



## CAN 300

**CAN Kommunikations-Baugruppe für S7-300  
mit CANopen, CAN Layer 2 oder SAE J1939 – Handtierung**

700-600-CAN01

700-600-CAN81

## Handbuch

Ausgabe 18 / 13.01.2012 ab HW1 & FW2.6  
Handbuch Bestellnummer: 900-600-CAN01/de



Alle Rechte, auch die der Übersetzung, des Nachdruckes und der Vervielfältigung dieses Handbuches, oder Teilen daraus, vorbehalten. Kein Teil des Handbuches darf ohne schriftliche Genehmigung der Systeme Helmholtz GmbH in irgendeiner Form (Fotokopie, Mikrofilm oder ein anderes Verfahren), auch nicht für Zwecke der Unterrichtsgestaltung, oder unter Verwendung elektronischer Systeme reproduziert, verarbeitet, vervielfältigt oder verbreitet werden. Alle Rechte für den Fall der Patenterteilung oder Gebrauchsmustereintragung vorbehalten.

Copyright © 2012 by

**Systeme Helmholtz GmbH**

Hannberger Weg 2, 91091 Großenseebach

**Hinweis:**

Der Inhalt dieses Handbuches ist von uns auf die Übereinstimmung mit der beschriebenen Hard- und Software überprüft worden. Da dennoch Abweichungen nicht ausgeschlossen sind, können wir für die vollständige Übereinstimmung keine Gewährleistung übernehmen. Die Angaben in diesem Handbuch werden jedoch regelmäßig überprüft und notwendige Korrekturen sind in den nachfolgenden Ausgaben enthalten. Für Verbesserungsvorschläge sind wir Ihnen dankbar.

## Änderungen in diesem Dokument:

Stand	Datum	Änderung
13	23.6.2006	Bestellnummer DNV-Baugruppe 700-600-CAN81 ergänzt
		Firmware V. 2.5
		Hinweis auf „Highspeed ISO 11898-2“
		CAN-Bus Stecker axial ergänzt
		CANopen Hantierung V2.5: FC43 Korrekturen beim Parameter FUNC (Kap. 6.3.14)
14	27.8.2007	Firmware V. 2.6
		Ergänzung SF-LED bei CAN-Error
		Debug-Dialog: Status of Statemachine → STAT-Byte (Seite 25ff.)
		Layer 2 Hantierung V2.1: FC62 hinzugefügt (Kap. 6.2)
		Layer 2 Hantierung V2.1: FC63 CANSENDTIMER umbenannt (Kap. 6.2.4)
		Layer 2 Hantierung V2.1: FC62 CANCTRL hinzugefügt (Kap. 6.2.5)
		Layer 2 Hantierung V2.1: STAT-Byte geändert mit FW 2.6 (Kap. 6.2.6)
		CANopen Hantierung V2.6: STAT-Byte in FC49 hinzugefügt (Kap. 6.3.9)
		CANopen Hantierung V2.6: FC47 Anpassung des Beispiel DBs (Kap. 6.3.16)
		SAE J1939 Kapitel eingefügt (Kap. 6.6)
		Rechtschreibkorrekturen
14B	5.9.2007	Dokumentenänderungsliste hinzugefügt
15	17.12.2007	SAE J1939 Tabelle Bitnummern korrigiert
16	30.10.2008	RS232-Schnittstellenbelegung korrigiert
		Hinweis CPU-Stop bei Herunterladen
		Hinweis SF-LED bei CAN-FIFO Overflow
17	5.11.2008	CANopen Hantierung V27: Nodeguarding FC47 angepasst CANopen Hantierung V27: FC42 SDO Segmented angepasst CANopen Hantierung V28: FC46 Service entfernt Hinweis auf Firmwareupdatefunktion
18	11.01.2012	Einige kleinere Korrekturen

# Inhaltsverzeichnis

<b>1</b>	<b>Sicherheitshinweise</b>	<b>8</b>
1.1	Allgemein	8
1.2	Zugangsbeschränkung	9
1.3	Benutzerhinweise	9
1.4	Bestimmungsgemäßer Gebrauch	9
1.5	Bestimmungswidrigen Gebrauch vermeiden!	9
<b>2</b>	<b>Installation und Montage</b>	<b>10</b>
2.1	Vertikaler und horizontaler Aufbau	10
2.2	Mindestabstand	11
2.3	Montage der Baugruppe auf die Profilschiene	11
<b>3</b>	<b>Systemübersicht</b>	<b>13</b>
3.1	Anwendung und Funktionsbeschreibung	13
3.2	Anschlüsse	14
3.3	CAN-Verkabelung	14
3.4	LED-Anzeigen	15
3.5	Lieferumfang	15
3.6	Zubehör	15
<b>4</b>	<b>Projektierung in der SPS</b>	<b>16</b>
<b>5</b>	<b>Projektierung der CAN 300 Baugruppe</b>	<b>18</b>
5.1	Erstellen eines neuen Projektes	19
5.2	Einstellung der CAN-Bus Baudrate	20
5.3	Einstellung der Übertragungsart (Protokoll)	20
5.4	Akzeptanzmasken	21
5.5	Netzmanagement	22
5.6	Timer	23
5.7	Synchronfenster	24
5.8	Herunterladen	24
5.9	Diagnose/Debug	25

5.10	Firmware update	26
<b>6</b>	<b>Programmierung in der SPS</b>	<b>27</b>
6.1	Übersicht	27
6.2	Layer 2 Kommunikation	27
6.2.1	Allgemein	27
6.2.2	FC 60 CANSEND	28
6.2.3	FC 61 CANRCV	29
6.2.4	FC 63 CANSENDTIMER	30
6.2.5	FC 62 CANCTRL	31
6.2.6	Inhalt des Statusbytes STAT	32
6.3	CANopen Kommunikation	33
6.3.1	Allgemein	33
6.3.2	Objekte	33
6.3.3	Funktionen	34
6.3.4	Netzmanagement	35
6.3.5	Hantierungsbausteine	37
6.3.6	CAN-DB	37
6.3.7	FC 40 Initialisierung	38
6.3.8	PDO-DBs	39
6.3.9	FC 49 Zyklus	40
6.3.10	FC 41 SDO lesen und schreiben	41
6.3.11	FC 42 SDO Segment download/upload	43
6.3.12	FC 44 PDO senden	44
6.3.13	FC 45 PDO anfordern	45
6.3.14	FC 43 Spontanempfang	46
6.3.15	FC 48 Netzmanagement	47
6.3.16	FC 47 Nodeguarding/Heartbeat	48
6.4	Erläuterung zum CANopen Beispielprogramm	49
6.4.1	Beispiel FC 10 (Zyklus/SDO/PDO/Netzmanagement)	49
6.5	Fehlernummern	50
6.5.1	Hantierungs-Baustein Fehler	50
6.5.2	Abortcodes	51
6.6	SAE J1939 Kommunikation	52
6.6.1	Allgemein	52
6.6.2	FC 67 CANSEND_SAE_J1939	53
6.6.3	FC 68 CANRCV_SAE_J1939	54
<b>7</b>	<b>Anhang</b>	<b>55</b>
7.1	Technische Daten	55
7.2	Steckerbelegung	56

7.3	Verbindungskabel	56
7.4	Weiterführende Dokumentation	56

# 1 Sicherheitshinweise

Beachten Sie die aufgeführten Sicherheitshinweise zu Ihrer eigenen Sicherheit und der Sicherheit Anderer. Die Sicherheitshinweise zeigen Ihnen mögliche Gefahren auf und geben Ihnen Hinweise, wie Sie Gefahrensituationen vermeiden können.

Im vorliegenden Handbuch werden folgende Piktogramme verwendet:



*Achtung, macht auf Gefahren und Fehlerquellen aufmerksam*



*gibt einen Hinweis*



*Gefahr allgemein oder spezifisch*



*Gefahr eines **Stromschlages***

## 1.1 Allgemein

Die CAN 300 Masterbaugruppe wird nur als Bestandteil eines Gesamtsystems eingesetzt.



*Der Betreiber einer Maschinenanlage ist für die Einhaltung der für den speziellen Einsatzfall geltenden Sicherheits- und Unfallverhütungsvorschriften verantwortlich.*



*Bei der Projektierung sind die einsatzspezifischen Sicherheits- und Unfallverhütungsvorschriften zu beachten.*



*Not-Aus-Einrichtungen gemäß EN 60204 / IEC 204 müssen in allen Betriebsarten der Maschinenanlage wirksam bleiben. Es darf zu keinem undefinierten Wiederanlauf der Anlage kommen.*



*In der Maschinenanlage auftretende Fehler, die Material- oder Personenschäden verursachen können, müssen durch zusätzliche externe Einrichtungen abgefangen werden. Diese Einrichtungen müssen auch im Fehlerfall einen sicheren Betriebszustand gewährleisten. Solche Einrichtungen sind z.B. elektromechanische Sicherheitsschalter, mechanische Verriegelungen usw. (siehe EN 954-1, Risikoabschätzung).*



*Sicherheitsrelevante Funktionen niemals über das Bedienterminal ausführen oder einleiten.*





*Zutritt zu den  
Baugruppen nur für  
berechtigte Personen!*

## **1.2 Zugangsbeschränkung**

Die Baugruppen sind offene Betriebsmittel und dürfen nur in elektrischen Betriebsräumen, Schränken oder Gehäusen installiert werden. Der Zugang zu den elektrischen Betriebsräumen, Schränken oder Gehäusen darf nur über Werkzeug oder Schlüssel möglich sein und nur unterwiesenem oder zugelassenem Personal gestattet werden. Siehe auch Kapitel 2.

## **1.3 Benutzerhinweise**

Dieses Handbuch richtet sich an Projektoren und Monteure der CAN 300 Baugruppe.

Es soll dem Projektoren als Programmierhandbuch und Nachschlagewerk dienen. Dem Monteur sollen alle zur Montage notwendigen Daten bereitgestellt werden.

Die CAN 300 Baugruppe ist ausschließlich zum Gebrauch in einem S7-300 Automatisierungsgerät der Firma Siemens. Aus diesem Grund sind von Projektoren, Anwender und Monteur die für den jeweiligen Einsatzfall geltenden Normen, Sicherheits- und Unfallverhütungsvorschriften unbedingt zu beachten. Der Betreiber des Automatisierungssystems ist für die Einhaltung dieser Vorschriften verantwortlich.

## **1.4 Bestimmungsgemäßer Gebrauch**

Die CAN 300 Baugruppe darf nur, wie im Handbuch beschrieben, als Kommunikationssystem verwendet werden.

## **1.5 Bestimmungswidrigen Gebrauch vermeiden!**

Sicherheitsrelevante Funktionen dürfen nicht mit über die CAN 300 Baugruppe allein gesteuert werden.

## 2 Installation und Montage

Die Installation der CAN 300 Baugruppe muß nach VDE 0100 IEC 364 erfolgen. Da es sich um „OPEN Type“ Baugruppen handelt, müssen sie in einen (Schalt-) Schrank eingebaut werden. Umgebungstemperatur: 0 °C – 60 °C.



*Bevor Installationsarbeiten durchgeführt werden, alle Systemkomponenten spannungsfrei schalten.*



*Gefahr eines Stromschlages!*



*Bei der Montage sind die einsatzspezifischen Sicherheits- und Unfallverhütungsvorschriften zu beachten.*

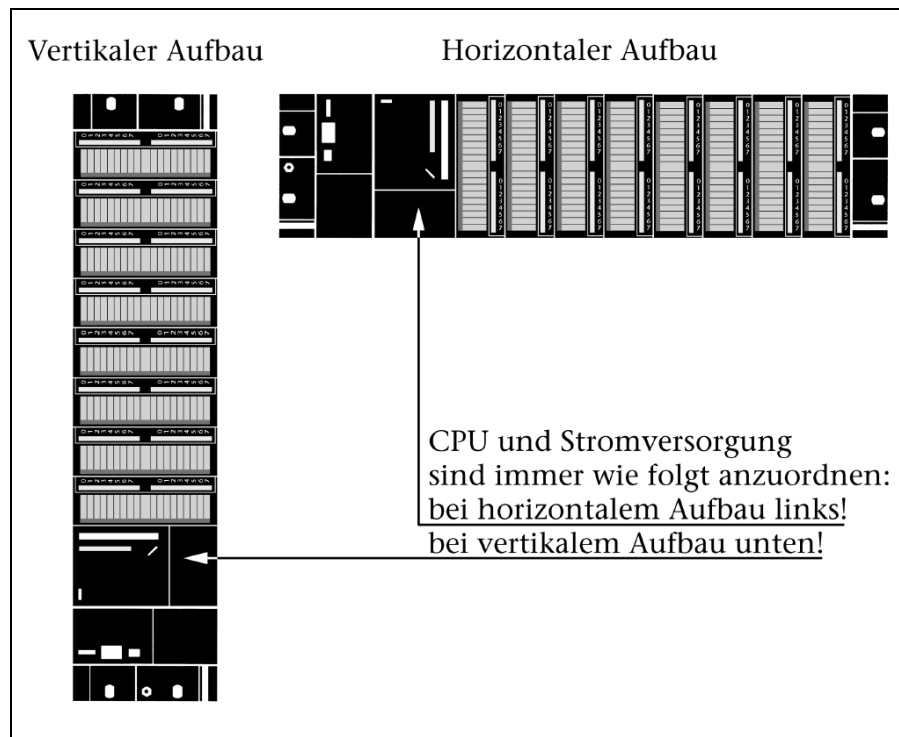
### 2.1 Vertikaler und horizontaler Aufbau

Die Baugruppen können sowohl vertikal als auch horizontal aufgebaut werden.

*Zulässige Umgebungstemperatur:*

bei vertikalem Aufbau: von 0 bis 40 °C

bei horizontalem Aufbau: von 0 bis 60 °C



## 2.2 Mindestabstand

Durch die Einhaltung von Mindestabständen

ist eine Abkühlung der CAN 300 Baugruppen gewährleistet  
ist genügend Raum zum Ein- und Aushängen der Baugruppen vorhanden

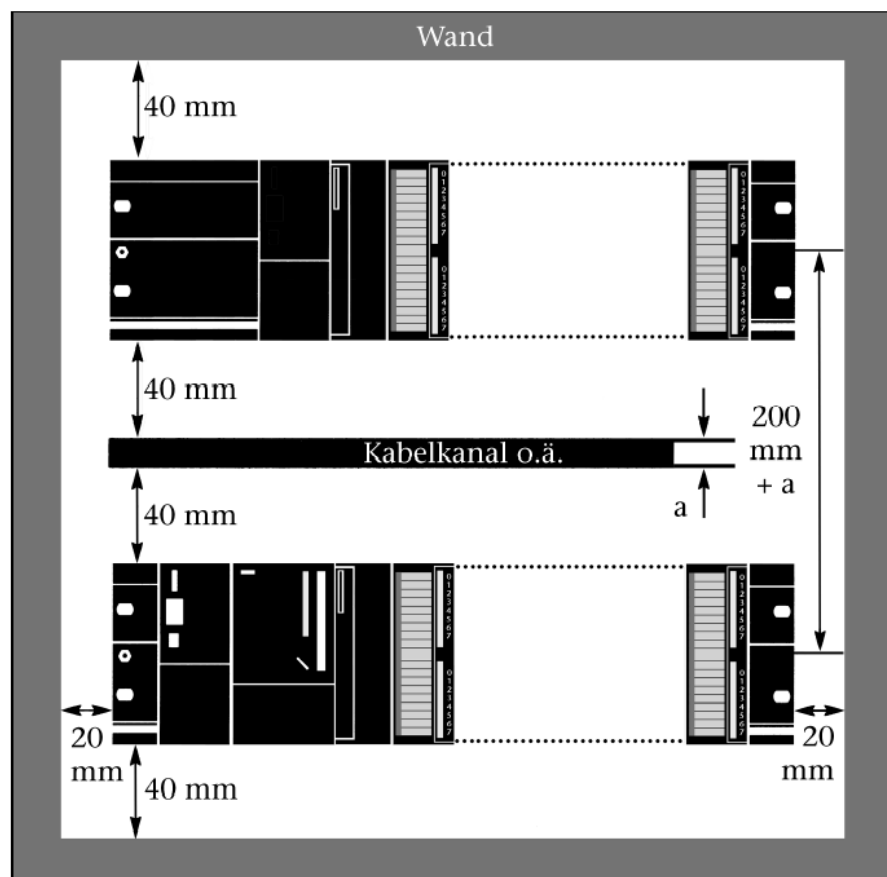
ist genügend Raum zum Verlegen von Leitungen vorhanden

erhöht sich die Einbauhöhe des Baugruppenträgers auf 185 mm,  
wobei trotzdem das Abstandsmaß von 40 mm eingehalten werden muss.

Im folgenden Bild sind für S7-300 Aufbauten auf mehreren Baugruppenträgern die Mindestabstandsmaße zwischen den jeweiligen Baugruppenträgern, sowie zu benachbarten Schrankwänden, Betriebsmitteln, Kabelkanälen etc. angegeben.



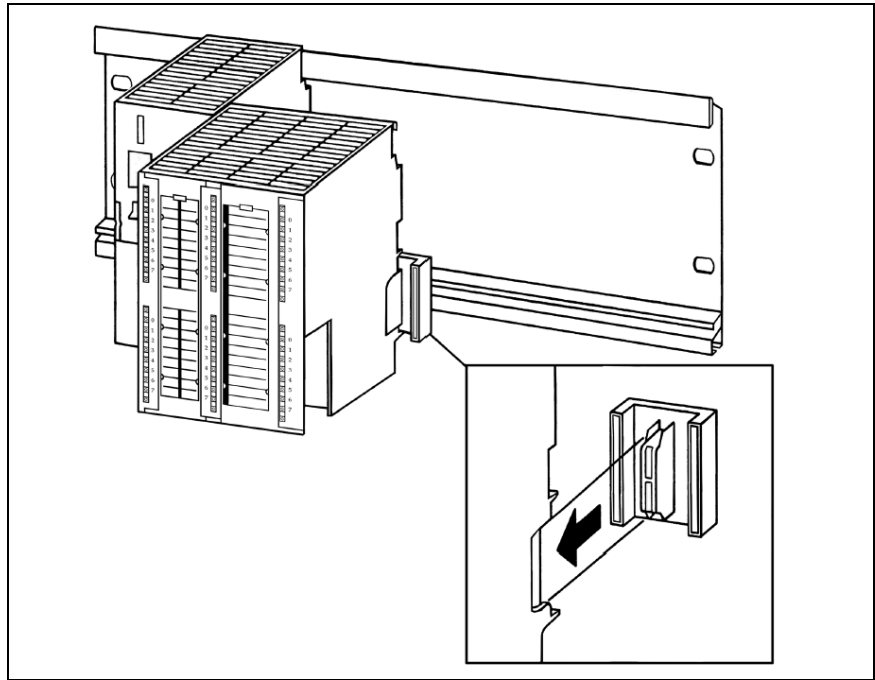
*Nichteinhaltung der Mindestabstände kann die Baugruppe bei hohen Umgebungstemperaturen zerstören!*



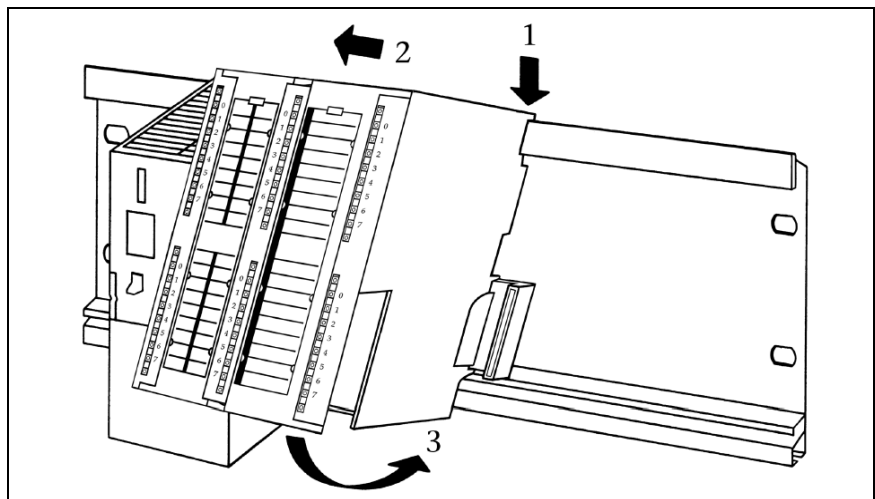
## 2.3 Montage der Baugruppe auf die Profilschiene

Ein Busverbinder liegt jeder Signalbaugruppe bei, nicht aber der CPU. Beim Aufstecken der Busverbinder immer bei der CPU beginnen.

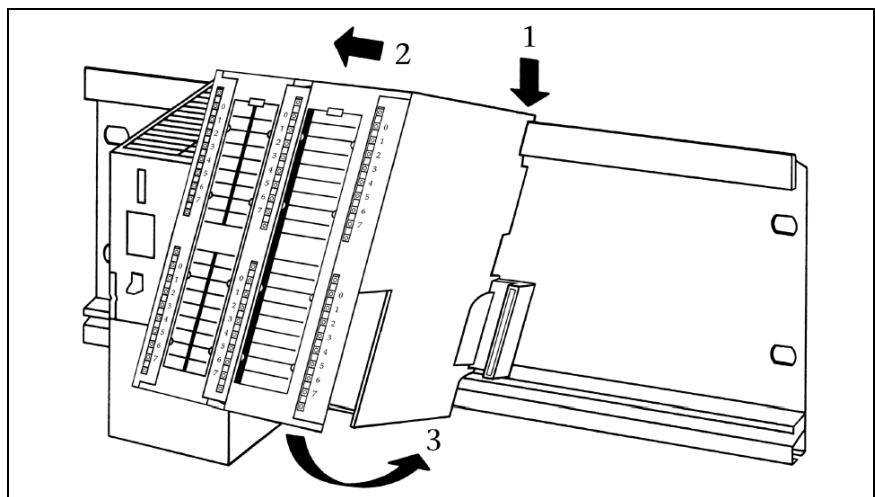
Den Busverbinder von der letzten Baugruppe nehmen und in die CPU stecken. Auf die letzte Baugruppe der Zeile keinen Busverbinder stecken.



Die Baugruppen einhängen (1), bis an die linke Baugruppe heranschieben (2) und nach unten schwenken (3).



Die Baugruppen mit einem Drehmoment von 0,8 bis 1,1 Nm festschrauben.



## 3 Systemübersicht

### 3.1 Anwendung und Funktionsbeschreibung



Die CAN 300 Baugruppe der Systeme Helmholz GmbH ermöglicht es, beliebige CAN-Teilnehmer mit dem Automatisierungsgerät zu verbinden. Die Baugruppe wird auf den Rückwandbus des Automatisierungsgerätes gesteckt. Sie kann sowohl im Zentralrahmen als auch im Erweiterungsrahmen eingesetzt werden.

Die CAN 300 Baugruppe ist im Hardwarekonfigurator als eine Kommunikationsbaugruppe zu parametrieren und belegt 16 Bytes im analogen Prozeßabbild. Der Datenaustausch mit der SPS erfolgt über den Rückwandbus.



Die Hantierungsbausteine, die das einfache Handling der CAN-Kommunikation ermöglichen, werden im Sourcecode mitgeliefert. Es stehen sowohl Hantierungsbausteine für eine einfache Layer 2-Kommunikation, SAE J1939 und Hantierungsbausteine für eine CANopen Master Kommunikation zur Verfügung.

Hantierungssoftware für die Verwendung der CAN 300 Baugruppe als CANopen Slave oder für die Ansteuerung von LENZE-Antrieben ist auf Anfrage erhältlich.

Im Lieferumfang ist weiterhin das Windows Parametriertool „CANParam“ zum Einstellen der CAN Kommunikationsparameter enthalten.

Die CAN 300 Baugruppe unterstützt sowohl CAN 2.0A (11 Bit) als auch CAN 2.0B (29 Bit) Telegramme als Highspeed Node nach ISO 11898-2 mit einer frei wählbaren Baudrate von 10Kbit/s bis 1Mbit/s, oder frei editierbares Bit-Timing.

Die CAN 300 Baugruppe beinhaltet die Netzmanagementfunktionen „Power On“, „Stop->Run“ und „Run->Stop“. Hinter jeder der drei Funktionen kann über eine einfache Makrosprache ein CAN-Bus Verhalten mit bis zu 512 Telegrammen projiziert werden, das automatisch von der Baugruppe ausgeführt wird, wenn das Ereignis eintritt.

Über eine mehrstufige Akzeptanzmaske können die für das Automatisierungsgerät relevanten IDs vorgefiltert werden. Es werden nur die gewünschten CAN-Telegramme übernommen, der Zyklus des Automatisierungsgerätes wird dadurch entlastet.

In der CAN 300 Baugruppe stehen 11 frei einstellbare Timer zur Verfügung. Jeder Timer kann ein frei programmierbares CAN-Telegramm auslösen. Somit sind auch die in der Antriebs- und Servosteuerung weit verbreiteten Synchronprotokolle einfach über die CAN 300 Baugruppe zu realisieren.

Weiterhin besteht die Möglichkeit die Daten über den CAN-Bus nur in einem Zeitfenster senden zu lassen. Die zu versendenden Daten werden vom Automatisierungsgerät azyklisch übertragen und von der CAN 300 Baugruppe nach dem Ablauf der parametrierten Zeit gesendet.

### 3.2 Anschlüsse

Die CAN 300 Baugruppe hat hinter der Frontklappe zwei 9polige SubD-Stecker.

Der obere Stecker ist für den CAN-Bus, der untere SubD-Stecker ist die RS232-Schnittstelle zum PC für die Projektierung der Baugruppe.

Steckerbelegung:

Pin	SubD-Stecker RS232	SubD-Stecker CAN
1	-	-
2	Rx	CAN Low
3	Tx	CAN GND
4	-	-
5	GND	-
6	-	-
7	RTS	CAN High
8	CTS	-
9	-	-



Eine 24V Spannungsversorgung ist nicht auf dem CAN-Bus Stecker aufgelegt.

### 3.3 CAN-Verkabelung

Eine CAN-Busleitung benötigt mindestens 3 Leitungen: CAN High, CAN Low und CAN Ground. Es ist nur eine Linienstrukt erlaubt. An beiden Enden der CAN-Bus Leitung muß ein Abschlußwiderstand von 120 Ohm zwischen CAN High und CAN Low geschaltet sein. Die CAN 300 Baugruppe hat keinen eingebauten Abschlußwiderstand.



In der CAN 300 Baugruppe ist kein Abschlußwiderstand integriert.

Überprüfen Sie die korrekte Verkabelung im Debug Dialog der CANParam (s.a. Kapitel 5.9)

Die maximalen Kabellängen hängen hauptsächlich von der verwendeten Baudrate ab.

Bitrate	Bus Länge	Bit Time
1 Mbit/s	30 m	1 µsek.
800 kBit/s	50 m	1,25 µsek.
500 kBit/s	100 m	2 µsek.
250 kBit/s	250 m	4 µsek.
125 kBit/s	500 m	8 µsek.
20 kBit/s	2500 m	50 µsek.
10 kBit/s	5000 m	100 µsek.

Die angegebenen Leitungslängen sind nur Richtwerte. Die maximale Leitungslänge hängt auch noch von der Anzahl der angeschlossenen Teilnehmer und vom Leitungstyp ab.

Genauere Informationen sind im Dokument „CANopen Recommendation DR 303-1“ erhältlich.

### 3.4 LED-Anzeigen

Die drei LEDs an der Vorderseite der Baugruppe informieren über den Betriebszustand.

#### LED „RUN“ (Grün):

Dauerlicht zeigt an, daß die Baugruppe sich im zyklischen Betrieb befindet. Blinklicht zeigt an, daß die Baugruppe sich im Anlauf befindet, oder daß die SPS im Stop ist. Die Kommunikation mit der CPU ist dann nicht möglich, CAN-Bus Telegramme können dann auch nicht gesendet oder empfangen werden.



#### LED „SF“ (Rot):

Dauerlicht zeigt einen Fehler in der Baugruppe an (z.B. fehlerhaftes Projekt).

Blinklicht zeigt einen CAN-Fehler an. Ein CAN-Fehler liegt vor, wenn die Error Counter nicht Null, der CAN-Status nicht „OK“ ist oder ein CAN-FIFO-Overflow vorliegt [ab FW V.2.6]. Weitere Informationen erhalten Sie im Debug Modus der CANParam Software (s.a. Kap. 5.9).

#### LED „CAN“ (Gelb):

CAN-Bus Aktiv: zeigt eine laufende Kommunikation (Senden und Empfang) über den CAN-Bus an.



Telegramme, die über die Timer der Baugruppe gesendet werden, werden nicht mit der gelben LED angezeigt.

### 3.5 Lieferumfang

Baugruppe CAN 300, Busverbinder

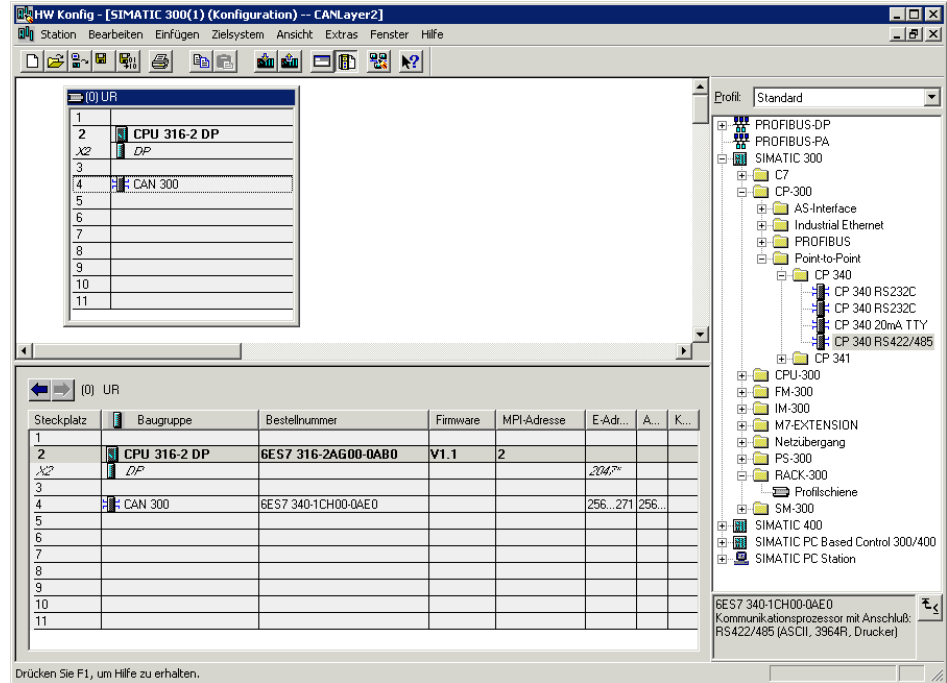
### 3.6 Zubehör

CAN CD mit Parametriersoftware „CANParam“, „Layer 2“ und „CANopen“ Hantierungsbausteinen	800-600-1AA11
CAN CD mit Parametriersoftware „CANParam“, „Layer 2“ und „LENZE-Systembus“ Hantierungsbausteinen	800-600-1LZ11
Handbuch, deutsch/englisch	900-600-CAN01
Programmierskabel PC <-> CAN 300 Baugruppe	700-610-0VK11
CAN-Bus Stecker	700-690-0BA12
CAN-Bus Stecker mit Anschlußstecker	700-690-0BB12
CAN-Bus Stecker mit axialem Kabelabgang	700-690-0CA11

## 4 Projektierung in der SPS

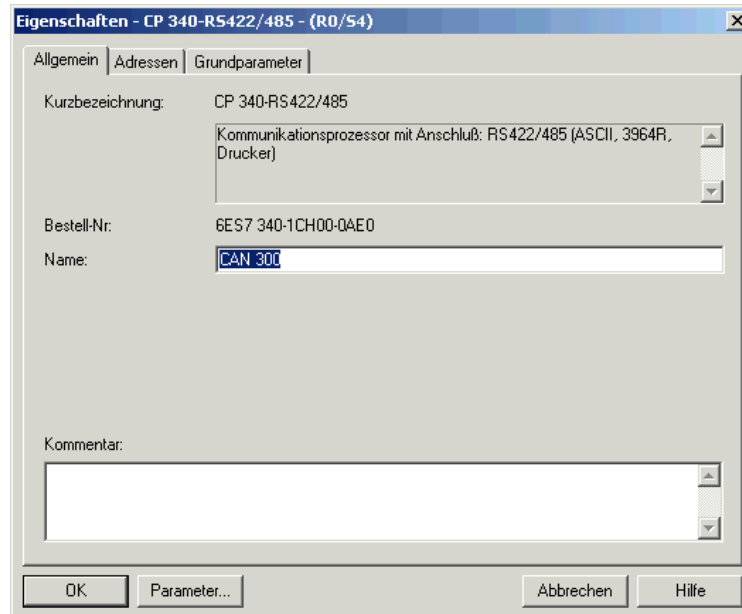
Die CAN 300 Baugruppe wird in der Programmiersoftware der SPS als CP340 Kommunikationsbaugruppe projektiert.

Auf der Installations CD finden Sie ein Step 7 Projekt, daß die benötigten Hantierungsbausteine enthält.



Der Einsatz der CAN300 Baugruppe in einem ET200M System ist nicht möglich.

Die Baugruppe kann überall dort eingesetzt werden, wo auch eine CP-Baugruppe erlaubt ist, d.h. auch im Erweiterungsrahmen nach einer Anschaltung.





Bei der Parametrierung der Baugruppe ist nur der Bereich der E/A-Adressen relevant. Alle anderen Einstellung haben keine Auswirkung auf die Baugruppe.



Die Adressen für die Eingänge und die Ausgänge müssen immer gleich sein, damit die Hantierungssoftware korrekt zugreifen kann.



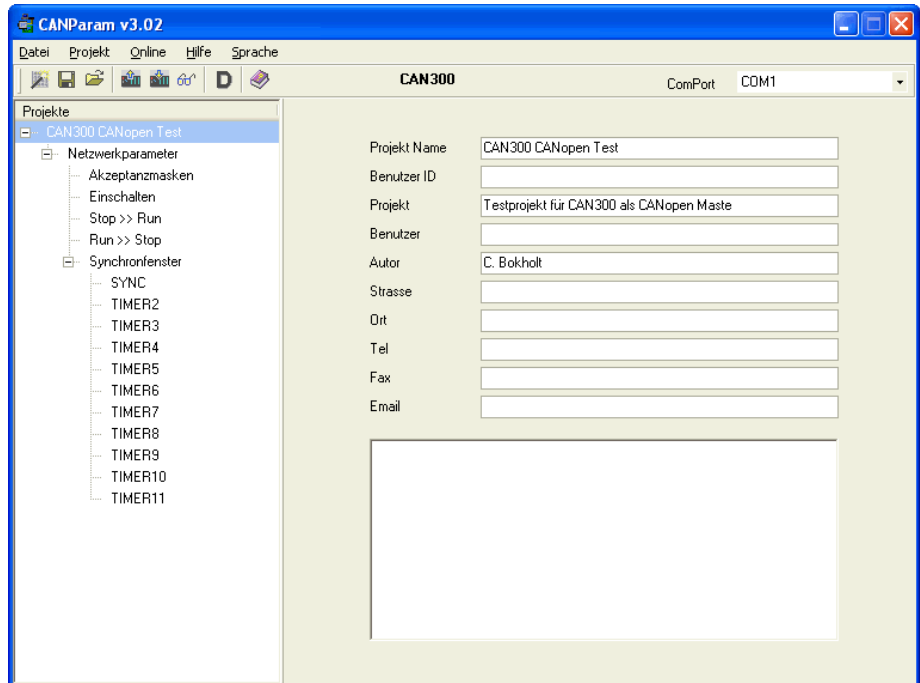
Die E/A-Adressen dürfen **nicht** im zyklischen Prozessabbild liegen!

The screenshot shows a software dialog box titled "Eigenschaften - CP 340-R5422/485 - (R0/54)". It has three tabs: "Allgemein", "Adressen", and "Grundparameter". The "Adressen" tab is active. It contains two sections: "Eingänge" and "Ausgänge". Each section has input fields for "Anfang" (256) and "Ende" (271), a "Prozeßabbild:" dropdown menu (set to "..."), and a checked checkbox for "Systemvorgabe". At the bottom, there are buttons for "OK", "Parameter...", "Abbrechen", and "Hilfe".

Beachten Sie, daß die Adressen der Baugruppe nicht innerhalb des (zyklischen) Prozeßabbildes liegen.

## 5 Projektierung der CAN 300 Baugruppe

Die Projektierung der CAN 300 Baugruppe wird auf dem PC mit der „CANParam V3“-Software durchgeführt. Diese Software wird zusammen mit den Hantierungsbausteinen für die S7 geliefert und ist auf jedem Windows 2000/XP Rechner lauffähig.

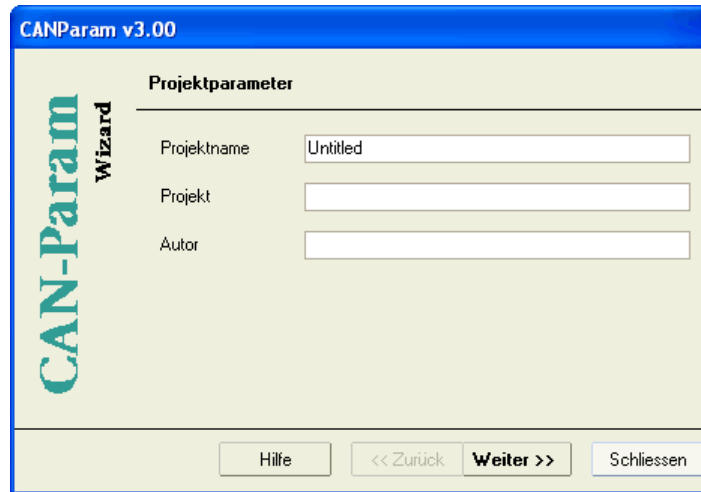


Die Projektierung einer Baugruppe kann in einer Projektdatei auf dem PC abgelegt werden.

Zur Verbindung zwischen dem PC und der CAN 300 Baugruppe kann ein handelsübliches Nullmodem-Kabel verwendet werden (siehe auch Kapitel 7.3). Nach der Installation und Start der CANParam Software sollte rechts oben in der Menüzeile die Schnittstelle eingestellt werden.

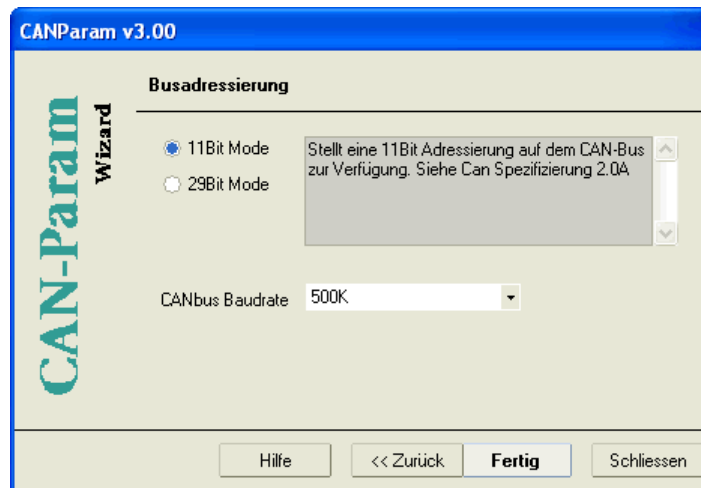
## 5.1 Erstellen eines neuen Projektes

Ein neues Projekt kann über die Funktion „Projekt / Projekt erstellen / neues Projekt“ oder mit dem Projekt Wizard erstellt werden.



The screenshot shows the 'Projektparameter' (Project Parameters) screen of the CAN-Param v3.00 Wizard. The window title is 'CANParam v3.00'. On the left side, there is a vertical logo for 'CAN-Param Wizard'. The main area is titled 'Projektparameter' and contains three input fields: 'Projektname' with the value 'Untitled', 'Projekt', and 'Autor'. At the bottom, there are four buttons: 'Hilfe', '<< Zurück', 'Weiter >>', and 'Schliessen'.

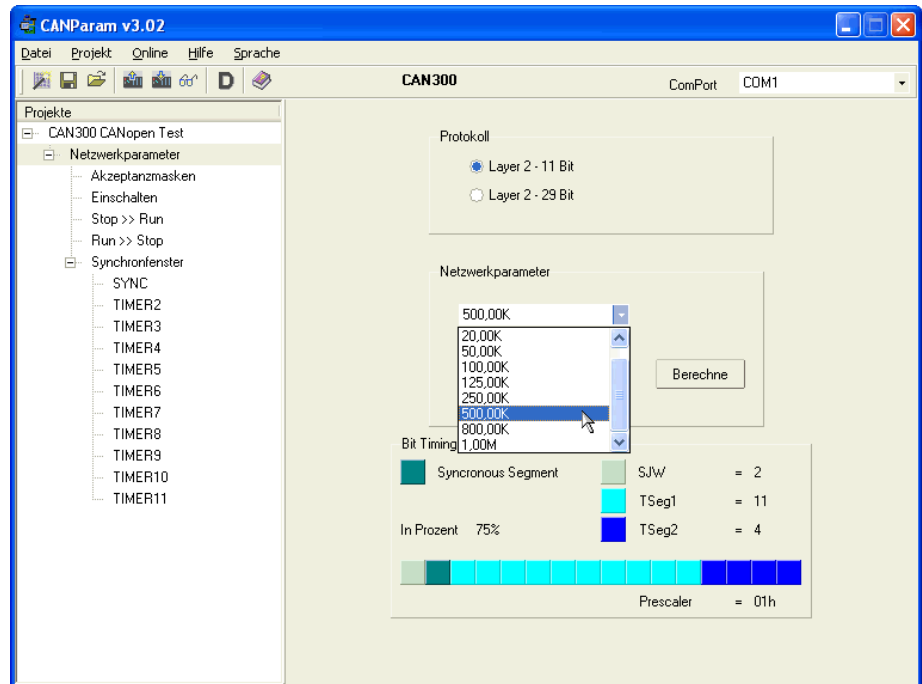
Der Projekt Wizard führt Sie durch die wichtigsten Einstellungen um ein neues und vollständiges Projekt zu erhalten.



The screenshot shows the 'Busadressierung' (Bus Addressing) screen of the CAN-Param v3.00 Wizard. The window title is 'CANParam v3.00'. On the left side, there is a vertical logo for 'CAN-Param Wizard'. The main area is titled 'Busadressierung' and contains two radio buttons: '11Bit Mode' (selected) and '29Bit Mode'. To the right of the radio buttons is a text box with the text: 'Stellt eine 11Bit Adressierung auf dem CAN-Bus zur Verfügung. Siehe Can Spezifizierung 2.0A'. Below this is a dropdown menu for 'CANbus Baudrate' with the value '500K'. At the bottom, there are four buttons: 'Hilfe', '<< Zurück', 'Fertig', and 'Schliessen'.

## 5.2 Einstellung der CAN-Bus Baudrate

Die CAN-Baudrate kann im Bereich von 10kBit/s bis 1Mbit/s gewählt werden.



Für spezielle Einsatzfälle kann das Bit Timing der Übertragung direkt definiert werden. Eine genaue Beschreibung des Bit Timing Aufbaus entnehmen Sie bitte der CAN Spezifikation 2.0 Part B ab Kapitel 10.

## 5.3 Einstellung der Übertragungsart (Protokoll)

Die CAN 300 Baugruppe unterstützt sowohl das Protokollformat CAN 2.0A (11 Bit), als auch CAN 2.0B (29 Bit).

Für die Verwendung der CANopen Hantierungsbausteine muß hier immer CAN 2.0A (11 Bit) gewählt werden.

Für die Verwendung der SAE J1939 Hantierungsbausteine muß immer CAN 2.0B (29 Bit) gewählt werden.

## 5.4 Akzeptanzmasken

Es stehen in der CAN 300 Baugruppe 16 Akzeptanzmasken zur Verfügung. Mit Hilfe dieser Masken ist es möglich, verschiedene Telegramm-IDs für den Empfang freizuschalten oder zu sperren.

	Anfang	Ende
<input checked="" type="checkbox"/> Maske 1	0x000	0x7FF
<input type="checkbox"/> Maske 2	0x000	0x000
<input type="checkbox"/> Maske 3	0x000	0x000
<input type="checkbox"/> Maske 4	0x000	0x000
<input type="checkbox"/> Express Maske	0x000	0x000
<hr/>		
<input type="checkbox"/> Maske 6	0x000	0x000
<input type="checkbox"/> Maske 7	0x000	0x000
<input type="checkbox"/> Maske 8	0x000	0x000
<input type="checkbox"/> Maske 9	0x000	0x000
<input type="checkbox"/> Maske 10	0x000	0x000
<input type="checkbox"/> Maske 11	0x000	0x000
<input type="checkbox"/> Maske 12	0x000	0x000
<input type="checkbox"/> Maske 13	0x000	0x000
<input type="checkbox"/> Maske 14	0x000	0x000
<input type="checkbox"/> Maske 15	0x000	0x000
<input type="checkbox"/> Maske 16	0x000	0x000



Die Defaulteinstellung der Akzeptanzmaske (0h bis 7FFh) erlaubt den Empfang aller Telegramme.

Mit der Akzeptanzmaske „Express Queue“ ist es möglich hochpriorie CAN-Telegramme zu behandeln. Telegramme die mit den dort eingestellten IDs empfangen werden, werden an dem normalen Empfangspuffer vorbei als nächstes Telegramm an die S7 übergeben.

## 5.5 Netzmanagement

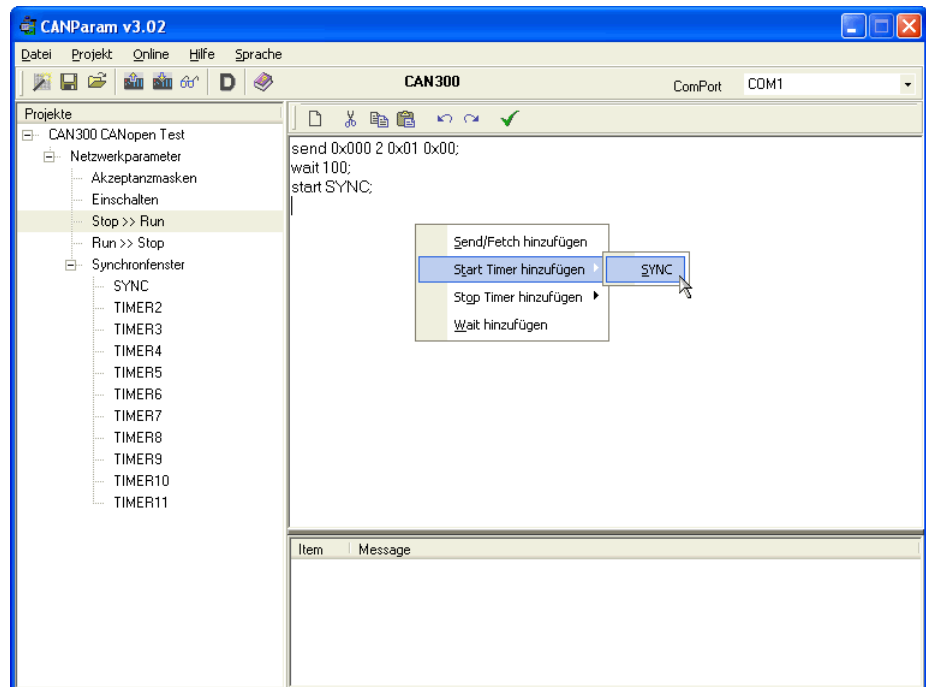
Die CAN 300 Baugruppe kann für die SPS-Ereignisse „Power ON“, „Stop -> Run“ und „Run -> Stop“ frei programmierbar CAN-Telegramme (Layer 2) senden sowie Timer starten und anhalten.

Folgende Befehle stehen zur Verfügung:

<b>Send</b>	Sende Telegramm (Aufbau: ID, Länge, Datenbyte 1, Datenbyte 2, etc. )
<b>Fetch</b>	Sende Telegramm mit RTR-Bit 1
<b>Start</b>	Start Timer X
<b>Stop</b>	Stoppe Timer X
<b>Wait</b>	Warte X ms
<b>//</b>	Kommentarzeile



Die Abarbeitung der Schritte in den Skripten wird in einem Raster von 50ms durchgeführt.



## 5.6 Timer

Für zeitabhängige Ereignisse stehen in der CAN 300 Baugruppe 11 Timer zur Verfügung. Jeder Timer kann ein beliebiges CAN-Telegramm senden.

The screenshot shows the configuration for a timer in the CAN 300 interface. It is divided into two main sections: 'Timer' and 'Aktion'.

**Timer Section:**

- Alias:
- Wiederholung:  msec
- Phase:  msec

**Aktion Section:**

- ID:
- Fetch
- RTR Länge:
- Daten: A list of 8 data bytes, each in a  field. Bytes 7 and 8 are disabled (greyed out).

Jedem Timer kann ein Alias-Name gegeben werden. Dieser Name kann dann in den Skripten der SPS-Ereignisse verwendet werden.

Die *Timer-Periode* gibt das Wiederholintervall für den Timer an, die *Phase* den Startpunkt innerhalb des Intervalls.

Für die *Timer-Periode* können Zeiten von 5 mSek. bis 1 Sek. in 5 mSek-Schritten eingestellt werden. Für die *Phase* 0mSek. bis 5 mSek. vor der Periodendauer.

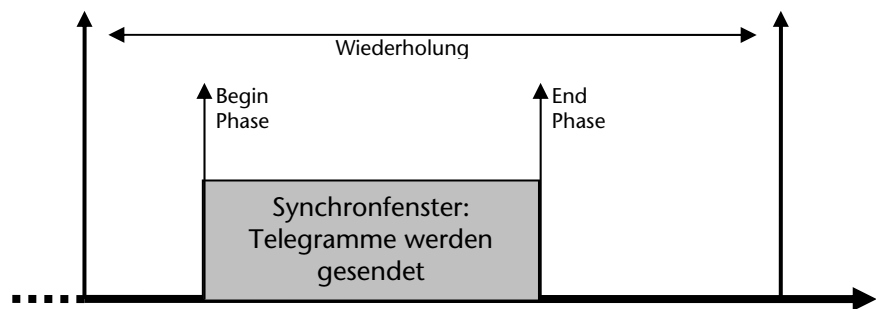
Die Daten des für den Timer definierten CAN-Telegrammes sind Initialisierungs-Daten und können durch die S7-CPU im zyklischen Betrieb durch den FC 63 „CANSYNCSSEND“ überschrieben werden.

## 5.7 Synchronfenster

Bei Verwendung des Synchron-Timer (Einstellung „Synchron Fenster“), werden die durch den FC60 „CANSEND“ gesendeten Telegramme innerhalb eines Zeitfensters gesendet. „Wiederholung“ gibt die Wiederholungsrate an, „Begin phase“ & „End phase“ legen den Bereich des Sendefensters innerhalb der Wiederholungszeit fest.

Die zu sendenden Telegramme werden nur innerhalb des Zeitbereiches zwischen „Begin phase“ & „End phase“ gesendet.

Ausserhalb des Synchronfensters wird damit Zeit auf dem Bus für Kommunikation von anderen Teilnehmern geschaffen.



Die Timer 1 würden bei der Verwendung des Synchronfensters intern verwendet.

Die Funktionalität der anderen Timer werden vom Synchronfenster nicht beeinflusst, d.h. diese können auch ausserhalb des Synchronfensters gesendet werden.

## 5.8 Herunterladen



Die CPU muß im STOP sein, damit die Projektübertragung möglich ist!

Das aktuell bearbeitete Projekt kann jederzeit neu in die CAN 300 Baugruppe eingespielt werden („Herunterladen“).

Das Hochladen eines Projektes aus der CAN 300 Baugruppe ist z.Zt. nicht möglich.



## 5.9 Diagnose/Debug

Zur Vereinfachung der Fehlersuche kann der Zustand der CAN 300 Baugruppe über den Menüpunkt „Debug“ abgefragt werden. Der Debug-Modus benötigt eine serielle Verbindung zur Baugruppe.

The screenshot shows a software window titled "Zielgerät ist CAN300". It contains several input fields for configuration and status:

- Version:** 2.3
- Protokoll:** 11Bit Modus
- Bit Timing:** 0x3A41
- Baudrate:** 500,00K
- SJW:** 2
- Fehlerzähler:** Tx Fehler Zähler: 0x00, Rx Fehler Zähler: 0x00
- Buffer Pointer:** CAN0 TxHead: 0x18, CAN0 TxTail: 0x18, CAN0 RxHead: 0x18, CAN0 RxTail: 0x18
- CAN Status:** 0x0480
- Node Status:** OK
- TR Anfrage:** 0x0000
- Status der Statemaschine:** 0x01
- Letzter Fehler der Event Funktion:** 0x01

At the bottom, there is a "Kanal 1" label, a "Trennen" button, and a "Schliessen" button.

Mit dem Button „Verbinden“ wird der Überwachungsmodus aktiviert. Nochmaliges Drücken des Buttons unterbricht die Verbindung wieder.

### Folgende Informationen liefert der Debug-Dialog:

<b>Version</b>	Versionsnummer des CAN 300 Betriebssystems
<b>Protokoll</b>	projektiertes CAN-Protokoll (11Bit/29Bit/etc.)
<b>Bit Timing</b>	Inhalt des Bittiming Registers, Baudrate und SJW werden im Klartext angezeigt
<b>CAN Status</b>	Inhalt des CAN-Status-Registers: Bit 15: Transmitting Bit 14: Receiving Bit 10: Node state transition occurred Bit 9+8: Node State (wird auch Klartext angezeigt) Bit 7: Enable Tx Bit 0: Halt/Bus Off State of CAN Controller
<b>Node-Status</b>	Inhalt Bits 9+8 des CAN-Status-Registers (s.o.): "OK", "Warning", "Passiv", "Bus Off"

**!**  
Node Status muß auf „OK“ stehen, damit eine CAN-Datenübertragung möglich ist.

**TR Request** Anzeige der Sende- & Empfangsanforderung:  
 Bit 15: Scriptverarbeitung  
 Bit 13 & 14: Asynchron Sendepuffer  
 Bit 11 - 1: Timer 10 – 0  
 Bit 0: Receivepuffer



*Error Counters* muß auf „0“ stehen, ansonsten ist die Datenübertragung am CAN-Bus gestört.

**Error counters** TX: Fehlerzähler Senden  
 RX: Fehlerzähler Empfang

**Hinweis:** Die Sende und Empfangsfehlerzähler werden vom CAN-Controller hochgezählt, wenn das Senden oder der Empfang eines Telegrammes fehlgeschlagen ist. Sobald dann wieder ein Telegramm korrekt gesendet, oder empfangen wurde, wird der entsprechende Zähler wieder heruntergezählt. Diese Zähler sollten bei einem korrekt arbeitenden CAN-Bus immer auf 0 stehen!

**Buffer pointers** Anzeige der Ringpufferzeiger:  
 TX Head: Sendedaten von S7  
 TX Tail: gesendete Daten über CAN  
 RX Head: Empfangsdaten von CAN  
 RX Tail: an S7 gesendete Empfangsdaten

**Hinweis:** Die CAN 300 Baugruppe hat einen Empfangspuffer und einen Sendepuffer von jeweils 128 Telegrammen. Die Buffer pointer zeigen an, wie weit die Puffer gefüllt sind. Wurde z.B. ein CAN-Telegramm empfangen, so wird „Rx Head“ hochgezählt. Wurde das Telegramm dann an die SPS weitergereicht (vom Hantierungsbaustein abgeholt), so wird „Rx Tail“ hochgezählt (Schreib-/Lesezeigerprinzip). Zwischen den Zeigernpaaren sollte nie eine grosse Differenz entstehen. Ist dies doch der Fall, so werden die CAN-Telegramme nicht schnell genug von der SPS abgeholt, oder zu schnell von der SPS gesendet.

**Status of state machine**

Entspricht dem Byte STAT (siehe Kap. 6.2.6).

**Last error from Event function**

Letzter Fehler der Scriptverarbeitung (interne Information für den Support)

**5.10 Firmware update**

Die CAN 300 Baugruppe kann über die Funktion „Firmware update“ im Menü „Online“ mit einer neuen Firmware versehen werden.

Dazu wird nur eine funktionierende serielle Verbindung benötigt. Die Firmwaredateien liegen im Ordner „binary“ im CANParam Programmverzeichnis.



*Bitte beachten Sie, daß durch unsachgemäße Durchführung des Updates die Baugruppe zerstört werden kann!*

## 6 Programmierung in der SPS

### 6.1 Übersicht

Die Programmierung der CAN 300 Baugruppe erfolgt in der SPS über die mitgelieferten Hantierungsbausteine.

Es stehen Hantierungsbausteine für eine reine Layer 2 Kommunikation (keine Interpretation der CAN-Telegramme), für das Protokoll SAE J1939 und für die Kommunikation mit CANopen Teilnehmern als Master zur Verfügung.

### 6.2 Layer 2 Kommunikation

#### 6.2.1 Allgemein

Mit den Layer 2 Hantierungsbausteinen ist es möglich jedes beliebige CAN-Protokoll zu verarbeiten. Die CAN-Telegramme werden ohne Interpretation empfangen, der SPS zur Verfügung gestellt und es können beliebige Telegramme versendet werden.

Folgende FCs stehen in Layer 2 zur Verfügung:

<b>FC 60</b>	<b>CANSEND</b>	Senden eines CAN-Telegrammes
<b>FC 61</b>	<b>CANRCV</b>	Empfangen eines CAN-Telegrammes
<b>FC 62</b>	<b>CANCTRL</b>	Baugruppen Managementfunktionen
<b>FC 63</b>	<b>CANSENDTIMER</b>	Ändern der Sendedaten eines Timers

Eine Initialisierung der Baugruppe in den Anlauf-OB's ist nicht erforderlich. Die Baugruppe startet automatisch, wenn die SPS in RUN geschaltet wird und stoppt, wenn die SPS in STOP geht.

Hier ein Aufrufbeispiel:

```
CALL FC 61
Base :=256
IDHI :=MW60
IDLO :=MW62
RTRLEN:=MW64
DW0 :=MW70
DW1 :=MW72
DW2 :=MW74
DW3 :=MW76
STAT :=MB66
Recd :=M67.0
UN M 67.0
SPB send
...

send: SET
= M 87.0
CALL FC 60
Base :=256
IDHI :=W#16#0
IDLO :=W#16#202
RTRLEN:=W#16#8
DW0 :=MW90
DW1 :=MW92
DW2 :=MW94
DW3 :=MW96
STAT :=MB86
Snd :=M87.0
BE
```

### 6.2.2 FC 60 CANSEND

Der Funktionsbaustein CANSEND (FC 60) überträgt ein CAN-Telegramm zur Baugruppe, von der dieses sofort gesendet wird.

Parameter	Richtung	Typ	Beispiel
Base	IN	INT	256
IDHI	IN	WORD	W#16#0
IDLO	IN	WORD	W#16#202
RTRLEN	IN	WORD	W#16#8
DW0	IN	WORD	MW90
DW1	IN	WORD	MW92
DW2	IN	WORD	MW94
DW3	IN	WORD	MW96
STAT	OUT	BYTE	MB 86
Snd	IN_OUT	BIT	M 87.0

Als Übergabeparameter muss die Basisadresse der Baugruppe als Integerzahl (Base), ein Statusbyte (STAT) und ein Bit zur Sendefreigabe (Snd) übergeben werden.

Das Wort RTRLEN enthält in den unteren 4 Bits (Bit 0 bis Bit 3) die Anzahl der Datenbytes (0...8). Das Bit 4 ist das RTR-Bit des CAN-Telegramms.

Das Bit Snd wird nach dem Durchlauf des Bausteines immer zurückgesetzt, das zu sendende Telegramm wird immer in die Baugruppe übertragen. Sollte der Sendepuffer in der Baugruppe voll sein, so werden ältere, nicht gesendete Telegramme gelöscht. Um dieses zu verhindern, sollte das Bit 4 des STAT Bytes immer vor dem Senden abgefragt werden.

Im Byte STAT steht der Status der CAN 300 Baugruppe (siehe Kap. 6.2.6). Das Byte wird immer versorgt, auch wenn das Snd Bit nicht gesetzt ist. Es empfiehlt sich den Baustein CANRCV immer vor dem Baustein CANSEND aufzurufen, damit die Statusinformationen aktuell vorliegen.

Sollte der Timer 0 als Synchron-Timer eingestellt worden sein, so werden die Daten immer nur in dem definierten Synchron-Zeitfenster gesendet.

Zuordnung der Parameter ID-HI und ID-LO:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	COB-ID 11 Bit										
0	0	0	COB-ID 29Bit																												
ID-HI													ID-LO																		



*Der FC60 „CANSEND“ darf nicht im OB 1 (Zyklus) und OB 35 (Zeit-OBs) gleichzeitig bzw. gemischt aufgerufen werden!*

### 6.2.3 FC 61 CANRCV

Der Funktionsbaustein CANRCV (FC 61) überträgt ein CAN-Telegramm von der Baugruppe in die SPS, wenn ein Telegramm empfangen worden ist und dieses auch durch die Akzeptanzfilter durchgelassen worden ist.

Parameter	Richtung	Typ	Beispiel
Base	IN	INT	256
IDHI	OUT	WORD	MW60
IDLO	OUT	WORD	MW62
RTRLEN	OUT	WORD	MW64
DW0	OUT	WORD	MW70
DW1	OUT	WORD	MW72
DW2	OUT	WORD	MW74
DW3	OUT	WORD	MW76
STAT	OUT	BYTE	MB 66
Rcvd	IN_OUT	BIT	M 67.0

Als Übergabeparameter muss die Basisadresse der Baugruppe als Integerzahl (Base) übergeben werden.

Die Elemente des Telegramms werden als Datenworte übergeben (IDHI, IDLO, RTRLEN, DW0...3).

Das Wort RTRLEN enthält in den unteren 4 Bits (Bit 0 bis Bit 3) die Anzahl der Datenbytes (0...8). Das Bit 4 ist das RTR-Bit des CAN-Telegramms.

Hat der Funktionsbaustein ein Telegramm von der CAN 300 Baugruppe gelesen, so wird das Bit Recd gesetzt.

Im Byte STAT steht der Status der CAN 300 Baugruppe (siehe Kap. 6.2.6). Das Byte wird immer versorgt, auch wenn kein Telegramm empfangen wurde. Es empfiehlt sich den Baustein CANRCV immer vor dem Baustein CANSEND aufzurufen, damit die Statusinformationen aktuell vorliegen.

Zuordnung der Parameter ID-HI und ID-LO:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	COB-ID 11 Bit										
0	0	0	COB-ID 29Bit																												
ID-HI														ID-LO																	

#### 6.2.4 FC 63 CANSENDTIMER

Der Funktionsbaustein CANSENDTIMER (FC 63) überträgt ein CAN-Telegramm zur Baugruppe um die Daten eines in der Baugruppe parametrisierten Timers zu aktualisieren. Die CAN 300 Baugruppe sucht anhand des IDs den zum Telegramm passenden Timer und übernimmt die Daten in den Timer. Die neuen Daten werden mit dem nächsten Timer-Event versendet.

Parameter	Richtung	Typ	Beispiel
Base	IN	INT	256
IDHI	IN	WORD	W#16#0
IDLO	IN	WORD	W#16#203
RTRLEN	IN	WORD	W#16#8
DW0	IN	WORD	DBW6
DW1	IN	WORD	DBW8
DW2	IN	WORD	DBW10
DW3	IN	WORD	DBW12
STAT	OUT	BYTE	MB 2
Snd	IN_OUT	BIT	M 1.0

Als Übergabeparameter muß die Basisadresse der Baugruppe als Integerzahl (*Base*), ein Statusbyte (*STAT*) und ein Bit zur Sende-freigabe (*Snd*) übergeben werden.

Die Elemente des Telegramms werden als Quelldatenworte über-geben (*IDHI*, *IDLO*, *DW0* . . 3). Die ID muss mit der ID des projektierten Timers übereinstimmen.

Das Bit *Snd* wird nach dem Durchlauf des Bausteines immer zurückgesetzt, das zu sendende Telegramm wird immer in die Baugruppe übertragen.

Im Byte *STAT* steht der Status der CAN 300 Baugruppe (siehe Kap. 6.2.6). Das Byte wird immer versorgt, auch wenn kein Telegramm empfangen wurde.



*IDHI/IDLO muss mit der ID des projektierten Timers übereinstimmen!*

### 6.2.5 FC 62 CANCTRL

Der Funktionsbaustein CANCTRL (FC 62) stellt Sonderfunktionen der CAN 300 Baugruppe zur Verfügung.

Parameter	Richtung	Typ	Beispiel
Base	IN	INT	256
Func	IN	INT	1
Param	IN	INT	2
STAT	OUT	BYTE	MB 2

Als Übergabeparameter muß die Basisadresse der Baugruppe als Integerzahl (`Base`), einen Funktionscode (`Func`) und bei Bedarf einen Parameter (`Param`) angegeben werden.

Folgende Funktionen stehen zur Verfügung:

- 1 = Timer starten (Timernummer 1-11 in `Param`)
- 2 = Timer anhalten (Timernummer 1-11 in `Param`)
- 4 = Reset der Baugruppe auslösen

Im Byte `STAT` steht der Status der CAN 300 Baugruppe (siehe Kap. 6.2.6). Das Byte wird immer versorgt, auch wenn kein Telegramm empfangen wurde.



*Bei vielen Anwendungen ist es notwendig, eine Reihe von Telegrammen in einem Zyklus an die Baugruppe zu senden. Der FIFO Umlaufpuffer ist 128 Telegramme lang. Wenn das Bit 4 des Statusbytes gesetzt wird, ist es möglich noch 64 Telegramme auf einmal in die Baugruppe zu senden.*

### 6.2.6 Inhalt des Statusbytes STAT

Das Statusbyte STAT hat bei allen Hantierungsbausteinen die gleiche Bedeutung und zeigt den Zustand der Baugruppe an:

Bit 0: Baugruppe läuft, Einlesen der Parameter abgeschlossen.

Bit 1+2: CAN-Controller Status Register (*ab FW 2.6*):

00 = OK	01 = Warning
10 = Error passiv	11 = Bus OFF

Bit 3: CAN-Receive FIFO voll. Werden weitere Telegramme empfangen, so werden ältere, nicht abgeholte Telegramme gelöscht. (*ab FW 2.6*)

Bit 4: CAN-Send FIFO mehr als Halbvoll, Overflow droht, die S7 sollte keine weiteren Telegramme mehr in die Baugruppe übertragen.

Bit 5: CAN-Receive FIFO mehr als Halbvoll, Overflow droht, die S7 sollte den FIFO schneller auslesen.

Bit 6: CAN-Send FIFO voll. Werden weitere Telegramme in die Baugruppe übertragen, so werden ältere, nicht gesendete Telegramme gelöscht.

Bit 7: Baugruppe in Reset/Programming Mode



# CANopen



CIA = CAN in Automation e.V., Am Weichselgarten 26, 91085 Erlangen



*CANopen arbeitet immer mit CAN 2.0A (11 Bit). Dieses muß bei der Projektierung der Baugruppe mit CANParam berücksichtigt werden.*

## 6.3 CANopen Kommunikation

### 6.3.1 Allgemein

Das CANopen Protokoll ist ein Schicht 7-Protokoll (Application Layer), das auf den CAN-Bus (ISO 11898) aufsetzt. Die Schichten 1 & 2 (Physical Layer/Data Link Layer) vom CAN-Bus bleiben unberührt.

Die CANopen Kommunikationsprofile für die verschiedenen Anwendungen werden von der CIA verwaltet.

Die von der Anwendungsschicht bereitgestellten Dienstelemente ermöglichen die Realisierung einer über das Netzwerk verteilten Applikation. Diese Dienstelemente sind in der „CAN Application Layer (CAL) for Industrial Applications“ beschrieben.

Der 11 Bit Identifier und die 8 Datenbytes eines CAN-Layer 2 Telegrammes bekommen eine feste Bedeutung.

Jedes Gerät in einem CANopen-Netz hat eine feste Node-ID (Modulnummer, 1-127).

### 6.3.2 Objekte

Der Datenaustausch mit einem CANopen Slave erfolgt entweder über fest definierte Servicedaten-Objekte (SDO) oder über frei konfigurierbare Prozessdaten-Objekte (PDO).

Jeder CANopen Slave besitzt ein festes Verzeichniss von SDOs, die über eine Objektnummer (16 Bit) und einen Index (8 Bit) angesprochen werden.

*Beispiel:* Objekt 0x1000/ Index 0 = Device Type, 32Bit Unsigned

SDOs mit 8/16/32 Bit Breite können mit einem CANopen-Telegramm gelesen und geschrieben werden. SDOs die länger sind, werden über mehrere Telegramme übertragen. Für sehr grosse Datenmengen ist die SDO-Blockübertragung vorgesehen.

SDOs können bearbeitet werden, sobald ein CANopen Slave betriebsbereit ist. Für die SDOs stehen nur die COB-ID Funktionen „SDO Anforderung“ oder „SDO Antwort“ bereit. Die Objekt-nummer, der Zugriffsmodus und Typ werden in den ersten 4 Bytes des CAN-Telegrammes hinterlegt.

Die letzten 4 Bytes des CAN-Telegrammes enthalten dann den Wert für das SDO.



Jedem CANopen Slave sollte ein Verzeichnis der von ihm unterstützten Objekte beiliegen.

PDOs enthalten die „Arbeitswerte“ eines CANopen Slaves für den zyklischen Prozeßbetrieb. Jeder CANopen Slave kann mehrere PDOs verwalten (im Normalfall bis zu 4 zum Senden und 4 zum Empfangen).

Jedes der vorhandenen PDOs hat eine eigene COB-ID. In den 8 Datenbytes des Telegrammes können beliebige Informationen des CANopen Slaves zum Lesen und Beschreiben „gemapped“ werden. Dieses können sowohl bereits vorhandene SDOs sein, als auch Aktualwerte des Slaves (z.B. Analogwert eines Einganges).

Die PDOs werden von den meisten CANopen Slaves automatisch beim Anlauf gemapped. Die Zuordnung kann dann über bestimmte SDOs geändert werden.

### 6.3.3 Funktionen

Die CANopen Funktionen teilen sich in drei Grundarten auf:

- SDO Lesen und Schreiben
- PDO Lesen und Schreiben
- Netzmanagement

Der Funktionscode wird in den oberen 4 Bit des Identifiers hinterlegt. Zusammen mit der Node-ID ergeben sie den COB-Identifizier.

COB-Identifizier (COB-ID):

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



Es ist mit speziellen Servicedata-Objekten (SDOs) möglich, einige COB-IDs auf andere Werte zu ändern. Dieses wird von der CANopen-Hantierung NICHT unterstützt!

Broadcast-Funktionen:

Funktion	Function code (binary)	Resulting COB-ID
NMT	0000	0h
SYNC	0001	80h
TIME STAMP	0010	100h

Node Funktionen:

Funktion	Function code (binary)	Resulting COB-ID
EMERGENCY	0001	81h – FFh
PDO1 (tx)	0011	181h – 1FFh
PDO1 (rx)	0100	201h – 27Fh
PDO2 (tx)	0101	281h – 2FFh
PDO2 (rx)	0110	301h – 37Fh
PDO3 (tx)	0111	381h – 3FFh
PDO3 (rx)	1000	401h – 47Fh
PDO4 (tx)	1001	481h – 4FFh
PDO4 (rx)	1010	501h – 57Fh
SDO (tx)	1011	581h – 5FFh
SDO (rx)	1100	601h – 67Fh
NMT Error Control	1110	701h – 77Fh



„Tx“ = wird vom Slave gesendet  
 „Rx“ = wird vom Slave empfangen

### 6.3.4 Netzmanagement

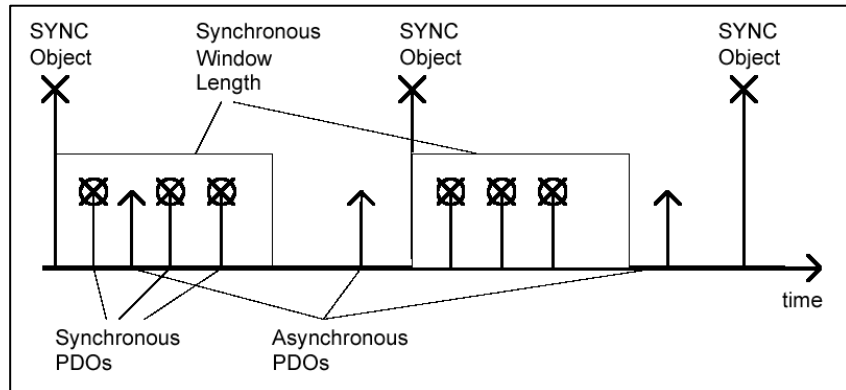


Das SYNC-Telegramm kann mit der CAN 300 Baugruppe über einen Timer realisiert werden.

#### SYNC:

Das SYNC-Telegramm ist ein periodisches "Broadcast"-Telegramm und gibt den Basis-Bustakt vor. Um einen zeitlich äquidistanten Abstand zu ermöglichen, besitzt das SYNC-Telegramm eine hohe Priorität.

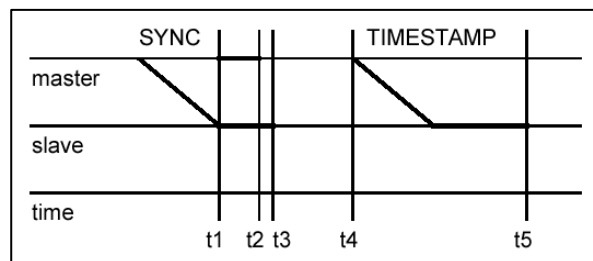
[COB-ID: 80h]



Das Time Stamp-Telegramm kann mit der CAN 300 Baugruppe über einen Timer realisiert werden.

#### Time Stamp:

Das Time Stamp-Telegramm ist ein periodisches "Broadcast"-Telegramm und gibt die Systemzeit vor. Das Time Stamp-Telegramm wird üblicherweise direkt nach einem SYNC-Telegramm übertragen und gibt dann die Systemzeit des SYNC-Telegrammes an.



Um einen zeitlich genaue Übertragung zu ermöglichen, besitzt das Time Stamp-Telegramm eine hohe Priorität.

[COB-ID: 100h]

#### Nodeguarding:

Beim Nodeguarding überwacht der Master die CANopen Slave Baugruppen durch zyklisch gesendete Telegramme an jeden Slave. Auf das Nodeguarding-Telegramm muß jeder CANopen Slave mit einem Status-Telegramm antworten.

Mittels Nodeguarding kann die Steuerung den Ausfall eines CANopen Slaves erkennen.

[COB-ID: 700h + Node-ID]

### *Lifeguarding:*

Beim Lifeguarding überwacht jeder CANopen Slave, ob der Master das einmal gestartete Nodeguarding kontinuierlich innerhalb bestimmter Zeitlimits durchführt.

Wenn das Nodeguarding Telegramm des Masters ausbleibt, kann die dezentrale Peripheriebaugruppe dieses mittels Lifeguarding feststellen und z.B. alle Ausgänge in den sicheren Zustand versetzen.

Nodeguarding und Lifeguarding laufen immer zusammen ab.

### *Heartbeat:*

Die Heartbeat Überwachung entspricht dem Nodeguarding, wobei aber keine Anforderungstelegramme vom CANopen Master generiert werden. Das Heartbeat Telegramm wird vom Node selbsttätig gesendet und kann im Master ausgewertet werden.



*Einige CANopen Slave Baugruppen erzeugen beim Ein- oder Ausschalten spezielle Emergency-Messages.*

### *Emergency-Message:*

Sollte bei einem CANopen Slave ein Störfall auftreten, z.B. der Ablauf des Lifeguarding Timers, so sendet er eine Emergency-Message auf den Bus.

[COB-ID: 80h + Node-ID]

Auf ein Emergency-Telegramm können alle Teilnehmer z.B. einen Notstop ausführen.

### *BootUp-Message:*

CANopen Slaves erzeugen nach dem Einschalten eine BootUp-Meldung, die der Master erkennen kann, um diesen neuen Teilnehmer zu initialisieren.

[COB-ID: 700h + Node-ID + 1 Byte Daten: 00h]



Die CANopen-Hantierungsbausteine sollten nicht mit den Layer 2 – Hantierungsbausteinen gemeinsam aufgerufen werden!

### 6.3.5 Hantierungsbausteine

Die Hantierungsbausteine für die CANopen Kommunikation stellen alle notwendigen Funktionen zur Verfügung um SDOs und PDOs zu verarbeiten und Netzmanagement durchführen zu können.

Die folgende Beschreibung bezieht sich auf die Version 2.7 der Hantierungsbausteine.

Die CAN 300 Baugruppe arbeitet mit diesen Hantierungsbausteinen als *Master* im CANopen Netz.

Baustein	Funktion	User / System	Kapitel
FC 40	Initialisierung (Neustart)	User	6.3.7
FC 41	SDO lesen	User	6.3.10
FC 41	SDO senden	User	6.3.10
FC 42	SDO segmented download	User	6.3.11
FC 42	SDO segmented upload	User	6.3.11
FC 43	Spontanempfang (NMT, PDO)	User	6.3.14
FC 44	PDO senden	User	6.3.12
FC 45	PDO anfordern	User	6.3.13
FC 47	Nodeguarding/Heartbeat	User	6.3.16
FC 48	Netz Management	User	6.3.15
FC 49	Zyklus	System	6.3.9
DB-PDO	Empfangene PDO-Daten	User	<b>Fehler! Verweisquelle konnte nicht gefunden werden.</b>
CAN-DB	Verwaltungs-DB	System	6.3.6

### 6.3.6 CAN-DB

Für jede CAN 300 Baugruppe wird ein CAN-DB benötigt (Länge 300 Byte), der die Verwaltungsinformationen enthält. Der CAN-DB wird durch den FC 40 im Anlauf initialisiert und von allen anderen FCs verwendet.

In diesem Baustein werden die empfangenen und zu sendenden CAN-Telegramme abgelegt, bevor sie weitergereicht werden, und die laufenden Aufträge verwaltet.

### 6.3.7 FC 40 Initialisierung

Der FC 40 muß im Anlauf der SPS aufgerufen werden. Der FC 40 initialisiert den CAN-DB, damit alle anderen CANopen Hantierungsbausteine korrekt arbeiten können.



*Der FC 40 führt keinen Neustart der Baugruppe aus. Er kann also nicht zum Reseten der Baugruppe verwendet werden!*

Parameter	Richtung	Typ	Beispiel
CanDB	IN	BLOCK-DB	DB 40
BaseAddr	IN	INT	256
PDO_DB_1	IN	INT	51
PDO_DB_2	IN	INT	0
PDO_DB_3	IN	INT	0
PDO_DB_4	IN	INT	0

CanDB            interner DB mit aktuellen CAN-Daten

BaseAddr        Basis Adresse der Baugruppe

PDO\_DB\_1..4    Nummer des DBs für den Empfang der PDO 1..4  
Daten aller Nodes

### 6.3.8 PDO-DBs

Die Daten von empfangenen PDO-Telegrammen werden vom FC 49 „Zyklus“ automatisch in DBs kopiert. Hierzu muß für jeden PDO (1-4) jeweils ein DB bei der Initialisierung (siehe 6.3.7) angegeben werden.

Jeder DB enthält für alle 63 Nodes jeweils Platz für 8 Bytes PDO-Daten. Somit muss jeder PDO-DB mindestens 512 Bytes lang sein.

		PDO1 DB	PDO2 DB	PDO3 DB	PDO4 DB
	<b>DBB0</b>	nicht verwendet	nicht verwendet	nicht verwendet	nicht verwendet
	...	nicht verwendet	nicht verwendet	nicht verwendet	nicht verwendet
	<b>DBB7</b>	nicht verwendet	nicht verwendet	nicht verwendet	nicht verwendet
<b>Node 1</b>	<b>DBB8</b>	1. Byte von Node 1 / PDO1	1. Byte von Node 1 / PDO2	1. Byte von Node 1 / PDO3	1. Byte von Node 1 / PDO4
	<b>DBB9</b>	2. Byte von Node 1 / PDO1	2. Byte von Node 1 / PDO2	2. Byte von Node 1 / PDO3	2. Byte von Node 1 / PDO4
	<b>DBB10</b>	3. Byte von Node 1 / PDO1	3. Byte von Node 1 / PDO2	3. Byte von Node 1 / PDO3	3. Byte von Node 1 / PDO4
	<b>DBB11</b>	4. Byte von Node 1 / PDO1	4. Byte von Node 1 / PDO2	4. Byte von Node 1 / PDO3	4. Byte von Node 1 / PDO4
	<b>DBB12</b>	5. Byte von Node 1 / PDO1	5. Byte von Node 1 / PDO2	5. Byte von Node 1 / PDO3	5. Byte von Node 1 / PDO4
	<b>DBB13</b>	8. Byte von Node 1 / PDO1	6. Byte von Node 1 / PDO2	6. Byte von Node 1 / PDO3	6. Byte von Node 1 / PDO4
	<b>DBB14</b>	7. Byte von Node 1 / PDO1	7. Byte von Node 1 / PDO2	7. Byte von Node 1 / PDO3	7. Byte von Node 1 / PDO4
	<b>DBB15</b>	8. Byte von Node 1 / PDO1	8. Byte von Node 1 / PDO2	8. Byte von Node 1 / PDO3	8. Byte von Node 1 / PDO4
<b>Node 2</b>	<b>DBB16</b>	1. Byte von Node 2 / PDO1	1. Byte von Node 2 / PDO2	1. Byte von Node 2 / PDO3	1. Byte von Node 2 / PDO4
	...	...	...	...	...
	<b>DBB23</b>	8. Byte von Node 2 / PDO1	8. Byte von Node 2 / PDO2	8. Byte von Node 2 / PDO3	8. Byte von Node 2 / PDO4
<b>Node 63</b>	<b>DBB504</b>	1. Byte von Node 63 / PDO1	1. Byte von Node 63 / PDO2	1. Byte von Node 63 / PDO3	1. Byte von Node 63 / PDO4
	...	...	...	...	...
	<b>DBB511</b>	8. Byte von Node 63 / PDO1	8. Byte von Node 63 / PDO2	8. Byte von Node 63 / PDO3	8. Byte von Node 63 / PDO4

Die COB-IDs der zutreffenden Telegramme sind fest eingestellt:

PDO1            180h + Node-ID  
 PDO2            280h + Node-ID  
 PDO3            380h + Node-ID  
 PDO4            480h + Node-ID

Sollte für einen PDO (1-4) kein DB angegeben werden so können die Daten auch mit dem FC 43 Spontanempfang (siehe 6.3.14) abgeholt werden.

### 6.3.9 FC 49 Zyklus

Der FC 49 muß im Zyklus des Programmes ausgeführt werden. Er sendet und empfängt die Telegramme von der CAN 300 Baugruppe und ordnet die Daten den jeweiligen Aufträgen zu.

Der FC 49 kopiert auch die PDO-Daten in die PDO Empfangs-DBs (siehe 6.3.7 FC 40 Initialisierung).

Parameter	Richtung	Typ	Beispiel
CanDB	IN	BLOCK-DB	DB 40
T	IN	TIMER	T 49
Bufferinfo	OUT	WORD	MW 140
Stat	OUT	BYTE	MB 142

CanDB interner DB mit aktuellen CAN-Daten, siehe FC 40 Initialisierung

T Timer zur internen Verwendung. Werden mehrere CAN 300-Baugruppen in einem Aufbau eingesetzt, so wird für jede Baugruppe ein Timer benötigt.

Bufferinfo Anzeige der belegten Empfangsfächer, bzw. der laufenden Aufträge (Sendefächer).  
Diese Information sollte vor dem Start neuer Aufträge ausgewertet werden:

*Unteres Byte für Sendepuffer:*

Bit 1 = SDO Sendepuffer belegt. Keinen neuen Auftrag mit FC 41 oder FC 42 starten.

*Oberes Byte für Empfangspuffer:*

Bit 0 = PDO empfangen (COB-IDs 200h-57Fh), wenn kein PDO-DB vorhanden ist.  
=> FC 43 Funktion 0 aufrufen.

Bit 2 = Timestamp empfangen (COB-IDs 100h)  
=> FC 43 Funktion 3 aufrufen

Bit 3 = NMT empfangen (COB-IDs 00h – 7Fh)  
=> FC 43 Funktion 3 aufrufen

Bit 4 = Emergency- (COB-IDs 81h-FFh) oder SYNC-Telegramm (80h) empfangen => FC 43 Funktion 4 aufrufen

Bit 5 = NMT-Error Telegramm empfangen (COB-IDs 700h-77Fh)=> FC 43 Funktion 5 aufrufen

Bit 6 = NMT-Service Telegramm empfangen (COB-IDs 780h-7FFh)=> FC 43 Funktion 1 aufrufen

Stat Status der CAN 300 Baugruppe (siehe Kap. 6.2.6).  
Das Byte wird immer versorgt, auch wenn kein Telegramm empfangen wurde. *[neu ab Version 2.6]*

Der FC 49 sollte bei längeren SPS Zykluszeiten mehrmals in einem Zyklus aufgerufen werden. Jeder Aufruf arbeitet maximal einen Auftrag zum Senden und einen Auftrag zum Empfangen ab, falls ein Auftrag vorliegt.

Werden mehrere CAN 300-Baugruppen in einem Aufbau eingesetzt, so muß für jede Baugruppe der FC 49 mit einem anderen CanDB aufgerufen werden.



### 6.3.10 FC 41 SDO lesen und schreiben

Mit diesem FC können SDOs von einem Slave mit bis zu 4 Datenbytes gelesen und geschrieben werden.

Parameter	Richtung	Typ	Beispiel
CanDB	IN	BLOCK-DB	DB 40
Node	IN	INT	2
Index	IN	WORD	W#16#7300
Subindex	IN	BYTE	B#16#1
Typ	IN	BYTE	B#16#2B
T	IN	TIMER	T 41
ReturnTyp	OUT	BYTE	MB 11
Status	OUT	BYTE	MB 10
Error	OUT	WORD	MW 16
Activate	IN_OUT	BOOL	M 1.2
Data	IN_OUT	DWORD	MD 12
AbortCode	IN_OUT	DWORD	MD 14

CanDB	interner DB mit aktuellen CAN-Daten, siehe FC 40 Initialisierung
Node	Nummer des CAN-Teilnehmers
Index	Index des Objektes
Subindex	Subindex beim Objekt
Typ	Größe & Richtung der Objektdaten: 40h = SDO lesen (8/16/32 Bit), 23h = 32 Bit SDO senden, 2Bh = 16 Bit SDO senden, 2Fh = 8 Bit SDO senden
T	Timer für Timeout, wenn keine Antwort empfangen wird.
ReturnTyp	Lesen eines SDOs: Größe der empfangenen Objektdaten (43h = 32 Bit, 4Bh = 16 Bit, 4Fh = 8 Bit) Schreiben eines SDOs: 60h = OK
Status	Statusbyte der Auftragsbearbeitung: Bit 0 = Auftrag läuft, Bit 5 = Es liegt ein AbortCode vor Bit 6 = Fehler (Fehlernummer in Error) Bit 7 = Auftrag fertig
Error	Fehlernummer bei fehlerhafter Ausführung
Activate	Aktivierungsbit zum Starten des Auftrages, wird nach Eintragen des Auftrages zurückgesetzt
Data	Übergabe der Daten (lesend und schreibend)
AbortCode	Fehlernummer vom CANopen Slave

Der FC muß zyklisch aufgerufen werden. Die SDO Übertragung wird erst ausgelöst, wenn das Aktivierungsbit (Activate) gesetzt wird. Der FC setzt das Bit nach Annahme des Auftrages sofort zurück. Der aktuelle Status der Auftragsbearbeitung kann im Byte Status beobachtet werden.

Der FC trägt den gewünschten Auftrag in den CAN-DB ein. Der Auftrag wird aber erst ausgeführt (auf den CAN-Bus gesendet), wenn der FC 49 aufgerufen wird.

Für die Übertragung von SDOs mit mehr als 4 Bytes muß der FC 42 (siehe 6.3.11) verwendet werden.

*Aufrufbeispiel:*

```
UN   M   9.1           // new job ?
UN   M  111.0         // job running ?
SPB  next

CALL FC 41
CanDB := DB 40
Node  := MW 28
Index := MW 30
Subindex := MB 32
Typ   := B#16#40
T     := T 41
ReturnTyp:= MB 33
Status := MB 111
Error  := MW 112
Activate := M 9.1
Data   := MD 34
AbortCode:= MD 94

UN   M  111.7         // Ready ?
SPB  next
U    M  111.6         // Error ?
SPB  err

L    MD  34           // Get SDO-Value !

// use here SDO Value depending on ReturnTyp (size)
next: ...
```

### 6.3.11 FC 42 SDO Segment download/upload

Mit diesem FC können SDOs von einem Slave mit mehr als 4 Datenbytes gelesen und geschrieben werden. Die Übertragung wird mit mehreren Telegrammen am CAN-Bus durchgeführt („SDO Segmented“).

Parameter	Richtung	Typ	Beispiel
CanDB	IN	BLOCK-DB	DB 40
SDO_DB	IN	INT	151
Timeout	IN	TIMER	T 150
Status	OUT	BYTE	MB 10
Error	OUT	WORD	MW 12
AbortCode	OUT	DWORD	MD 14
Activate	IN_OUT	BOOL	M 1.2

- CanDB           interner DB mit aktuellen CAN-Daten, siehe FC 40 Initialisierung
- SDO\_DB        DB mit den Daten
- Timeout       Timer für Timeoutüberwachung
- Status         Statusbyte der Auftragsbearbeitung:  
                   Bit 0 = Auftrag läuft  
                   Bit 5 = Es liegt ein Abortcode vor  
                   Bit 6 = Fehler (Fehlernummer in Error)  
                   Bit 7 = Auftrag fertig  
                   alle Bits 0 = es läuft kein Auftrag mehr
- Error          Fehlernummer bei fehlerhafter Ausführung
- AbortCode     Fehlernummer vom CANopen Slave
- Activate       Aktivierungsbit zum Starten des Auftrages

Die Informationen der SDO Übertragung müssen in einem DB hinterlegt (BlockDB) werden:



*Es wird pro FC 42 Aufruf nur ein Auftrag aus dem DB ausgeführt. Verkettung von mehreren Aufträgen ist nicht möglich.*

Byte	Typ	Beispiel	Zweck
0	BYTE	1	Richtung: 2=Segment Upload, 3=Segment Download
2	WORD	4	Node
4	WORD	1004h	SDO Index
6	BYTE	1h	SDO Subindex
7	BYTE	32d	Size (Anzahl Bytes)
8	ARRAY 1...n		
	BYTE		Nettodaten
	ENDARRAY		



*Eine Timeoutüberwachung der Aufträge muß von der S7-Applikation durchgeführt werden. Die Antwortzeiten der CANopen-Slaves können sehr unterschiedlich sein.*

Der FC muß zyklisch aufgerufen werden. Die SDO Übertragung wird erst ausgelöst, wenn das Aktivierungsbit (Activate) gesetzt wird. Der FC setzt das Bit nach Annahme des Auftrages sofort zurück. Der aktuelle Status der Auftragsbearbeitung kann im Byte Status beobachtet werden.

Der FC trägt den gewünschten Auftrag in den CAN-DB ein. Der Auftrag wird erst ausgeführt, wenn der FC 49 aufgerufen wird.

### 6.3.12 FC 44 PDO senden

Dieser FC sendet ein PDO mit Daten an einen Slave.

Parameter	Richtung	Typ	Beispiel
CanDB	IN	BLOCK-DB	DB 40
Node	IN	INT	2
PDO	IN	INT	1
Length	IN	INT	4
Data1234	IN	DWORD	W#16#10203040
Data5678	IN	DWORD	W#16#00000000
SYNC	IN	INT	1
Status	OUT	BYTE	MB 10
Error	OUT	WORD	MW 12

CanDB	interner DB mit aktuellen CAN-Daten, siehe FC 40 Initialisierung
Node	Nummer des CAN-Teilnehmers
PDO	Nummer des PDOs
Length	Länge der Telegramm Daten
Data1234	die ersten 4 Bytes Daten (bytweweise links -> rechts)
Data5678	die letzten 4 Bytes Daten (bytweweise links -> rechts)
Sync	PDO-Telegramm wird entweder asynchron versendet (Sync = 1) oder mit einem in der CAN 300 Baugruppe hinterlegten Timer synchron (Sync = 3), d.h. die Daten des entsprechenden Timers werden aktualisiert.
Status	Statusbyte der Auftragsbearbeitung: Bit 6 = Fehler (Fehlernummer in Error) Bit 7 = Auftrag fertig
Error	Fehlernummer bei fehlerhafter Ausführung

Die Datenbytes werden bytweweise von links nach rechts in das PDO-Telegramm übernommen. Sollte z.B. nur ein Byte gesendet werden, so muß es in den obersten 8 Bit des Parameters Data1234 stehen.

Der FC übergibt den PDO-Auftrag sofort an die Baugruppe, ein Aufruf des FC 49 ist nicht nötig. Es können somit mehrere PDO-Aufträge in einem Zyklus hintereinander versendet werden!

### 6.3.13 FC 45 PDO anfordern

Dieser FC fordert ein PDO von einem Slave an. Es wird ein PDO-Telegramm mit gesetztem RTR-Bit gesendet, woraufhin der Slave ein PDO mit seinen aktuellen Daten senden sollte.

Parameter	Richtung	Typ	Beispiel
CanDB	IN	BLOCK-DB	DB 40
Node	IN	INT	2
PDO	IN	INT	1
Status	OUT	BYTE	MB 10
Error	OUT	WORD	MW 12

CanDB	interner DB mit aktuellen CAN-Daten, siehe FC 40 Initialisierung
Node	Nummer des CAN-Teilnehmers
PDO	Nummer des PDOs
Status	Statusbyte der Auftragsbearbeitung: Bit 6 = Fehler (Fehlernummer in Error) Bit 7 = Auftrag fertig
Error	Fehlernummer bei fehlerhafter Ausführung

Die Daten des Antworttelegrammes sind dann im PDO-DB zu finden, oder können mit dem FC 43 (siehe 6.3.14) abgeholt werden.

Der FC übergibt den PDO-Auftrag sofort an die Baugruppe, ein Aufruf des FC 49 ist nicht nötig. Es können somit mehrere PDO-Aufträge in einem Zyklus hintereinander versendet werden!

### 6.3.14 FC 43 Spontanempfang

Mit dem diesem FC können Telegramme abgeholt werden, die ohne zugehörigen Auftrag vom CAN-Bus empfangen werden.

Parameter	Richtung	Typ	Beispiel
CanDB	IN	BLOCK-DB	DB 40
Func	IN	INT	4
Status	OUT	BYTE	MB 10
Error	OUT	WORD	MW 12
Node	OUT	INT	MW 14
PDO	OUT	INT	MW 16
Length	OUT	WORD	MW 18
Data1234	OUT	DWORD	MD 20
Data5678	OUT	DWORD	MD 24

CanDB interner DB mit aktuellen CAN-Daten, siehe FC 40 Initialisierung

Func Abfrage-Funktion:  
 0 = PDO-Telegramme vom Slave, für die kein PDO-DB definiert ist (COB-ID 180h-1FFh, 280h-2FFh, 380h-3FFh, 480h-4FFh)  
 1 = NMT-Service Telegramme (COB-ID 780h-7FDh)  
 2 = TimeStamp-Telegramme (COB-ID 100h-17Fh)  
 3 = NMT-Telegramme (COB-ID 00h-7Fh)  
 4 = Emergency-Telegramme (COB-ID 81h-FFh) und SYNC-Telegramme (80h)  
 5 = NMT-Error-Telegramme (COB-ID 700h-77Fh)

Status Statusbyte der Auftragsbearbeitung:  
 Bit 6 = Fehler (Fehlernummer in Error)  
 Bit 7 = Daten da

Error Fehlernummer bei fehlerhafter Ausführung

Node COB-ID oder Nummer des CAN-Teilnehmers (Funktion 0)

PDO Nummer des PDOs (für Funktion 0)

Length Länge der Telegramm Daten

Data1234 die ersten 4 Bytes Daten (bytweweise links -> rechts)

Data5678 die letzten 4 Bytes Daten (bytweweise links -> rechts)

Liegt ein neues Telegramm vor, wird der FC 43 mit dem Status Bit 7 verlassen, sonst wird ein Fehler ausgegeben. Über den Parameter Bufferinfo im Aufruf des FC 49 Zyklus erhält man die Information, ob ein Empfangsfach belegt ist.

Für jede Telegrammart (siehe Parameter Func) existiert nur ein Empfangsfach im CAN-DB. Deshalb sollte dieser FC gleich nach dem FC 49 aufgerufen werden. Natürlich nur, wenn die Telegramme von Bedeutung für die S7-Applikation sind. Ansonsten können die unwichtigen Telegramme ignoriert oder von vorneherein mit den Akzeptanzmasken der Baugruppe ausgefiltert werden (=> weniger Zykluszeitbelastung).

### 6.3.15 FC 48 Netzmanagement

Zum Senden von Netzmanagement-Telegrammen kann der FC 48 verwendet werden.



*Für die Ausführung von Netzmanagement-funktionen im Anlauf oder Stop der CAN 300 Baugruppe sollten die Skripten verwendet werden (siehe 5.5)!*

Parameter	Richtung	Typ	Beispiel
CanDB	IN	BLOCK-DB	DB 40
Node	IN	INT	2
Func	IN	INT	1
Status	OUT	BYTE	MB 10
Error	OUT	WORD	MW 12

- CanDB      interner DB mit aktuellen CAN-Daten, siehe FC 40 Initialisierung
- Node      Nummer des CAN-Teilnehmers (Node=0 -> alle Nodes) oder Timernummer 1-11 (für Funktionen 10 & 11)
- Func      Netzmanagement Funktion:  
             1 = Start Node  
             2 = Stop Node  
             3 = Disconnect Node  
             4 = Enter Preoperational  
             5 = Reset Node  
             6 = Reset Communication  
             10 = Timer Start (Timernummer in Node)  
             11 = Timer Stop (Timernummer in Node)
- Status    Statusbyte der Auftragsbearbeitung:  
             Bit 6 = Fehler (Fehlernummer in Error)  
             Bit 7 = Auftrag fehlerfrei ausgeführt
- Error     Fehlernummer bei fehlerhafter Ausführung

Der FC übergibt den NMT-Auftrag sofort an die Baugruppe, ein Aufruf des FC 49 ist nicht nötig.

### 6.3.16 FC 47 Nodeguarding/Heartbeat

Zum Senden von Nodeguarding-Telegrammen oder Empfang von Heartbeat-Telegrammen kann der FC 47 verwendet werden.

Parameter	Richtung	Typ	Beispiel
CanDB	IN	BLOCK-DB	DB 40
BeatDB	IN	BLOCK-DB	DB 47
GuardTimer	IN	T	T 47
Error	OUT	WORD	MW 12

- CanDB           interner DB mit aktuellen CAN-Daten, siehe FC 40 Initialisierung
- BeatDB         DB mit Liste der am Netz zu überwachenden Slaves
- GuardTimer    Überwachungstimer für Nodeguarding
- Error          Fehlernummer bei fehlerhafter Ausführung

Für die Überwachung der Slaves mittels Nodeguarding/Heartbeat wird ein DB benötigt, der eine Liste der zu überwachenden Slaves enthält (BeatDB):

Byte	Typ	Beispiel	Zweck
0 + 1	INT	50	Überwachungszeit (in 10ms Schritten)
2	BYTE	-	Zähler (intern verwendet)
3	BYTE	-	Zustand (intern verwendet)
4	BYTE	1	Nodenummer des 1. Slaves
5	BYTE	0	Empfangener Status des 1. Slaves
6	BYTE	2	Nodenummer des 2. Slaves
7	BYTE	0	Empfangener Status des 2. Slaves
8	BYTE	3	Nodenummer des 3. Slaves
9	BYTE	0	Empfangener Status des 3. Slaves
...	...	...	...
X	BYTE	0	Ende der Liste
X+1	BYTE	0	

**i**  
 Der Zustand  $EE_{hex}$  zeigt einen fehlenden Slave an.

Wenn ein Heartbeat-Telegramm empfangen wird, so wird der Status dieses Telegrammes in den BeatDB bei der passenden Nodenummer eingetragen.

Sollte innerhalb der Überwachungszeit kein Statustelegamm empfangen werden, so wird der Status auf  $EE_{hex}$  gestellt. Das oberste Bit in der Nodenummer signalisiert, daß ein Nodeguarding-Telegramm gerade empfangen wurde.

#### Beispiel mit DB47 als BeatDB:

26	//Nodeguarding		
27	M 109.5	BOOL	true
28	DB47.DBW 4	HEX	W#16#8205
29	DB47.DBW 6	HEX	W#16#04EE
30			

Vom Slave mit Nodenummer 2 wurde ein Nodeguarding oder Heartbeat Telegramm mit der Kennung 05 (Operational) empfangen. Vom Slave 4 wurde kein Telegramm empfangen, der Zustand 0xEE zeigt an, daß der Teilnehmer sich nicht gemeldet hat.

Slave Zustände: 4 = Stop, 5 = Operational, 127 = Preoperational



## 6.4 Erläuterung zum CANopen Beispielprogramm

Das mit der CANopen Handlung mitgelieferte Beispielprogramm zeigt die Verwendung der Handlungsbauklne in einer sehr einfachen Form. Die Funktionalitäten der Handlungsbauklne werden durch die Bits des Eingangsbyte 8 ausgelöst.

Verwendet wurde ein einfacher CANopen IO-Slave mit 8 Ausgängen und 8 Eingängen als Node 7. Die Eingänge sind mit den Ausgängen für diesen Test direkt verdrahtet worden.

Die CAN 300 Baugruppe muß auf der Adresse 256 konfiguriert sein. Im OB100 wird die Handlung initialisiert, wobei der DB40 als CAN-DB, der DB51 als PDO1-Daten-DB und der DB52 als PDO1-Daten-DB verwendet wird.

Das Beispielprojekt „C3 CANopen Master.PAR“ (wird mit CANParam mit installiert) sollte auf das Testgerät angepaßt werden (z.B. Baudrate des CAN-Bus) und in die CAN 300 Baugruppe eingespielt worden sein.

### 6.4.1 Beispiel FC 10 (Zyklus/SDO/PDO/Netzmanagement)

Am Anfang des FC 10 wird er Zyklusbaustein FC 49 (→ Kapitel 6.3.9) aufgerufen, um über den CAN-Bus empfangene Telegramme zu holen, oder Sendeaufträge abzuarbeiten. Die Bufferinfo wird in MW 10 abgelegt und auf dem AW 0 angezeigt.

Im zweiten Netzwerk kann mit dem Eingangsbit 8.7 kann das Nodeguarding über den FC 47 (→ 6.3.16) aktiviert werden. Der DB47 enthält eine Liste mit den Nodenummern 1, 2, 3 und 7, die zyklisch abgefragt werden.

Im Netzwerk 3 wird als erstes aus dem PDO1-DB die ersten beiden Bytes des PDO1 vom Node 7 geladen und auf dem AW 2 ausgegeben.

Mit dem Eingangsbit 8.0 kann hier nun ein zyklisches Senden des PDO1 angestoßen werden (FC 44, → 6.3.12). Der Wert wird immer um 1 erhöht und gesendet, wenn der letzte Wert vom Teilnehmer über das Empfangs PDO1 zurückgekommen ist. Entfernen Sie die Vergleichsabfrage in diesen Zeilen, falls Sie in jedem Zyklus senden wollen. Die Daten stehen im MB12 - MB19, wobei nur das MW12 hochgezählt wird.

Netzwerk 4 enthält das Holen und Senden eines SDOs (FC 41, → 6.3.10) über die Eingangsbits 8.1 und 8.2. Die Übergabeparameter sind auf MW gelegt worden. Eine Variablen-tabelle (VAT\_1) zum Testen ist im Projekt hinterlegt.

Im Netzwerk 5 wird der FC 48 (→ 6.3.15) für das Netzmanagement aufgerufen. Beachten Sie bitte, daß im Beispielprojekt „C3 CANopen Master.PAR“ in den Skripten für Start und Stop den CPU bereits die CAN-Telegramme für „NMT start all nodes“ und „NMT stop all nodes“ hinterlegt sind.

## 6.5 Fehlernummern

### 6.5.1 Hantierungs-Baustein Fehler

Mögliche Fehlernummern des Rückgabeparameters Error.

Nummer	Bedeutung
1	Node unter 1
2	Node über 63
3	PDO unter 1, bzw. Timernummer unter 0
4	PDO über 4, bzw. Timernummer über 11
6	Keine Daten vorhanden
11	Node unter 0
12	EndSegmentMode bei UP gibts nicht
15	Anstoß noch da
16	keine Antwort erwartet
17	Node falsch
18	Index falsch
19	Subindex falsch
22	ComSpec falsch bei DN
23	ComSpec falsch bei DN
24	ComSpec falsch
25	ComSpec falsch bei DN
26	ComSpec falsch bei SDO-Write
27	ComSpec falsch bei SDO-Read
31	BlockSize falsch bei DN
32	DB-Block zu klein
33	DB-Block undefiniert
35	DB-Block zu klein
36	DB-Block schreibgeschützt
80	Togglebit steht falsch
90	ComSpec falsch bei UP
91	Expidited falsch bei UP
92	ComSpec falsch bei UP
93	ComSpec falsch bei UP
94	Segmentnummer falsch bei UP
94	Segmentnr falsch bei DN
99	Timeout bei SDO-Auftrag, keine Antwort vom CANopen Slave
101	Puffer belegt, mit einem Auftrag beschäftigt
102	Aboartcode empfangen
105	Funktionsnummer unbekannt
140	Baugruppe nicht bereit
141	Baugruppe kann nicht senden (Pufferüberlauf ?)
142	Systemfehler
254	System Fehler Node-Scan
255	Functionscode undefiniert



Bitte beachten Sie auch die bei den Hantierungsbausteinen direkt angegebenen Fehlernummern!

### 6.5.2 Abortcodes

Im folgenden finden Sie typische Fehlermeldungen, die von einem CANopen Slave generiert werden können.

Diese Fehlermeldungen erhalten Sie, wenn Sie eine SDO-Übertragung angefordert haben (FC 41, FC 42).

Code	Bedeutung
0503 0000h	"Toggle Bit "wurde nicht alterniert
0504 0000h	SDO Protokoll "time out "
0504 0001h	Client/Server Kommando Bezeichner nicht gültig oder unbekannt
0504 0002h	Unbekannte Blockgröße (block mode only)
0504 0003h	Unbekannte Blocknummer (block mode only)
0504 0004h	CRC Fehler (block mode only)
0504 0005h	Außerhalb des Speichers
0601 0000h	Zugriff auf dieses Objekt wird nicht unterstützt
0601 0001h	Versuchter Lesezugriff auf ein Objekt, das nur geschrieben werden kann
0601 0002h	Versuchter Schreibzugriff auf ein Objekt, das nur gelesen werden kann
0602 0000h	Objekt existiert im Objektverzeichnis nicht
0604 0041h	Objekt kann nicht in ein PDO "gemappt "werden
0604 0042h	Größe und Anzahl der "gemappten "Objekte übersteigt die mögliche PDO Länge
0604 0043h	Allgemeine Parameter -Inkompatibilität
0604 0047h	Allgemeine Inkompatibilität im Gerät
0606 0000h	Zugriffsverletzung aufgrund eines Hardwarefehlers
0607 0010h	Datentyp paßt nicht, Länge des Service Parameters paßt nicht
0607 0012h	Datentyp paßt nicht, Länge des Service Parameters zu groß
0607 0013h	Datentyp paßt nicht, Länge des Service Parameters zu klein
0609 0011h	Subindex existiert nicht
0609 0030h	Wertebereich des Parameters verlassen (nur für Schreibzugriffe)
0609 0031h	Wert des Parameters zu groß
0609 0032h	Wert des Parameters zu klein
0609 0036h	Maximaler Wert ist kleiner als minimaler Wert
0800 0000h	Allgemeiner Fehler
0800 0020h	Datum kann nicht übertragen oder gespeichert werden
0800 0021h	Datum kann aufgrund lokaler Geräteführung nicht übertragen /gespeichert werden
0800 0022h	Datum kann aufgrund des Gerätezustandes nicht übertragen /gespeichert werden
0800 0023 h	Dynamisches generieren des Objektverzeichnisses nicht möglich oder bereits vorhanden

## 6.6 SAE J1939 Kommunikation

### 6.6.1 Allgemein

Mit den SAE J1939 Hantierungsbausteinen ist es möglich CAN-Telegramme nach dem SAE J1939 Protokoll zu senden und zu empfangen. Das SAE J1939 Protokoll verwendet immer CAN-Identifizier mit 29Bit. Es ist darauf zu achten, daß ein entsprechendes Projekt mit der Einstellung „29 Bit Mode“ in die CAN 300 Baugruppe eingespielt wurde.

Folgende FCs stehen in für SAE J1939 zur Verfügung:

**FC 67 CANSEND\_SAE\_J1939** Senden eines Telegrammes

**FC 68 CANRCV\_SAE\_J1939** Empfangen eines Telegrammes

Eine Initialisierung der Baugruppe in den Anlauf-OB's ist nicht erforderlich. Die Baugruppe startet automatisch, wenn die SPS in RUN geschaltet wird und stoppt, wenn die SPS in STOP geht.

Multipacket messages werden z.Zt. noch nicht unterstützt.



*Für die SAE\_J1939  
Hantierung muß die  
CAN 300 Baugruppe  
auf „29 Bit Mode“  
eingestellt sein!*

### 6.6.2 FC 67 CANSEND\_SAE\_J1939

Der Funktionsbaustein CANSEND\_SAE\_J1939 (FC 67) überträgt ein CAN-Telegramm zur Baugruppe, von der dieses sofort gesendet wird.

Parameter	Richtung	Typ	Beispiel
Base	IN	INT	256
Priority	IN	INT	7
DataPageBit	IN	BOOL	FALSE
PDU_Format	IN	INT	123
PDU_Specific	IN	INT	2
Source_Addr	IN	INT	1
B0	IN	BYTE	B#16#0
B1	IN	BYTE	B#16#0
B2	IN	BYTE	B#16#0
B3	IN	BYTE	B#16#0
B4	IN	BYTE	B#16#0
B5	IN	BYTE	B#16#0
B6	IN	BYTE	B#16#0
B7	IN	BYTE	B#16#0
STAT	OUT	BYTE	MB 80
Snd	IN_OUT	BIT	M 87.0

Als Übergabeparameter muss die Basisadresse der Baugruppe als Integerzahl (Base), ein Statusbyte (STAT) und ein Bit zur Sendefreigabe (Snd) übergeben werden.

Das Bit Snd wird nach dem Durchlauf des Bausteines immer zurückgesetzt, das zu sendende Telegramm wird immer in die Baugruppe übertragen. Sollte der Sendepuffer in der Baugruppe voll sein, so werden ältere, nicht gesendete Telegramme gelöscht. Um dieses zu verhindern, sollte das Bit 4 des STAT Bytes immer vor dem Senden abgefragt werden.

Im Byte STAT steht der Status der CAN 300 Baugruppe (siehe Kap. 6.2.6). Das Byte wird immer versorgt, auch wenn das Snd Bit nicht gesetzt ist. Es empfiehlt sich den Baustein CANRCV\_SAE\_J1939 immer vor dem Baustein CANSEND\_SAE\_J1939 aufzurufen, damit die Statusinformationen aktuell vorliegen.

Sollte der Timer 0 als Synchron-Timer eingestellt worden sein, so werden die Daten immer nur in dem definierten Synchron-Zeitfenster gesendet.

Zuordnung der Parameter zum 29Bit CAN-Identifizier nach dem SAE J1939 Protokoll:



Der FC60 „CANSEND“ darf nicht im OB 1 (Zyklus) und OB 35 (Zeit-OBs) gleichzeitig bzw. gemischt aufgerufen werden!

28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Priority	R	P	PGN (Parameter Group Number)														SA (Source Address)											
	R	P	PDU-Format										PDU-Specific															
	R	P	0...239: PDU1 Format										DA (Destination Address)															
	R	P	240..255: PDU2 Format										Group Extension															

### 6.6.3 FC 68 CANRCV\_SAE\_J1939

Der Funktionsbaustein CANRCV\_SAE\_J1939 (FC 68) überträgt ein CAN-Telegramm von der Baugruppe in die SPS, wenn ein Telegramm empfangen worden ist.

Parameter	Richtung	Typ	Beispiel
Base	IN	INT	256
Priority	IN	INT	MW60
DataPageBit	IN	BOOL	MB68.0
PDU_Format	IN	INT	MW62
PDU_Specific	IN	INT	MW64
Source_Addr	IN	INT	MW66
B0	IN	BYTE	MB70
B1	IN	BYTE	MB71
B2	IN	BYTE	MB72
B3	IN	BYTE	MB73
B4	IN	BYTE	MB74
B5	IN	BYTE	MB75
B6	IN	BYTE	MB76
B7	IN	BYTE	MB77
STAT	OUT	BYTE	MB 78
Recd	IN_OUT	BIT	M 79.0

Als Übergabeparameter muss die Basisadresse der Baugruppe als Integerzahl (*Base*) übergeben werden.

Hat der Funktionsbaustein ein Telegramm von der CAN 300 Baugruppe gelesen, so wird das Bit *Recd* gesetzt.

Im Byte *STAT* steht der Status der CAN 300 Baugruppe (siehe Kap. 6.2.6). Das Byte wird immer versorgt, auch wenn kein Telegramm empfangen wurde. Es empfiehlt sich den Baustein CANRCV\_SAE\_J1939 immer vor dem Baustein CANSEND\_SAE\_J1939 aufzurufen, damit die Statusinformationen aktuell vorliegen.

Zuordnung der Parameter zum 29Bit CAN-Identifizier nach dem SAE J1939 Protokoll:

28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Priority	R	P	<i>PGN (Parameter Group Number)</i>														SA (Source Address)											
	R	P	<i>PDU-Format</i>							<i>PDU-Specific</i>																		
	R	P	<i>0...239: PDU1 Format</i>							<i>DA (Destination Address)</i>																		
	R	P	<i>240..255: PDU2 Format</i>							<i>Group Extension</i>																		

## 7 Anhang

### 7.1 Technische Daten

**Bestellnummer** CAN 300 700-600-CAN01  
CAN 300 (DNV) 700-600-CAN81

**Abmessungen** 116 x 40 x 125 mm (LxBxH)

**Gewicht** ca. 280g

#### CAN-Schnittstelle

**Typ:** ISO/DIN 11898-2,  
CAN High Speed physical Layer  
**Übertragungsrate:** 10 kbit/s bis 1Mbit/s  
**Protokolle:** CAN 2.0A (11Bit)  
CAN 2.0B (29Bit)  
SAE J1939  
CANopen Master  
CANopen Slave *auf Anfrage*  
DEVICENET-Slave *in Vorbereitung*

**Anschluss:** Stecker, SUB-D 9-polig

#### Projektierungsschnittstelle

**Typ:** RS232, seriell asynchron  
**Übertragungsrate:** 9.6 kbit/s  
**Format:** 8/N/1  
**Anschluss:** Stecker, SUB-D 9-polig

#### Versorgung

**Spannung:** DC +5V über Rückwandbus  
**Stromaufnahme:** 160mA (typ.) / 190mA (max.)

#### Besonderheiten

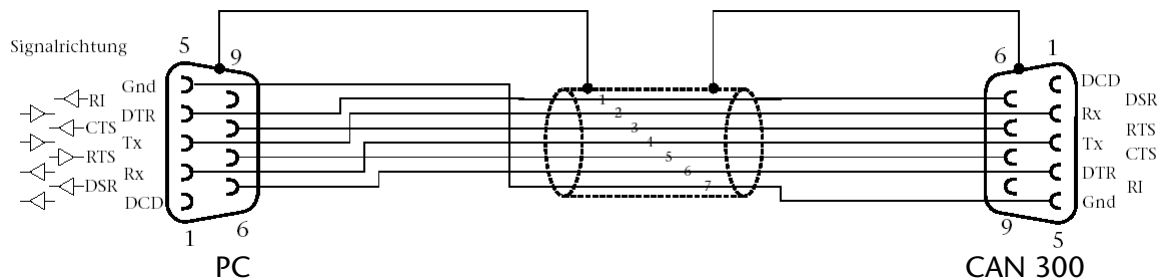
**Qualitätssicherung:** nach ISO 9001:2000  
**Wartung:** Wartungsfrei (keine Batterie oder Akku)

## 7.2 Steckerbelegung

Pin	SubD-Stecker RS232	SubD-Stecker CAN
1	-	-
2	Rx	CAN Low
3	Tx	CAN GND
4	-	-
5	GND	-
6	-	-
7	RTS	CAN High
8	CTS	-
9	-	-

## 7.3 Verbindungskabel

RS232 Parametrierung (700-610-0VK11) / Nullmodem:



## 7.4 Weiterführende Dokumentation

Internet: [www.can-cia.org](http://www.can-cia.org)

CAN Specification 2.0, Part A & Part B

High Layer Protocol CANopen

Holger Zeltwanger: "CANopen", VDE Verlag, ISBN 3-8007-2448-0