofern Sie "Diagnostic" in WinCoCT unter "PLC-Parameters" aktiviert haben, wird bei folgenden Ereignissen eine Diagnose an die CPU gemeldet und der entsprechenden OB aufgerufen:

- Wechsel des CANopen-Status (OB 82)
- Slave-Ausfall und -Wiederkehr (OB 86)
- Guarding-Fehler (OB 82)
- Emergency-Telegramm (OB 82)

Mit SFC 13 Diagnose auswerten

In dem entsprechenden OB können Sie mit dem SFC 13 DPNRM_DG auf die Diagnosedaten zugreifen.

- Durch den Eingangsparameter RECORD wird der Zielbereich festgelegt, in den die gelesenen Daten nach fehlerfreier Datenübertragung eingetragen werden.
- Der Lesevorgang wird gestartet, wenn am Eingangsparameter REQ der Wert 1 anliegt.



Näheres zum Einsatz dieses Bausteins finden Sie im Handbuch "SPEED7 Operationsliste" von Yaskawa.

5.9.1 Aufbau der Diagnosedaten

In der Regel haben die Diagnosedaten eine Länge von 35Byte. Ist im Stationsstatus 1 das Bit 3 "DiagExtDiag" = 0, werden nur die CAN-Diagnose-Daten mit einer Länge von 6Byte übertragen. Die nachfolgende Tabelle enthält Informationen zum prinzipiellen Aufbau der Diagnosedaten:

| | Byte | Bedeutung |
|-------------------|----------|------------------|
| CAN-Diagnose | 0 | Stationsstatus 1 |
| | 1 | Stationsstatus 2 |
| | 2 | Stationsstatus 3 |
| | 3 | Node-ID |
| | 4 | fix 0 |
| | 5 | Device type |
| Extended Diagnose | 6 ... 34 | Status message |

Stationsstatus 1

| Bit | Name | Bedeutung |
|-------|------------------------|---|
| 0 | DiagStationNonExistent | 1 = Station existiert nicht 0 = Station existiert (das Bit wird auf 0 gesetzt, wenn eine Bootup-Message empfangen worden ist oder das Node Guarding aktiviert wurde.) |
| 1 | DiagStationNotReady | 1 = Station ist im Zustand pre-operational 0 = Station ist im Zustand operational |
| 2 | - | reserviert |
| 3 | DiagExtDiag | 0 = Station hat nur die CAN-Diagnose 1 = Station hat erweiterte Diagnosedaten |
| 7...4 | - | reserviert |

Stationsstatus 2

| Bit | Name | Bedeutung |
|-------|------------|--|
| 0 | DiagPrmReq | 0 = Station ist erfolgreich parametrier 1 = Station muss neu parametrier werden |
| 1 | - | reserviert |
| 2 | - | fix 1 |
| 3 | DiagWD_ON | 0 = Node Guarding wird nicht unterstützt 1 = Node Guarding ist aktiviert |
| 7...4 | - | reserviert |

Stationsstatus 3

Das Byte ist reserviert für zukünftige Erweiterungen.

Node-ID

ID der Station, von der die Diagnose kommt.

Device type

Typ der Station, von der die Diagnose kommt.

0 = Slave

1 = NMT-Master

Status message

| Byte | Name | Bedeutung |
|------|--------|-----------|
| 0 | Header | fix 29 |
| 1 | Type | fix 81hex |
| 2 | SlotNr | fix 0 |

| Byte | Name | Bedeutung |
|---------|---------------|--|
| 3 | Specifier | Eigenschaft der Status message 0 = keine weitere Unterscheidung 1 = Status message kommend 2 = Status message gehend |
| 4...7 | VendorID | CANopen Index 1018 SubIndex 1 |
| 8...11 | ProductCode | CANopen Index 1018 SubIndex 2 |
| 12...15 | RevisionNr | CANopen Index 1018 SubIndex 3 |
| 16...19 | SerialNr | CANopen Index 1018 SubIndex 4 |
| 20 | DiagError | Diagnose-Fehlernummer (10h ... 31h) 10h = DIAG_SLAVEBOOTUP 11h = DIAG_SLAVEGRDERROR 12h = DIAG_SLAVESDOERROR 13h = DIAG_SLAVEEMCYIND |
| 21...28 | DiagErrorData | Zusatzdaten zum Diagnosefehler |

Übersicht DiagError DiagErrorData

Die Länge der Zusatzdaten beträgt immer 8Byte.

DIAG_SLAVE BOOTUP (10h)

Diese Meldung wird generiert sobald der Master die Bootup-Meldung von der entsprechenden Slave-Station empfangen hat.

Zusatzdaten: 8Byte fix 0

DIAG_SLAVE GRDERROR (11h)

Diese Meldung wird vom Master generiert, wenn die entsprechende Slave-Station auf Node Guarding Telegramme nicht mehr antwortet oder keinen Heartbeat generiert.

Zusatzdaten:

| Byte | Code |
|-------|------------------|
| 0 | Event code |
| 1 | Active Status |
| 2 | Respected Status |
| 3...7 | fix 0 |

| Event code | Bedeutung |
|------------|--|
| 0 | Das Guarding ist nicht aktiviert. |
| 1 | Das Guarding wurde (wieder) aktiviert. Diese Meldung erfolgt auch, wenn das Guarding für eine Slave-Station vom Fehlerzustand in einen fehlerfreien Zustand überführt wurde. |
| 2 | Die Guard-Antwort einer Slave-Station ist innerhalb einer Guardzeit nicht empfangen worden. |
| 3 | Die Guard-Antwort einer Slave-Station ist innerhalb der Zeit $Guardtime\ t * LifeTimeFactor\ n$ nicht empfangen worden. Vor diesem Ereignis wurde bereits n mal der Code 2 gemeldet. Das Guarding für die Slave-Station ist damit ausgefallen. |

| Event code | Bedeutung |
|------------|--|
| 4 | Das Toggle-Bit der Slave-Nachricht stimmt nicht mit dem erwarteten Wert überein. Der Master passt seinen Toggle-Wert an, so dass dieser Fehler nur einmalig ausgelöst wird. |
| 5 | Die Slave-Station hat einen Kommunikationsstatus gemeldet, den der Master nicht vorgegeben hat. Dieser Fehler tritt bei einem lokalen Statusübergang in der Slave-Station auf. Der Fehler wird permanent gemeldet, bis die Inkonsistenz behoben ist. |
| 6 | Ein Heartbeat-Event ist aufgetreten. Die Heartbeat-Zeit für eine in der Heartbeat-Tabelle eingetragenen Slave-Station ist abgelaufen, ohne dass ein Heartbeat empfangen werden konnte. |

| Active/respected status code | Bedeutung |
|------------------------------|---------------------|
| 4 | Prepared |
| 5 | Operational |
| 6 | Reset |
| 7 | Reset Communication |
| 127 | Pre-operational |

**DIAG_SLAVE SDOERROR
(12h)****Zusatzdaten:**

| Byte | Code |
|-------|---------------------|
| 0 | High-Byte SDO-Index |
| 1 | Low-Byte SDO-Index |
| 2 | SDO-Subindex |
| 3...6 | CANOPENERROR |
| 7 | fix 0 |

| Code | Bedeutung |
|------------|---|
| 0x05030000 | Toggle-Bit nicht geändert |
| 0x05040000 | SDO Protokoll Time-out |
| 0x05040001 | Client/Server Befehlsspezifizierung nicht gültig oder unbekannt |
| 0x05040002 | Ungültige Blockgröße (nur Block-Modus) |
| 0x05040003 | Ungültige Sequenznummer (nur Block-Modus) |
| 0x05040004 | CRC Fehler (nur Block-Modus) |
| 0x05040005 | Unzureichender Speicher |
| 0x06010000 | Nicht unterstützter Zugriff auf ein Objekt |
| 0x06010001 | Lesezugriff auf ein Nur-Schreiben-Objekt |
| 0x06010002 | Schreibzugriff auf ein Nur-Lesen-Objekt |
| 0x06020000 | Objekt nicht im Objektverzeichnis vorhanden |
| 0x06040041 | Objekt kann nicht ins PDO gemappt werden |
| 0x06040042 | Anzahl und Länge der zu mappenden Objekte überschreitet PDO-Länge |
| 0x06040043 | Generelle Parameterinkompatibilität |
| 0x06040047 | Generelle interne Inkompatibilität im Gerät |
| 0x06060000 | Zugriffsfehler wegen Hardwareausfall |
| 0x06070010 | Datentyp nicht korrekt, Länge der Serviceparameter nicht korrekt |
| 0x06070012 | Datentyp nicht korrekt, Serviceparameter zu lang |
| 0x06070013 | Datentyp nicht korrekt, Serviceparameter zu kurz |
| 0x06090011 | Subindex existiert nicht |
| 0x06090030 | Wertebereich der Parameter überschritten (nur für Schreibzugriff) |
| 0x06090031 | Zu schreibender Parameterwert ist zu hoch |
| 0x06090032 | Zu schreibender Parameterwert ist zu niedrig |
| 0x06090036 | Maximumwert ist kleiner als Minimumwert |
| 0x08000000 | Genereller Fehler |
| 0x08000020 | Die Daten können entweder nicht transferiert oder nicht in der SPS gespeichert werden. |
| 0x08000021 | Die Daten können wegen lokaler Kontrollen entweder nicht transferiert oder nicht in der SPS gespeichert werden. |
| 0x08000022 | Die Daten können wegen aktuellem Modulstatus entweder nicht transferiert oder nicht in der SPS gespeichert werden. |
| 0x08000023 | Dynamische Objektverzeichnisgenerierung fehlgeschlagen oder kein Objektverzeichnis gefunden (z.B. Objektverzeichnis wird aus Datei generiert und ein Dateifehler ist aufgetreten). |
| 0x08000024 | Die angefragte Aktion wird von der Applikation nicht unterstützt. |

**DIAG_SLAVE EMCYIND
(13h)**

Zusatzdaten: Emergency-Telegramm

Um anderen Teilnehmern am CANopen-Bus interne Gerätefehler oder CAN-Busfehler mitteilen zu können, verfügen die CANopen CPs über das Emergency-Object. Es ist mit einer hohen Priorität versehen und liefert wertvolle Informationen über den Zustand des Gerätes und des Netzes. Das Emergency-Telegramm ist immer 8Byte lang. Es enthält zunächst den 2Byte Error Code, dann das 1Byte Error Register und schließlich den 5Byte großen Additional Code.

Telegramm-Aufbau

| Error Code | | ErrorRegister Index 0x1001 | Info 0 | Info 1 | Info 2 | Info 3 | Info 4 |
|------------|----------|----------------------------|--------|--------|--------|--------|--------|
| LowByte | HighByte | | | | | | |

| Error code | Meaning | Info 0 | Info 1 | Info 2 | Info 3 | Info4 |
|------------|---------------------------------------|-------------------------------|--------------------------------|-------------------------|------------------------|-------------------------|
| 0x0000 | Reset Emergency | | | | | |
| 0x1000 | PDO Control | 0xFF | 0x10 | PDO Number | LowByte Timer Value | HighByte Timer Value |
| 0x6200 | PLC-STOP | 1=PLC-STOP | 0x00 | 0x00 | 0x00 | 0x00 |
| 0x6363 | PDO-Mapping | LowByte: Mapping parameter | HighByte: Mapping parameter | Mapping entries | 0x00 | 0x00 |
| 0x8100 | Heartbeat Consumer | Node ID | LowByte Timer Value | HighByte Timer Value | 0x00 | 0x00 |
| 0x8100 | SDO Block Transfer | 0xF1 | LowByte Index | HighByte Index | SubIndex | 0x00 |
| 0x8130 | Node Guarding Error | LowByte GuardTime | HighByte GuardTime | LifeTime | 0x00 | 0x00 |
| 0x8210 | PDO not processed due to length error | PDO Number | Wrong length | PDO length | 0x00 | 0x00 |
| 0x8220 | PDO length exceeded | PDO Number | Wrong length | PDO length | 0x00 | 0x00 |

5.10 SZL auslesen

Übersicht

Die Systemzustandsliste (SZL) beschreibt den aktuellen Zustand Ihres Automatisierungssystems. Auf die SZL haben Sie ausschließlich lesenden Zugriff in Form von Teillisten(-auszügen). Diese Listen werden von der CPU auf Anforderung zusammengestellt. Zur Identifikation einer Teilliste gibt es die SZL-ID.

5.10.1 SFC 51 - RDSYSST - Auslesen der Informationen der SZL

Beschreibung

Mit dem SFC 51 RDSYSST (read system status) können Sie eine Teilliste bzw. einen Teillistenauszug der SZL (**S**ystemzustandsliste) anfordern. Hierbei bestimmen Sie mit den Parametern *SZL_ID* und *INDEX* was Sie auslesen möchten.

Der *INDEX* ist nicht immer erforderlich. Er dient der Bestimmung eines Objekts innerhalb einer Teilliste.

Durch Setzen von *REQ* starten Sie die Abfrage. Sobald *BUSY* = 0 zurückgemeldet wird, liegen die Daten im Zielbereich DR ab.

Informationen zur SZL finden Sie im Teil "Systemzustandsliste SZL".

Parameter

| Parameter | Deklaration | Datentyp | Speicherbereich | Beschreibung |
|------------|-------------|----------|--------------------------|--|
| REQ | INPUT | BOOL | E, A, M, D, L, Konstante | REQ = 1: Anstoß der Bearbeitung |
| SZL_ID | INPUT | WORD | E, A, M, D, L, Konstante | SZL_ID der Teilliste oder des Teillistenauszugs |
| INDEX | INPUT | WORD | E, A, M, D, L, Konstante | Typ oder Nummer eines Objekts in einer Teilliste |
| RET_VAL | OUTPUT | INT | E, A, M, D, L | Tritt während der Bearbeitung der Funktion ein Fehler auf, enthält der Rückgabewert einen Fehlercode |
| BUSY | OUTPUT | BOOL | E, A, M, D, L | BUSY = 1: Lesevorgang noch nicht abgeschlossen |
| SZL_HEADER | OUTPUT | STRUCT | D, L | Struktur mit 2 WORD-Typen: LENGTHDR: Länge Datensatz WORDN_DR: Anzahl vorhandener zugehöriger Datensätze (bei Zugriff auf Teillistenkopfinfo) oder Anzahl der in DR übertragenen Datensätze. |
| DR | OUTPUT | ANY | E, A, M, D, L | Zielbereich für die gelesene SZL-Teilliste bzw. den gelesenen SZL-Teillistenauszug: Falls Sie nur die SZL-Teillistenkopfinfo einer SZL-Teilliste ausgelesen haben, dürfen Sie DR nicht auswerten, sondern nur <i>SZL_HEADER</i> . Andernfalls gibt das Produkt aus LENGTHDR und N_DR an, wie viele Bytes in DR eingetragen wurden. |

RET_VAL (Rückgabewert)

Tritt während der Bearbeitung der Funktion ein Fehler auf, enthält der Rückgabewert einen Fehlercode.

SZL auslesen > SZL-Listen des CAN-Masters

| Wert | Beschreibung |
|-------|--|
| 0000h | kein Fehler |
| 0081h | Länge des Ergebnisfeldes zu klein. Es werden trotzdem so viele Datensätze wie möglich geliefert. Der SZL-Header zeigt diese Anzahl an. |
| 7000h | Erstaufruf mit <i>REQ</i> = 0: keine Datenübertragung aktiv; <i>BUSY</i> = 0. |
| 7001h | Erstaufruf mit <i>REQ</i> = 1: Datenübertragung angestoßen; <i>BUSY</i> = 1. |
| 7002h | Zwischenaufruf (<i>REQ</i> irrelevant): Datenübertragung bereits aktiv; <i>BUSY</i> = 1. |
| 8081h | Länge des Ergebnisfeldes zu klein. Platz reicht nicht für einen Datensatz. |
| 8082h | <i>SZL_ID</i> ist falsch oder in der CPU bzw. in dem SFC unbekannt. |
| 8083h | <i>INDEX</i> ist falsch oder nicht erlaubt. |
| 8085h | Die Information ist systembedingt momentan nicht verfügbar, z.B. wegen Ressourcenmangels. |
| 8086h | Datensatz ist nicht lesbar wegen eines Systemfehlers. |
| 8087h | Datensatz ist nicht lesbar, weil das Modul nicht vorhanden ist oder nicht quittiert. |
| 8088h | Datensatz ist nicht lesbar, weil die tatsächliche Typkennung von der Solltypkennung abweicht. |
| 8089h | Datensatz ist nicht lesbar, weil das Modul nicht diagnosefähig ist. |
| 80A2h | DP-Protokollfehler - Layer-2-Fehler (temporärer Fehler). |
| 80A3h | DP-Protokollfehler bei User-Interface/User (temporärer Fehler) |
| 80A4h | Kommunikation am Bus ist gestört. Fehler tritt auf zwischen CPU und externer DP-Anschaltung (temporärer Fehler). |
| 80C5h | Dezentrale Peripherie nicht verfügbar (temporärer Fehler). |

5.10.2 SZL-Listen des CAN-Masters

Die hier aufgeführten SZL-Listen sind jeweils 8 Worte lang. Bei 0 beginnend belegt jede Node-ID in aufsteigender Reihenfolge ein Bit in der SZL. Das Bit für Node-ID 1 befindet sich in Bit 0 von Byte 0, das für Node-ID 12 in Bit 3 von Byte 1. Vom CAN-Master werden folgende SZL-IDs unterstützt:

| SZL-ID | Bedeutung |
|--------|--|
| 0x92 | Status projektierte Stationen des CAN-Mastersystems. <ul style="list-style-type: none"> ■ Bit=0 :Station ist nicht projektiert ■ Bit=1: Station ist projektiert |
| 0x192 | Status Aktivierte Stationen des CAN-Mastersystems. <ul style="list-style-type: none"> ■ Bit=0: Station ist nicht projektiert oder projektiert und aktiviert ■ Bit=1: Station ist projektiert und deaktiviert |

| SZL-ID | Bedeutung |
|--------|---|
| 0x292 | Ist-Zustand der Stationen des CAN-Mastersystems. <ul style="list-style-type: none">■ Bit=0: Station ist ausgefallen, deaktiviert oder nicht projiziert■ Bit=1: Station ist vorhanden, aktiviert und im Zustand operational |
| 0x692 | Diagnose-Zustand der Stationen des CAN-Mastersystems. <ul style="list-style-type: none">■ Bit=0: Station ist vorhanden, verfügbar, nicht gestört und aktiviert■ Bit=1: Station ist nicht OK oder deaktiviert |

5.11 Station (de-)aktivieren

Übersicht

Mit dem SFC 12 haben Sie die Möglichkeit angebundene CAN-Slave-Stationen zu aktivieren bzw. deaktivieren und deren Zustand zu ermitteln. Wenn Sie in einer CPU Stationen konfigurieren, die real nicht vorhanden sind oder aktuell nicht benötigt werden, greift die CPU dennoch regelmäßig auf diese Slaves zu. Nach deren Deaktivierung unterbleiben weitere CPU-Zugriffe. Dadurch kann der schnellstmögliche CAN-Buszyklus erreicht werden, und die entsprechenden Fehlerereignisse treten nicht mehr auf.



So lange ein oder mehrere SFC 12-Aufträge aktiv sind, können Sie keine geänderte Konfiguration vom PG in die CPU laden. Während des Ladens einer geänderten Konfiguration vom PG in die CPU weist die CPU die Aktivierung eines SFC 12-Auftrages ab.



Näheres zum Einsatz dieses Bausteins finden Sie im Handbuch "SPEED7 Operationsliste" von Yaskawa.