



# **CAN-CBM-BRIDGE**

**Intelligente CAN-Bridge**

# **CAN-CBM-PLC/331-2**

**Automationsrechner mit  
2 CAN-Schnittstellen**

**Hardware - Handbuch**



Der Inhalt dieses Handbuches wurde mit größter Sorgfalt erarbeitet und geprüft. **esd** übernimmt jedoch keine Verantwortung für Schäden, die aus Fehlern in der Dokumentation resultieren könnten. Insbesondere Beschreibungen und technische Daten sind keine zugesicherten Eigenschaften im rechtlichen Sinne.

**esd** hat das Recht, Änderungen am beschriebenen Produkt oder an der Dokumentation ohne vorherige Ankündigung vorzunehmen, wenn sie aus Gründen der Zuverlässigkeit oder Qualitätssicherung vorgenommen werden oder dem technischen Fortschritt dienen.

Sämtliche Rechte an der Dokumentation liegen bei **esd**. Die Weitergabe an Dritte und Vervielfältigung jeder Art, auch auszugsweise, sind nur mit schriftlicher Genehmigung durch **esd** gestattet.

**esd electronic system design gmbh**

Vahrenwalder Str. 207

30165 Hannover

Tel.: 0511/372 98-0

FAX : 0511/372 98-68

E-Mail: [info@esd.electronics.com](mailto:info@esd.electronics.com)

Internet: [www.esd-electronics.com](http://www.esd-electronics.com)

<b>Dokument-Datei:</b>	I:\Texte\Doku\MANUALS\CAN\CBM\CAN2\CBMC2_24H.ma9
<b>Datum des Ausdrucks:</b>	2008-01-07

<b>Platinenversionen:</b>	CPU331 Rev. 1.1 CBM-CAN2 Rev. 1.1 SIO3311 Rev. 1.1
---------------------------	--

### Änderungen in den Kapiteln

Die hier aufgeführten Änderungen im Anwenderhandbuch betreffen sowohl Änderungen in der Hardware als auch reine Änderungen in der Beschreibung der Sachverhalte.

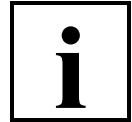
Kapitel	Änderungen gegenüber Vorversion
-	Redaktionelle Überarbeitung (Formatierung für Ausdruck als PDF-Datei)

Weitere technische Änderungen vorbehalten.

Diese Seite ist bewußt unbedruckt.

<b>1. Übersicht</b>	3
1.1 Beschreibung des Moduls	3
1.2 Frontansicht mit Steckern und Kodierschaltern	4
1.3 Zusammenfassung der technischen Daten	5
1.3.1 Allgemeine technische Daten	5
1.3.2 Microcontroller-Baugruppe	6
1.3.3 CAN/DeviceNet-Schnittstelle	6
1.3.4 Serielle Schnittstelle	7
1.4 Bestellhinweise	9
<b>2. Beschreibung der Baugruppen</b>	11
2.1 CAN/DeviceNet-Baugruppe	11
2.1.1 Interface-Schaltung	11
2.2 Serielle Schnittstelle	13
2.2.1 Grundeinstellung der CAN-CBM-Module	13
2.2.2 Konfiguration	13
2.2.3 Anschluss der RS-232-Schnittstelle	14
2.3 Funktion der Kodierschalter	14
2.4 LED-Anzeige	14
<b>3. Steckerbelegungen</b>	15
3.1 CAN-Bus (X400, Combicon-Style)	15
3.2 DeviceNet (X400, Combicon-Style)	16
3.3 RS-232-Interface (X100, 9-pol. DSUB Stift)	17
3.4 Spannungszuführung (X101, UEGM)	18
<b>4. Konfiguration des CAN-CBM-Bridge-Moduls</b>	19
4.1 Bridge mit 11-Bit CAN-Identifizier (C.2842.03)	19
4.1.1 Befehlssatz	19
4.1.2 Konfigurationsbeispiel	25
4.2 Bridge mit 29-Bit CAN-Identifizier (C.2842.05)	26
4.2.1 Befehlssatz	26
4.2.2 Funktion der LEDs	31
<b>5. Konfiguration des CAN-CBM-PLC/331-1/-2 Moduls</b>	33
<b>6. Korrekte Verdrahtung galvanisch getrennter CAN-Netze</b>	43
<b>7. CAN-Bus Troubleshooting Guide</b>	47
7.1 Bus-Abschluss	47
7.2 CAN_H/CAN_L-Spannungen	48
7.3 Erdung	48
7.4 CAN Transceiver-Widerstands-Test	49

Diese Seite ist bewußt unbedruckt.



# 1. Übersicht

## 1.1 Beschreibung des Moduls

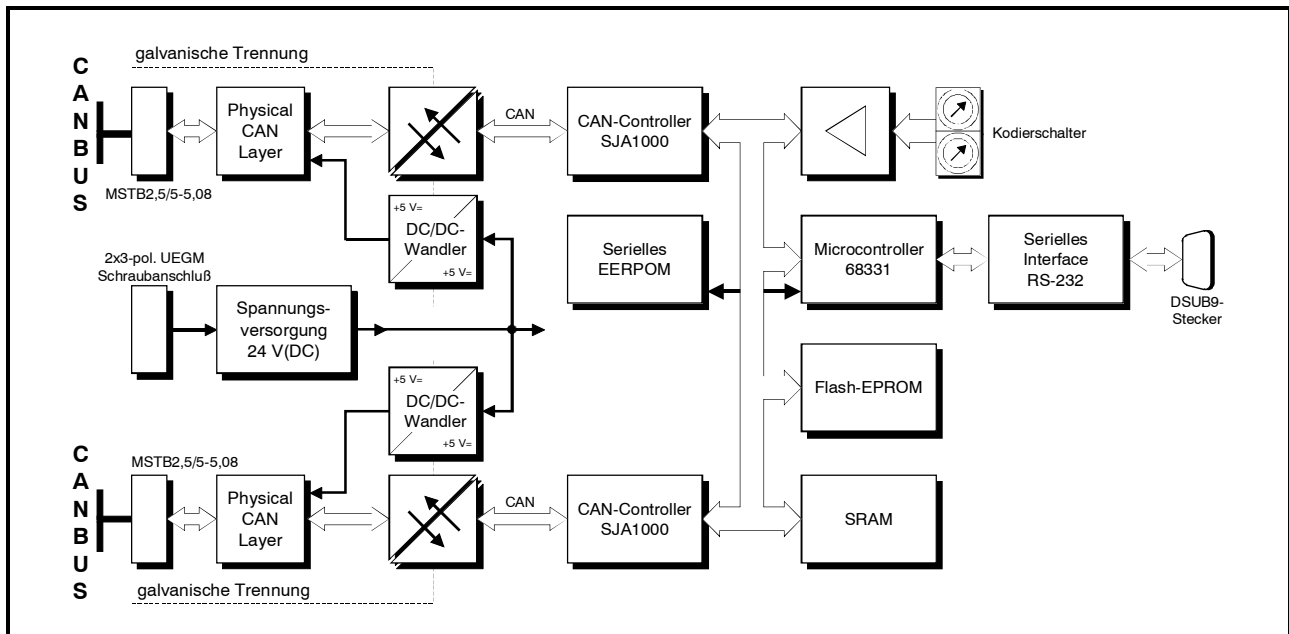


Abb. 1.1: Blockschaubild der CAN-CBM-Module

Mit dem Modul **CAN-CBM-BRIDGE** können zwei eigenständige CAN-Netze miteinander verbunden werden. Die Netze können mit unterschiedlichen Bitraten betrieben werden.

Das Modul **CAN-CBM-PLC/331-2** ist als SPS-Controller mit dem Software-Tool CoDeSys konfiguriert. Es ist mit zwei eigenständigen CAN-Netzen zum Anschluß von CAN-I/O-Modulen ausgestattet.

Beide **CAN-CBM-Module** arbeiten mit einem 68331 Microcontroller, der die CAN-Daten in einem lokalen SRAM zwischenspeichert. Die Firmware wird im Flash gehalten. Zur Speicherung von Parametern dient ein serielles EEPROM.

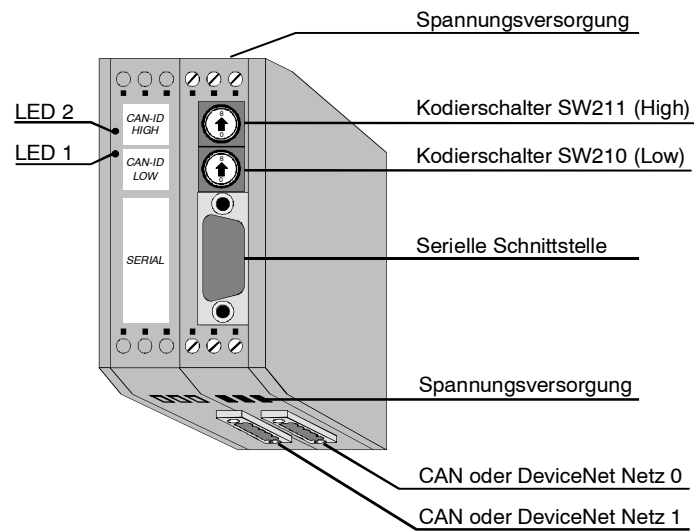
Die zu ISO 11898 kompatiblen CAN-Schnittstellen gestatten jeweils eine maximale Datenübertragungsrate von 1 MBit/s. Die CAN-Interfaces sind durch Optokoppler und DC/DC-Wandler galvanisch getrennt. Der Anschluß erfolgt über 5-polige Schraub/ Steckverbinder in Combicon-Bauweise. Optional ist das Modul mit DeviceNet-Interfaces lieferbar.

Für Service und Entwicklung ist das Modul mit einer seriellen Schnittstelle ausgerüstet. Als physikalische Schnittstelle steht RS-232 zur Verfügung. Der Anschluss erfolgt über einen DSUB9-Stecker.

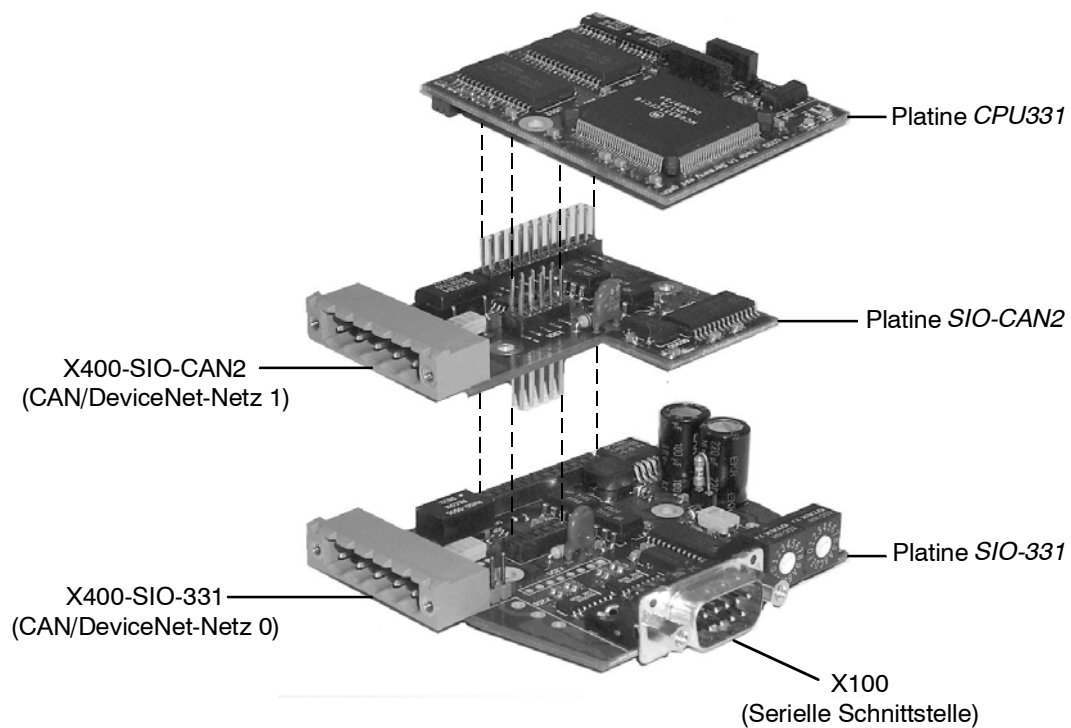
## 1.2 Frontansicht mit Steckern und Kodierschaltern



## Übersicht

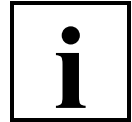


**Abb. 1.2.1:** Position der Steckverbinder und Bedienelemente



**Abb. 1.2.2:** Interner Aufbau des Moduls mit Platinenbezeichnung und Steckernamen (Abbildung ohne Stecker für Versorgungsspannung)





## 1.3 Zusammenfassung der technischen Daten

### 1.3.1 Allgemeine technische Daten

Versorgungsspannung	Nennspannung 24 V/DC $\pm 10\%$ , Stromaufnahme (bei 20 °C): typ. 85 mA
Steckverbinder	X100 (DSUB9, Stifte) - serielle Schnittstelle X101 (6-pol. Schraubverbinder UEGM) - 24V-Spannungsversorgung X400-SIO331 (Combicon-Bauform, 5-pol. MSTB2.5/5-5.08) - CAN oder DeviceNet Netz 0 X400-SIO-CAN2 (Combicon-Bauform, 5-pol. MSTB2.5/5-5.08) - CAN oder DeviceNet Netz 1
Temperaturbereich	0...50 °C Umgebungstemperatur
Luftfeuchtigkeit	max. 90%, nicht kondensierend
Maße der Gehäuses (B x H x T)	Breite: 40 mm, Höhe: 85 mm, Tiefe: 83 mm (einschließlich Hutschienehalterung und Steckerüberstand DSUB9, ohne CAN/DeviceNet-Stecker)
Gewicht	ca. 200 g

**Tabelle 1.3.1:** Allgemeine Daten der CAN-CBM-Module



## Übersicht

### 1.3.2 Microcontroller-Baugruppe

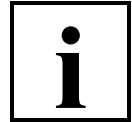
Microcontroller	68331
Speicher	SRAM: 128 k x 16 Bit (optional 512 k x 16 Bit) Flash-EPROM: 128 k x 8 Bit EEPROM: serielles I <sup>2</sup> C-EEPROM
Debug-Interface	für Service und Programmierung

**Tabelle 1.3.2:** Microcontroller-Baugruppe

### 1.3.3 CAN/DeviceNet-Schnittstelle

Anzahl der CAN-Interfaces	2x CAN Option: alternativ 2x DeviceNet
CAN-Controller	SJA1000, CAN 2.0A/B
Galvanische Trennung des CAN-Interfaces gegenüber den anderen Baugruppen	über Optokoppler und DC/DC-Wandler
Physical Layer CAN	Physical Layer gemäß ISO 11898, Übertragungsrate programmierbar von 10 kBit/s bis 1 MBit/s
Physical Layer DeviceNet (Option)	Physical Layer gemäß DeviceNet-Spezifikation Rev. 2.0, Bitrate: 125 kBit/s, 250 kBit/s, 500 kBit/s

**Tabelle 1.3.3:** Daten der CAN-Schnittstelle



### 1.3.4 Serielle Schnittstelle

Controller	68331
Interface	Standard: RS-232
Anschluss	9-pol. DSUB-Stecker

**Tabelle 1.3.4:** Daten der seriellen Schnittstelle

Diese Seite ist bewußt unbedruckt.



## 1.4 Bestellhinweise

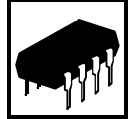
Typ	Eigenschaften	Bestell-Nr.
CAN-CBM-BRIDGE	CAN-Bridge mit 11-Bit-Identifizier, 2x CAN 2.0A	C.2842.03
CAN-CBM-BRIDGE	CAN-Bridge mit 29-Bit-Identifizier, 2x CAN 2.0A/B	C.2842.05
CAN-CBM-PLC/331-2	Automationsrechner mit 2x CAN 2.0A/B	C.2845.04
CAN-CBM-BRIDGE-MD	Anwenderhandbuch in deutsch zu C.2842.03 und C.2842.05 <sup>1*)</sup>	C.2842.20
CAN-CBM-PLC/331-2-MD	Anwenderhandbuch in deutsch zu C.2845.04 <sup>1*)</sup>	C.2845.20
CAN-CBM-BRIDGE-ENG	Engineering Manual in englisch <sup>2*)</sup> Inhalt: Schaltpläne, Bauteilpositionen, Datenblätter wichtiger Bauteile	C.2842.25

1 \*)... Werden diese Handbücher gemeinsam mit dem Produkt bestellt, so werden sie kostenlos mitgeliefert.

2 \*)... Für dieses Handbuch wird eine Schutzgebühr erhoben.

**Tabelle 1.4.1:** Bestellhinweise

Diese Seite ist bewußt unbedruckt.



## 2. Beschreibung der Baugruppen

### 2.1 CAN/DeviceNet-Baugruppe

#### 2.1.1 Interface-Schaltung

Das CAN-CBM-BRIDGE-Modul und das CAN-CBM-PLC/331-2-Modul sind mit zwei CAN-Interfaces gemäß ISO11898 oder zwei DeviceNet-Interfaces lieferbar. Für beide Ausführungen wird der gleiche Steckverbinder in Combicon-Bauweise verwendet. Die Belegung des Steckers ist jedoch unterschiedlich. Die folgenden Abbildungen zeigen die beiden Interfaces.

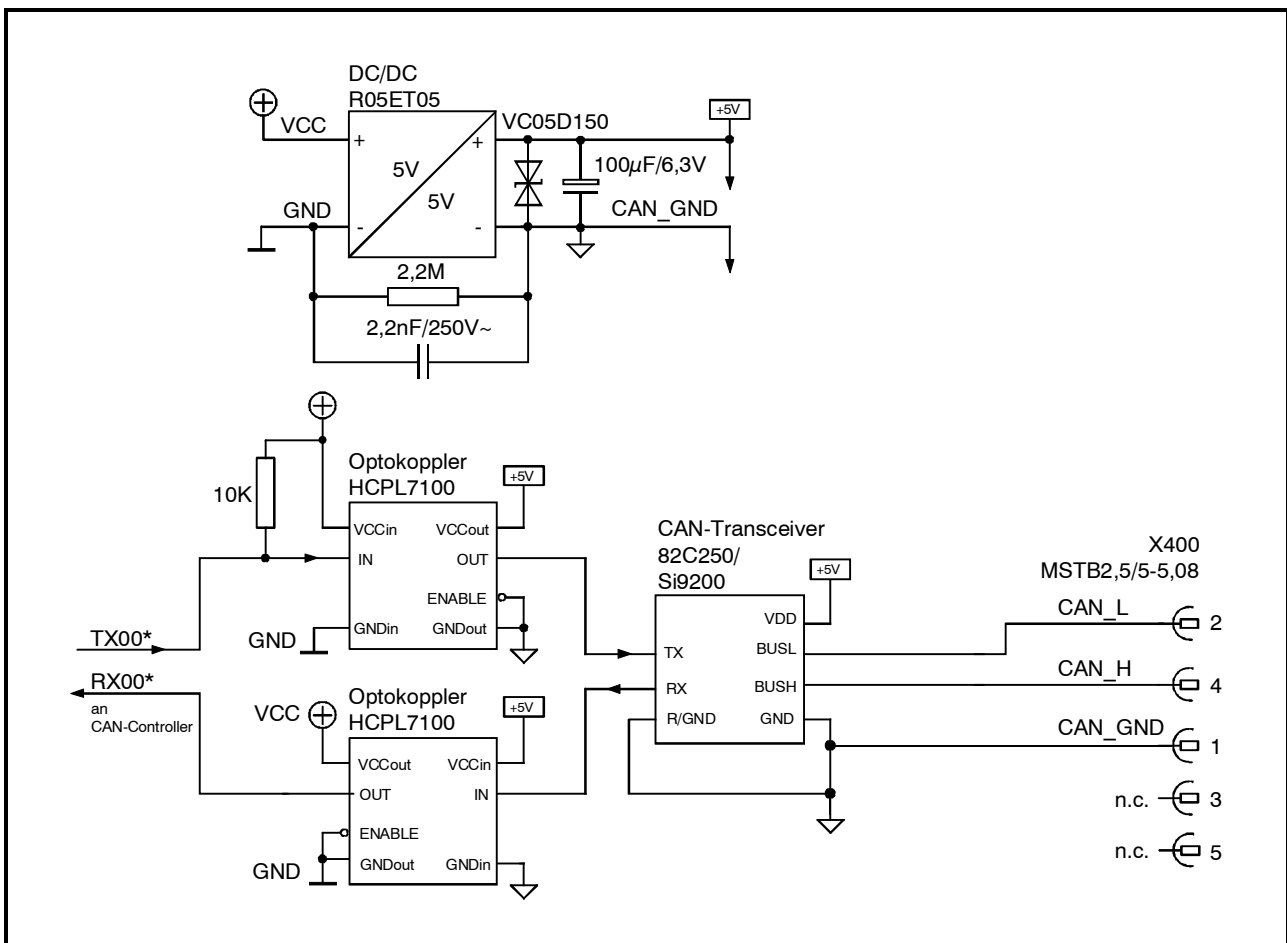


Abb. 2.1.1: Schaltung des CAN-Interfaces



## Beschreibung der Baugruppen

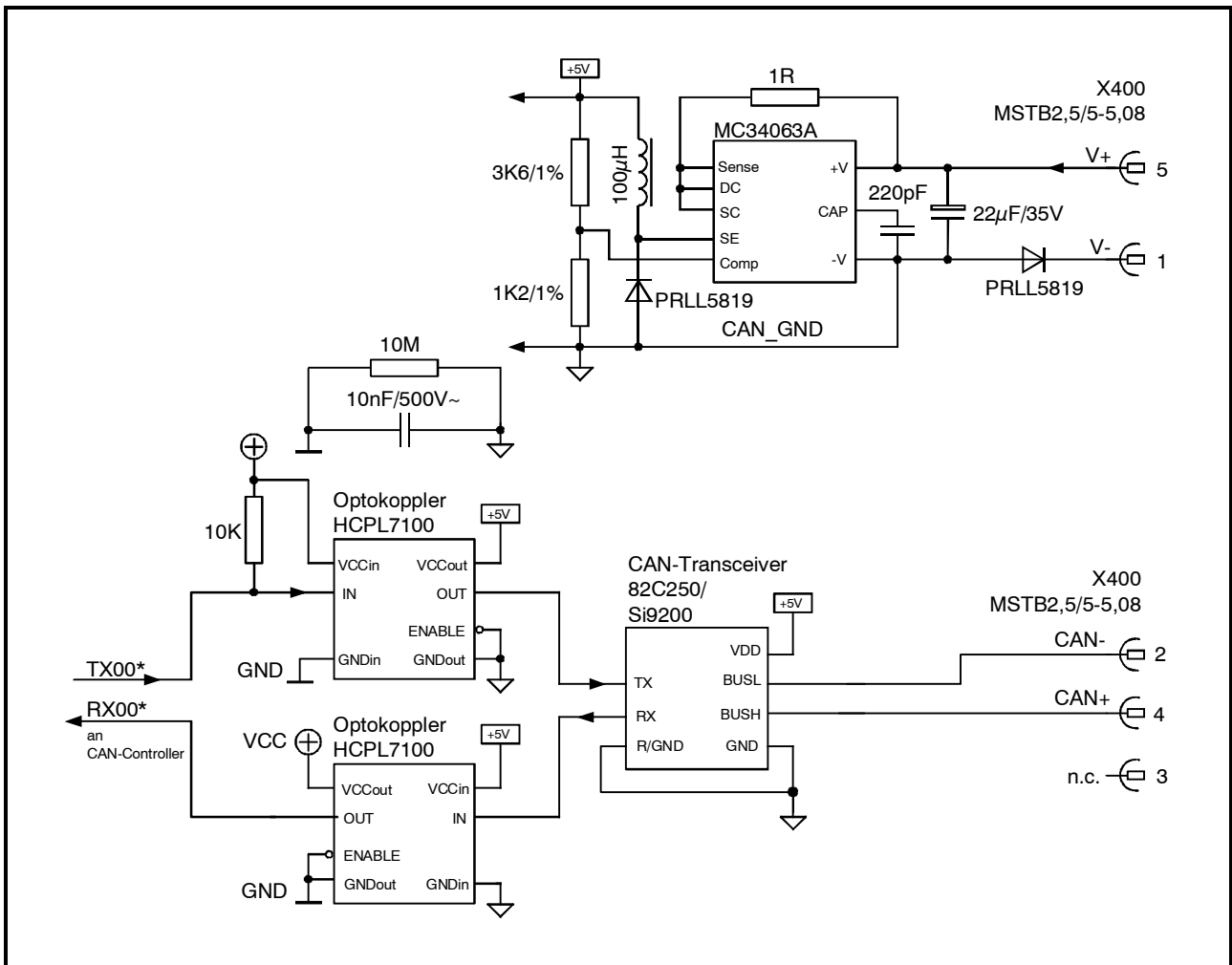
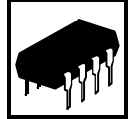


Abb. 2.1.2: Schaltung des DeviceNet-Interfaces





## 2.2 Serielle Schnittstelle

### 2.2.1 Grundeinstellung der CAN-CBM-Module

Bitrate: 19200 Baud  
 Daten-Bits: 8  
 Parity: no  
 Stop-Bits: 1  
 Handshake: XON/XOFF

### 2.2.2 Konfiguration

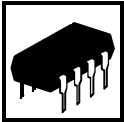
Die serielle Schnittstelle wird vom Microcontroller 68331 gesteuert. Die Bitrate der Schnittstelle ist parametrierbar. Der im 68331 integrierte serielle Controller unterstützt in dieser Applikation Bitraten bis zu 500 kBit/s. Mit den eingesetzten RS-232-Treibern sind maximal 38,4 kBit/s möglich.

Die folgenden Bitraten sind mit Hilfe der Software einstellbar. Die Werte der zweiten Spalte zeigen die tatsächlichen Bitraten, die sich durch die Controller-interne Umrechnung ergeben.

Bitrate (Sollwert) [Bit/s]	Bitrate (Istwert) [Bit/s]
(500.000) <sup>1*)</sup>	(500.000) <sup>1*)</sup>
38.400	38.462
<b>19.200</b>	19.231
9.600	9.615
4.800	4.808
2.400	2.404
1.200	1.199
600	600,2
300	299,9

<sup>1\*)</sup> Diese Bitrate kann nicht mit der RS-232-Schnittstelle umgesetzt werden.

**Tabelle 2.2.:** Einstellbare Bitraten



## Beschreibung der Baugruppen

### 2.2.3 Anschluss der RS-232-Schnittstelle

Die Abbildung soll die im Anhang (Steckerbelegung) verwendeten Kurzbezeichnungen der Signale erläutern. Die Signalbezeichnung ist beispielhaft für den Anschluss der CAN-CBM-BRIDGE und CAN-CBM-PLC/331-2 als Sender (Terminal DEE) angegeben.

**Hinweis:** Für den Anschluss des CAN-CBM-Moduls an die RS-232-Schnittstelle eines PCs ist ein Nullmodem erforderlich (sofern dies nicht bei der Konfiguration der seriellen Leitung bereits berücksichtigt wurde).

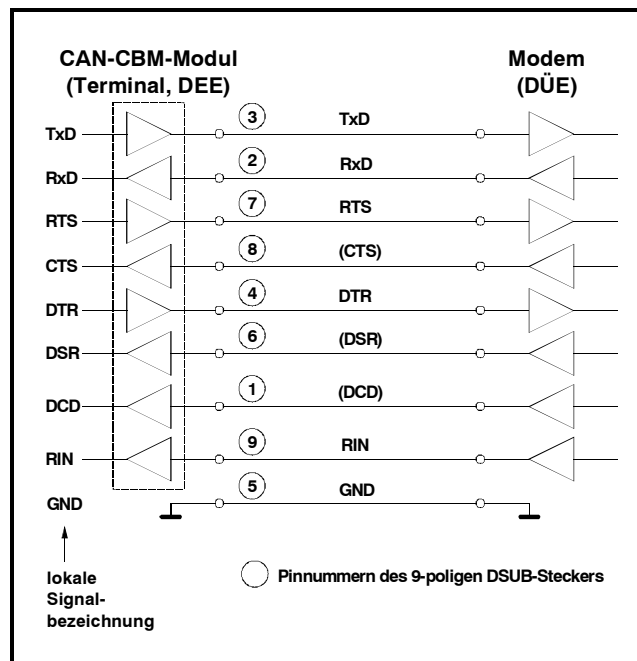


Abb. 2.2: Anschlußschema für RS-232-Betrieb

## 2.3 Funktion der Kodierschalter

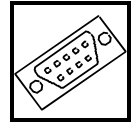
Kodierschalter	Funktion	Standard-Einstellung bei Auslieferung
SW210 (Low)	zur Zeit noch nicht definiert *)	-
SW211 (High)	zur Zeit noch nicht definiert *)	-

\*) Bei der CAN-CBM-Bridge mit 29-Bit-Identifizier-Software (Best.-Nr. C.2845.05) müssen die Kodierschalter beim Einschalten der Spannungsversorgung immer auf '0' eingestellt sein!

Tabelle 2.3: Funktion der Kodierschalter

## 2.4 LED-Anzeige

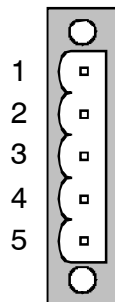
Die LEDs werden zur Zeit nicht unterstützt. Der Leuchtzustand ist nicht definiert.



### 3. Steckerbelegungen

#### 3.1 CAN-Bus (X400, Combicon-Style)

##### Pin-Zuordnung:



##### Pin-Belegung:

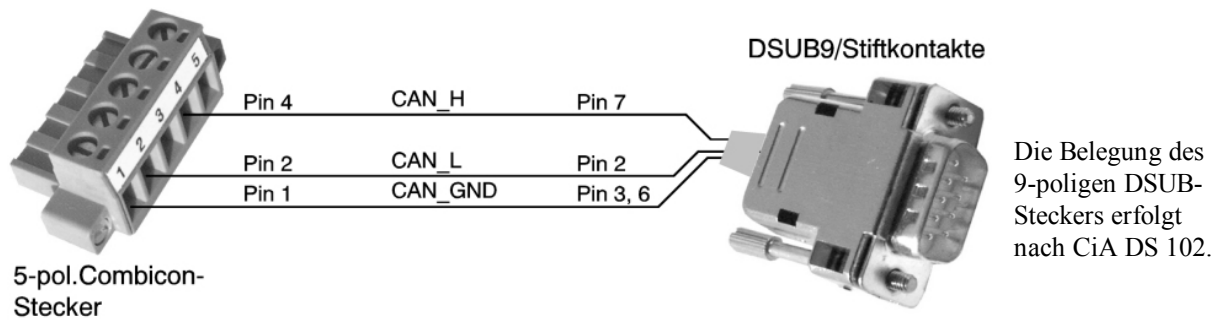
Pin	Signal
1	CAN_GND
2	CAN_L
3	n.c.
4	CAN_H
5	n.c.

##### Signalbeschreibung:

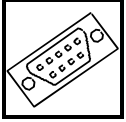
CAN\_L,

CAN\_H... CAN-Signalleitungen

CAN\_GND ... Bezugspotential des lokalen CAN-Physical Layers



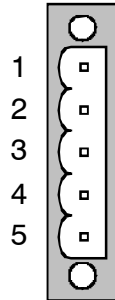
**Abb 3.1:** Adapter-Kabel 5-pol-Combicon auf 9-pol-DSUB



## Steckerbelegung

### 3.2 DeviceNet (X400, Combicon-Style)

#### Pin-Zuordnung:

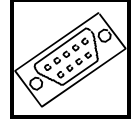


#### Pin-Belegung:

Pin	Signal
1	V-
2	CAN-
3	n.c.
4	CAN+
5	V+

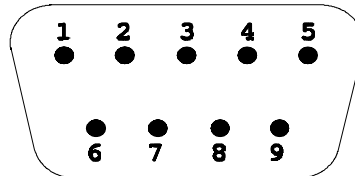
#### Signalbeschreibung:

- V+... Zuführung der Spannungsversorgung ( $U_{VCC} = 24\text{ V} \pm 4\%$ )  
V-... Bezugspotential zu V+ und zu CAN+/CAN-  
CAN+, CAN-... CAN-Signalleitungen



### 3.3 RS-232-Interface (X100, 9-pol. DSUB Stift)

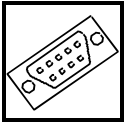
#### Pin-Zuordnung:



#### Pin-Belegung:

Signal	Pin		Signal
DSR* (Eingang)	6	1	DCD* (Eingang)
		2	RxD* (Eingang)
RTS* (Ausgang)	7	3	TxD* (Ausgang)
CTS* (Eingang)	8	4	DTR* (Ausgang)
RIN* (Eingang)	9	5	GND

9-poliger DSUB-Stecker



## Steckerbelegung

### 3.4 Spannungszuführung (X101, UEGM)

Die Spannungszuführung erfolgt über die im Gehäuse integrierten UEGM-Schraubverbinder. Sie können Leitungen mit einem Querschnitt von bis zu 2,5 mm<sup>2</sup> aufnehmen.

Die Schraubverbinder sind an beiden Seiten des Gehäuses gleich belegt. Sie können alternativ genutzt werden. Jeweils der mittlere Kontakt ist für +24V vorgesehen und die beiden äußeren für GND.

**Anmerkung:** Es ist **nicht** zulässig, die 24V-Versorgungsspannung ‘durchzuschleifen’, d.h. die eine Seite als 24V-Eingang und die andere Seite als 24V-Ausgang zu verwenden, um z.B. weitere Geräte zu versorgen !

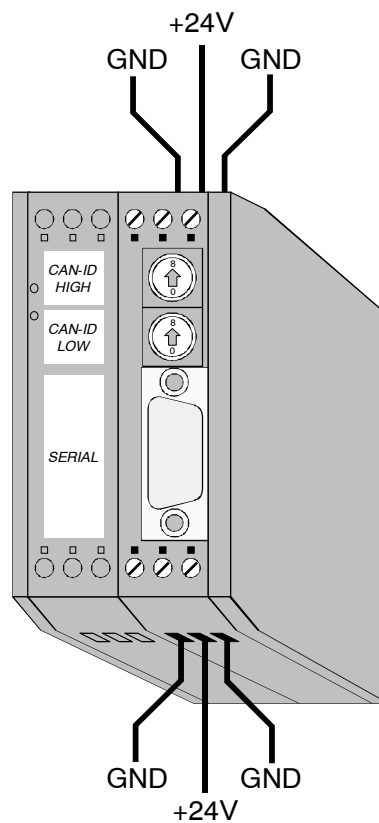
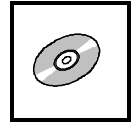


Abb 3.4: Spannungszuführung



## 4. Konfiguration des CAN-CBM-Bridge-Moduls

Das Kapitel beschreibt die Vorgehensweise für die Konfiguration des CAN-CBM-BRIDGE-Moduls die z.B. mit Hilfe des Hyperterminals von Windows erfolgen kann.

Im Folgenden werden die verwendeten Befehle zur Konfiguration der Parameter wie Baudrate und Verknüpfung der Identifier einmal für 11-Bit Identifier-Module und einmal für 29-Bit Identifier-Module beschrieben.

### 4.1 Bridge mit 11-Bit CAN-Identifier (C.2842.03)

Die serielle Schnittstelle des PCs ist auf die Werte zu konfigurieren, die in Kapitel 'Grundeinstellung der CAN-CBM-Module' (Seite 13), beschrieben sind.

#### 4.1.1 Befehlssatz

Nach dem Einschalten der Betriebsspannung wacht das CAN-CBM-BRIDGE-Modul im Hyperterminal mit folgender Meldung auf:

```

CAN-CBM-CAN2-Bridge - HyperTerminal
Datei Bearbeiten Ansicht Anruf Übertragung ?
Read from Flash... ok
esd gmbh, Hannover (c) 2000
BRIDGE V1.0 is running !
Press ctrl-A and type BRIDGE for help :-)
```

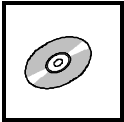
Abb. 4.1.1: Meldung bei Start

Jede Befehlseingabe muß mit der Tastenkombination >Strg< und >A< eingeleitet werden. Nach >Strg< und >A< erscheint ein '\*' am Zeilenanfang zur Bestätigung.

Geben Sie den Befehl ein und bestätigen Sie mit >RETURN<.

Eingabe	
Tastenkombination:	>CTRL< + >A<
Ausgabe:	*
Eingabe:	* 'BEFEHL' >RETURN<

Die Befehle können in Groß- oder Kleinschreibung eingegeben werden.



## Konfiguration

### BEFEHLE:

#### **bridge**

Über den Befehl **bridge** erhält man eine Übersicht über alle Befehle und die einstellbaren Baudraten mit zugeordnetem HexIndex.

```
*bridge
BRIDGE CLEAR           : Stop bridge operations and enable table edit
BRIDGE DEFAULT        : The same as CLEAR but erase the table
BRIDGE SET IDA=hhh IDB=hhh : Define connection CANbus-A<->CANbus-B
BRIDGE SET IDA=hhh REMOVE : Removes the connection
BRIDGE SET IDA=800     : Define all 1:1 connections CANbus-A<->CANbus-B
BRIDGE BAUDA=[HexIndex] : Set baudrate for CANbus-A
BRIDGE BAUDB=[HexIndex] : Set baudrate for CANbus-B
BRIDGE INIT           : Save the table and starts bridge operations
BRIDGE GET            : Dump the bridge table

Identifizier hhh=[0..7FF]

HexIndex Baudrate [KBits/s] HexIndex Baudrate [KBits/s]
0         1000                7         100
1         666.6                8         66.6
2         500                  9         50
3         333.3                A         33.3
4         250                  B         20
5         166                  C         12.5
6         125                  D         10
7         100                  E         5
8         66.6                 F         1600
```

Abb. 4.1.2: Übersicht

#### **bridge get**

Mit **bridge get** wird die aktuelle Konfiguration des Moduls angezeigt. In diesem Beispiel ist das Modul jedoch neu und noch nicht konfiguriert.

```
*bridge get
WAIT; BRIDGE CLEAR --
WAIT; BRIDGE INIT --
```

Abb. 4.1.3: Anzeige der aktuellen Konfiguration

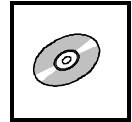
#### **bridge clear**

Bevor das Modul konfiguriert werden kann, muß die interne Firmware mit dem Befehl **bridge clear** gestoppt werden. Darauf erscheint folgende Meldung:

```
*bridge clear
Read from Flash... ok
BRIDGE clear ok !
```

Abb. 4.1.4: Stoppen der Firmware





Befindet sich das Modul noch im Betriebszustand RUN, wenn Sie versuchen es zu konfigurieren, erhalten Sie die folgende Fehlermeldung:

```
*bridge bauda=2
```

```
Bridge is not cleared !
```

### bridge bauda=HexIndex

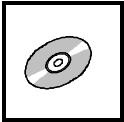
Befehl zum Setzen der Baudrate von CAN Netz A.

Wählen Sie aus der Index Tabelle den *HexIndex* für die Baudrate aus. Alternativ können auch direkt die Werte der Register BTR0 und BTR1 eingegeben werden.

<i>HexIndex</i>			Bit-Rate [kBit/s]	typische Werte der erreichbaren Leitungslänge $l_{\max}$ [m]	minimal erreichbare Leitungslänge $l_{\min}$ [m]
Index- Eingabe 0...E [HEX]	SJA1000-Register				
	BTR0 [HEX]	BTR1 [HEX]			
0	00	14	1000	37	20
1	00	18	666.6	80	65
2	00	1C	500	130	110
3	01	18	333.3	180	160
4	01	1C	250	270	250
5	02	1C	166	420	400
6	03	1C	125	570	550
7	04	1C	100	710	700
8	45	2F	66.6	1000	980
9	09	1C	50	1400	1400
A	4B	2F	33.3	2000	2000
B	18	1C	20	3600	3600
C	5F	2F	12.5	5400	5400
D	31	1C	10	7300	7300
E	00	16	800	59	42

Die Angaben in der Tabelle basieren auf den Grenzwerten des Bit-Timings des CAN-Protokolls, den Laufzeiten des lokalen CAN-Interface und den Laufzeiten des Kabels. Die Laufzeit des Kabels ist mit ca. 5,5 ns/m angenommen. In den Angaben sind weitere Einflüsse z.B. durch fehlende Abschlußwiderstände, den spezifischen Widerstand, die Geometrie des Kabels oder äußere Störeinflüsse bei der Übertragung nicht miteinbezogen!

**Tabelle 4.1:** Index der Baudrate



## Konfiguration

In diesem Beispiel soll die Baudrate auf 250 KBit/s gesetzt werden. Aus der Tabelle 4.1 erhält man den *HexIndex* = 4. Die Eingabe lautet dann wie folgt:

```
*bridge bauda=4
```

Oder, bei Eingabe der Registerwerte für BTR0 und BTR1:

```
*bridge bauda=011C
```

### **bridge baudb=HexIndex**

Befehl zum Setzen der Baudrate von CAN Netz B. Den *HexIndex* für die Baudrate erhalten Sie aus der Tabelle 4.1.

Für dieses Beispiel soll eine Baudrate von 500 KBits/s gesetzt werden. Der *HexIndex* ist dann 2.

```
*bridge baudb=2
```

### **bridge set**

Verbindungen unterschiedlicher Identifier der zwei CAN-Netze.

Die Identifier liegen zwischen 0 hex und 7FF hex.

Der Identifier **ida=800** verbindet CANbus-A und CANbus-B komplett. D.h. der Datenverkehr von Netz A wird auf Netz B gespiegelt und umgekehrt.

### **bridge set ida=ID Netz A idb=ID Netz B**

Zuordnung eines Identifier von CAN Netz A zu einem Identifier von CAN Netz B.

**ida= Identifier Netz a , idb= Identifier Netz b**

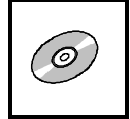
Im folgenden Beispiel wird der Identifier 200 hex des Netzes A mit dem Identifier 300 hex des Netzes B verbunden.

```
* bridge set ida=200 idb=300
```

Wählen Sie einen Identifier für Netz A der außerhalb des zulässigen Bereiches liegt, erhalten Sie folgende Fehlermeldung:

```
* bridge set ida=g5 idb=300
```

```
IDA Error
```



Entsprechend wird `IDB Error` für eine falsche Eingabe in Netz B angezeigt.  
Sind beide Eingaben falsch, werden beide Fehlermeldungen ausgegeben.  
Geben Sie die korrekten Werte ein.

Die aktuelle Konfiguration für das Beispiel erhalten Sie jetzt mit **bridge get:**

```
* bridge get
```

```
WAIT; BRIDGE CLEAR --  
WAIT; BRIDGE BAUDA=0004 --  
WAIT; BRIDGE BAUDA=0002 --  
WAIT; BRIDGE IDA=0200 IDB=0300 --  
WAIT; BRIDGE INIT --
```

### **bridge set ida=800**

Mit diesem Befehl werden der Reihe nach sämtliche Identifier beider Netze eins zu eins verbunden. Alle CAN Telegramme von Netz A und Netz B werden gespiegelt. D.h. Identifier 1 von Netz A wird mit Identifier 1 von Netz B verbunden, Identifier 2 von Netz A mit Identifier 2 von Netz B usw..



## Konfiguration

**bridge init** Nach erfolgter Konfiguration muß das CAN-CBM-BRIDGE-Modul mit dem Befehl **bridge init** gestartet werden.

```
* bridge init

Clear Flash... ok
Write to Flash... ok
BRIDGE init ok !
```

Das Modul ist nun im Betriebszustand RUN und erfüllt die gewünschten Funktionen

### **bridge set id[a]=(ID) remove**

Dieser Befehl löscht bereits konfigurierte Verbindungen von Netzen. Dafür muß das Modul zuvor mit **bridge clear** gestoppt werden.

In unserem Beispiel soll die zuvor konfigurierte Verbindung (ida=200, idb=300) wieder gelöscht werden:

```
* bridge set ida=200 remove
```

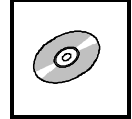
### **bridge default**

Dieser Befehl stoppt die Firmware und löscht mögliche Konfigurationen.

```
* bridge default

Clear Flash... ok
BRIDGE default ok
```

Das CAN-CBM-BRIDGE-Modul ist so nicht lauffähig. Zum Neustart muß das Modul wieder neu konfiguriert werden.



### 4.1.2 Konfigurationsbeispiel

Die Parameter der CAN-CBM-BRIDGE:

Channel A = 1000 kbaud

Channel B = 125 kbaud

Die CAN Telegramme sollen alle 1:1 gespiegelt werden.

```
<ctrl-A>
*bridge clear    <enter>
Read from Flash... ok
BRIDGE clear ok !

<ctrl-A>
*bridge get      <enter>
WAIT; BRIDGE CLEAR --
WAIT; BRIDGE INIT --

<ctrl-A>
*bridge bauda=0 <enter>

<ctrl-A>
*bridge baudb=6 <enter>

<ctrl-A>
*bridge set ida=800 <enter>

<ctrl-A>
*bridge init     <enter>
Clear Flash... ok
Write to Flash... ok
BRIDGE init ok !
```



## Konfiguration

### 4.2 Bridge mit 29-Bit CAN-Identifizier (C.2842.05)

Die serielle Schnittstelle des PCs ist auf die Werte zu konfigurieren, die in Kapitel 'Grundeinstellung der CAN-CBM-Module' (Seite 13), beschrieben sind. Für die Baudrate wird hier jedoch eine andere Einstellung gefordert.

Baudrate: 9,6 kBaud

**Achtung:** Beim Einschalten der Spannungsversorgung müssen beide Kodierschalter auf '0' eingestellt sein!  
Einstellungen ungleich '0' sind nur für Test- und Service-Aufgaben zulässig.

#### 4.2.1 Befehlssatz

Nach dem Einschalten der Betriebsspannung wacht das CAN-CBM-Modul im Hyperterminal mit der Ausgabe einer Meldung auf.

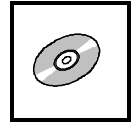
Sie können die Befehle jetzt direkt eingeben und mit >Enter< bestätigen.

#### BEFEHLE:

**R** Mit dem Befehl **R** wird die aktuelle Konfiguration des Moduls angezeigt. In diesem Beispiel hat das Modul noch die Default-Einstellung. Sie erhalten folgende Meldung:

Eingabe:   **R** >Enter<  
  
Ausgabe:   B0 : 6  
              B1 : 6

Bei Auslieferung sind beide CAN-Netze mit einer Default-Baudrate von 125 KBit/s, das entspricht einem *HexIndex* von 6 (siehe Tabelle 4.2.1), konfiguriert.



**Bn:HexIndex** Mit dem Befehl **Bn** : setzen sie die gewünschte Baudrate des CAN-Netzes mit der Netznummer **n**, mit: **n**= 0 für Netz 0  
**n**= 1 für Netz 1

Werden für *HexIndex* Werte von 0 hex bis F hex übergeben, so erfolgt die Einstellung der Baudrate entsprechend der folgenden Tabelle:

<i>HexIndex</i>	Baudrate [KBit/s]	<i>HexIndex</i>	Baudrate [KBit/s]
0	1000	8	66.6
1	666.6	9	50
2	500	A	33.3
3	333.3	B	20
4	250	C	12.5
5	166	D	10
6	125	E	5
7	100	F	reserviert

**Tabelle 4.2.1:** Index der Baudrate

Im folgenden Beispiel soll die Baudrate von Netz 1 ( $n=1$ ) auf 10 KBit/s gesetzt werden. Aus der Tabelle 4.2.1 erhält man den *HexIndex* = D. Die Eingabe lautet dann wie folgt:

Eingabe: **B1:D** >Enter<

**Bn:8000yyzz** Wahlweise können die Bit-Timing-Register des verwendeten SJA1000 auch direkt gesetzt werden. In diesem Fall wird bei Modulen mit den CAN-Controllern 82C200, SJA1000, 82527 (und allen anderen Controllern mit dieser Baudratenregisterstruktur) direkt der Registerwert für die Bit-Timing-Register BTR0 und BTR1 übergeben.

Dabei gilt: **n**: 0,1... Netznummer  
**yy**: Wert für BTR0  
**zz**: Wert für BTR1

Die Berechnung des Bit-Timings und der Baudrate nach den Registerwerten kann den Handbüchern der Controller 82C200 (Philips), SJA1000 (Philips) oder 82527 (Intel) entnommen werden.



## Konfiguration

### **I0:ID Netz 0 I1:ID Netz 1**

Dieser Befehl ordnet einem Identifier des CAN-Netzes **0** einen Identifier des CAN-Netzes **1** zu. Der Identifier *ID Netz 0*, der auf CAN-Netz 0 empfangen wird, wird Identifier *ID Netz 1* auf CAN-Netz 1 gesendet.

#### **Hinweis:**

Sollen 29-Bit Identifier (Wertebereich Bit 28...Bit 0) eingestellt werden, so muß Bit 29 zur Kennzeichnung gesetzt werden (20000000 hex entsprechend CANopen)!

In diesem Beispiel soll der 29-Bit Identifier 3456789 hex des Netzes 0 auf den 11-Bit Identifier 543 hex des Netzes 1 gesendet werden

```
Eingabe: I0:23456789 I1:543 >Enter<
```

Im folgenden Beispiel soll der 11-Bit Identifier *ID Netz 0* = 200 hex auf den 11-Bit Identifier *ID Netz 1* = 300 hex gesendet werden.

```
Eingabe: I0:200 I1:300 >Enter<
```

### **I1:ID Netz 1 I0:ID Netz 0**

Dieser Befehl ordnet einem Identifier des CAN-Netzes **1** einen Identifier des CAN-Netzes **0** zu. Der Identifier *ID Netz 1* der auf CAN-Netz 1 empfangen wird, wird an Identifier *ID Netz 0* auf CAN-Netz 0 gesendet.

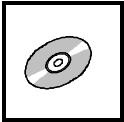
Im folgenden Beispiel soll der 29-Bit Identifier *ID Netz 1* = 4567893 hex auf den Identifier *ID Netz 0* = 205 hex gesendet werden.

```
Eingabe: I1:24567893 I0:205 >Enter<
```

Wird ein Identifier konfiguriert, der nicht zwischen 0 hex und 7FF hex oder im 29-Bit Bereich liegt, wird die Zuordnung nicht übernommen.  
Zur Zeit sind für beide Richtungen insgesamt 32 ID-Zuordnungen möglich.







## Konfiguration

Die aktuelle Konfiguration des CAN-CBM-Bridge-Moduls durch die vorausgegangenen Beispiele läßt sich nun mit dem Befehl **R** anzeigen.

```
Eingabe:  R >Enter<

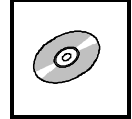
Ausgabe:  B0:6
          I0:200 I1:300
          I0:23456789 I1:543
          M0:1:00000000000000000000000000000000xxxxxxx1
          B1:D
          I1:24567893 I0:205
```

**E** Nach erfolgreicher Konfiguration werden die konfigurierten Daten durch den Befehl **E** in den Konfigurationsspeicher übernommen. Erst nach dieser Übernahme werden die Änderungen wirksam. Das CAN-CBM-Bridge-Modul befindet sich nun im Betriebszustand RUN und erfüllt die gewünschte Bridge Funktion.

```
Eingabe:  E >Enter<
```

**C** Eine Konfiguration kann mit dem Befehl **C** wieder gelöscht werden. Der Befehl löscht **alle** Identifier-Zuordnungen und setzt die CAN-Bitraten auf den Default- Wert von *HexIndex* 6, also einer Baudrate von 125 KBit/s Der Konfigurationsspeicher wird ebenfalls gelöscht.

```
Eingabe:  C >Enter<
```



### 4.2.2 Funktion der LEDs

Im fehlerfreien Zustand leuchten beide LEDs kontinuierlich. Tritt auf einem der beiden CAN-Netze ein Fehler auf, fängt die zugehörige LED an, einen Blinkcode auszugeben.

Die obere LED (siehe Frontansicht Seite 18) zeigt dabei Fehler im Netz 0 und die untere LED Fehler im Netz 1 an.

Leuchtzustand	Bedeutung
an	CAN-Netz OK
3x lang	Bus-Off
kurz-lang-lang	Transmit-Timeout
1x kurz	Transmit-Error

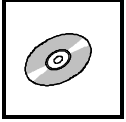
**Tabelle 4.2.2:** Leuchtzustände der LEDs

Sollte der CAN-Treiber einen anderen als den vorgenannten Fehler melden, wird über die serielle Schnittstelle folgender String ausgegeben:

```
cbs:xxxx
```

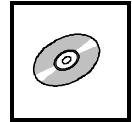
xxxx... Fehlernummer des CAN-Treibers

Tritt eine solche Fehlermeldung auf, kontaktieren Sie bitte **esd**.



## Konfiguration

Diese Seite ist bewußt unbedruckt.



## 5. Konfiguration des CAN-CBM-PLC/331-1/-2 Moduls

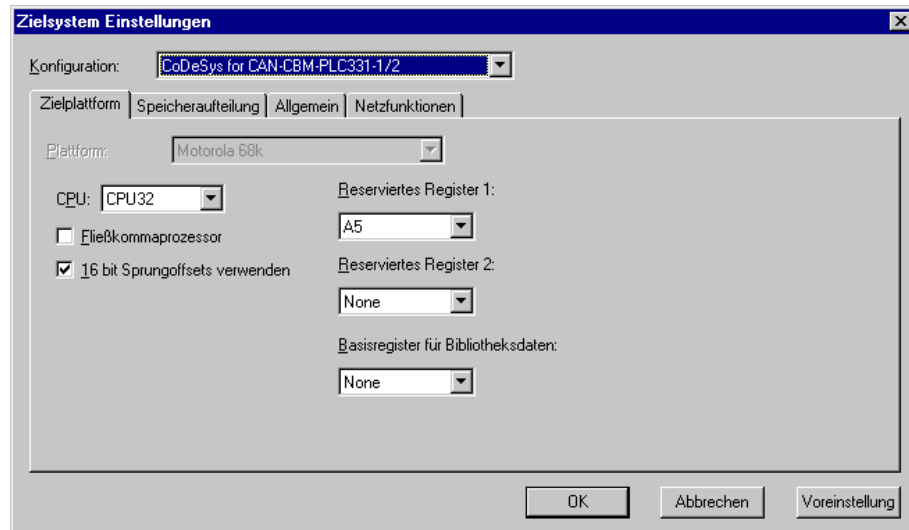
Das Kapitel beschreibt die Vorgehensweise für die Inbetriebnahme und Konfiguration des CAN-CBM-PLC/331-1/-2 Moduls mit der CoDeSys Programmierumgebung.

Die CoDeSys-Software wird mit einer Online-Hilfe geliefert, die die vielfältigen Möglichkeiten von CoDeSys beschreibt.

Weitere Informationen über CANopen enthält die Spezifikation CANopen CiA Draft Standard 301.

**Um das CAN-CBM-PLC/331-1/-2-Modul zu konfigurieren, ist wie folgt vorzugehen:**

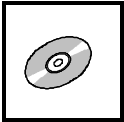
1. **Importieren der verschiedenen Dateien:** Installieren des Target Support Packages mit dem Installationsprogramm *Install Target.exe*. Kontrollieren Sie, ob die EDS-Dateien der gewünschten Module im Unterverzeichnis:  
`%CoDeSys%\Targets\ESD\ESD_CAN-Module\` des Bibliotheksverzeichnisses liegen. Importieren Sie benötigte EDS-Dateien gegebenenfalls.
2. **Starten der CoDeSys Entwicklungsumgebung.**
3. **Konfiguration:** Im Menü *Datei* den Menüpunkt *Neu* wählen. Es erscheint die in der folgenden Abbildung dargestellte Dialogbox *Zielsystem Einstellungen*.



**Abb. 1:** Einstellungen des Zielsystems

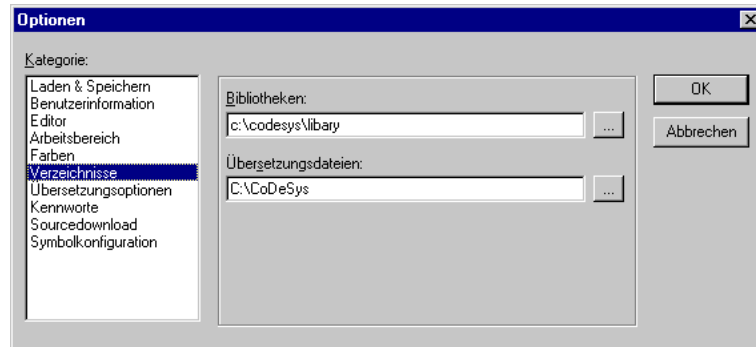
Die **Konfiguration** muß auf 'CoDeSys for CAN-CBM-PLC/331-1/2' eingestellt sein. Durch die Auswahl dieses 'Targets' wird die plattformspezifische Grundkonfiguration geladen.

**CPU** auf 'CPU32' einstellen. Bestätigen Sie mit **OK**.



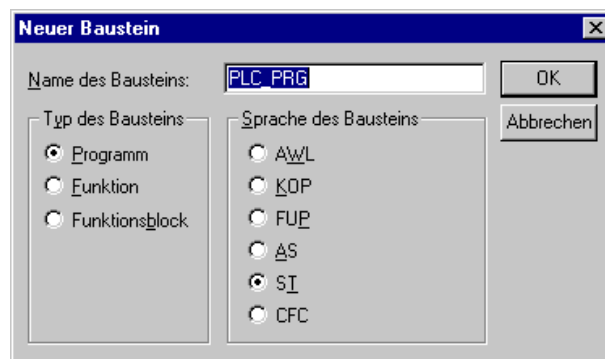
## Konfiguration

Kontrollieren Sie das aktuelle Projektverzeichnis und das Library-Verzeichnis. Wählen Sie dafür unter dem Menü **Projekt**, Menüpunkt **Optionen** und weiter Menüpunkt **Verzeichnisse** und kontrollieren Sie die Pfadangaben unter **Bibliotheken** (z.B.: C:\codesys\library) und **Übersetzungsdateien** (z.B.: C:\CoDeSys).



**Abb. 2:** Bibliotheken und Übersetzungsdateien kontrollieren

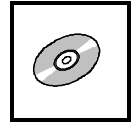
**4. Neuer Baustein:** Bestätigen Sie Ihre Auswahl im **Zielsystem Einstellungen** mit **OK**, öffnet sich die Dialogbox **Neuer Baustein:**



**Abb. 3:** Dialogbox **Neuer Baustein**

Der Baustein PLC\_PRG ist speziell vordefiniert und wird automatisch für jedes neue Projekt angelegt. Er darf nicht gelöscht oder umbenannt werden (gilt nicht bei Verwendung von Taskkonfiguration). Weitere Informationen dazu finden Sie in der CoDeSys Online-Hilfe.

Bestätigen Sie die Einstellungen ohne weitere Änderung mit **OK**.

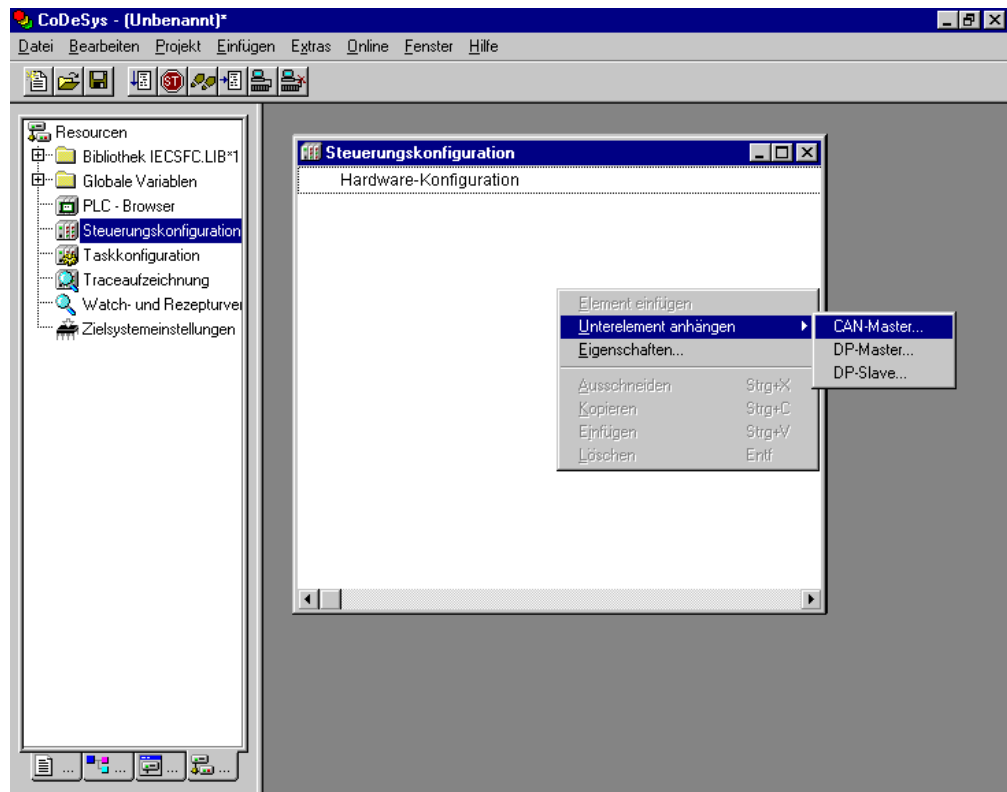


## 5. Auswahl des CAN-Master:

Wechseln Sie in das Register *Resourcen*  
(Register, linke untere Bildschirmecke).

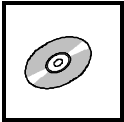


Wählen Sie den Menüpunkt *Steuerungskonfiguration* aus.



**Abb. 4:** CAN-Master auswählen

Es erscheint das Feld *Steuerungskonfiguration* auf dem Bildschirm. Klicken Sie mit der rechten Maustaste auf das Feld und wählen Sie den Menüpunkt *Unterelement anhängen* und dann *CAN-Master*.



## Konfiguration

### 6. CAN-Eigenschaften auswählen:

In der erscheinenden Dialogbox können Sie jetzt die gewünschten CAN-Eigenschaften eingeben:

**Globale CAN-Eigenschaften**

Baudrate: 125000

Com. Cycle Period (µsec): 0

Sync. Window Length (µsec): 0

Sync. COB-ID: 128 aktivieren:

Diagnoseadresse: %MBO

Adresse automatisch:  Automatisch starten:

Nodid: 1

OK

Abbrechen

**Abb. 5:** Globale CAN-Eigenschaften einstellen

Die Einstellungen der aufgelisteten Parameter sind abhängig von der jeweiligen Anwendung. Weitere Informationen finden Sie in der CoDeSys-Online Hilfe.

**Baudrate:** Stellen Sie die für die Übertragung gewünschte Baudrate ein (hier 125 kBaud)

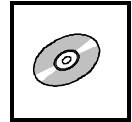
**Com. Cycle Period:** Zykluszeit für Sync. Telegramm, d.h. Zeit, die zwischen dem Absenden von zwei SYNC Telegrammen des SYNC-Masters vergeht. Die Zykluszeit ist abhängig von Slaves, Busgeschwindigkeit und interner Datenverarbeitungsrate.

**Achtung!** Die **Com.Cycle Periode** muß größer als die **Sync. Window Length** sein um sicherzustellen, daß alle Busteilnehmer die synchronen PDO's (Process Data Objects) empfangen haben. Siehe auch Abb. 6

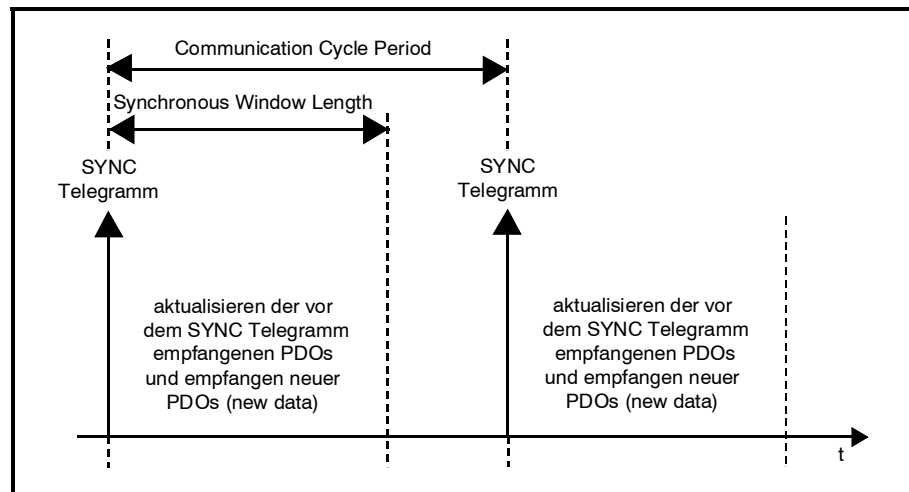
**Sync.Windows Length:** Gibt die Zeit an, die vom Absenden einer SYNC vergeht, bis alle synchronen PDO's gesendet worden sind. Da sie kleiner ist als die **Com.Cycle Periode** wird so sichergestellt, dass alle geforderten Daten übertragen werden, bevor ein neues SYNC Telegramm gestartet werden kann. Siehe Abb. 6

**Achtung!** Sind die Felder **Com. Cycle Period** und **Sync.Windows Length**





mit '0' belegt, werden keine SYNC Telegramme verschickt.



**Abb. 6:** Bus Synchronisation

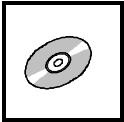
**Sync. COB-ID:** Identifier unter dem SYNC Telegramme gesendet oder empfangen werden.

**Diagnoseadresse:** Hier muß eine Merkadresse angegeben werden, unter der die Diagnosedaten abgelegt werden.

**Node ID:** Identifier der CAN-CBM-PLC/331-1/2 (zwischen 1 und 127, dezimale Eingabe).

Bestätigen Sie ihre Auswahl mit **OK**, wird der CAN-Master im Feld **Steuerungskonfiguration** in das Konfigurationsschema unter Hardwarekonfiguration aufgenommen (Siehe Abb. 7. 'Unterelemente anhängen').

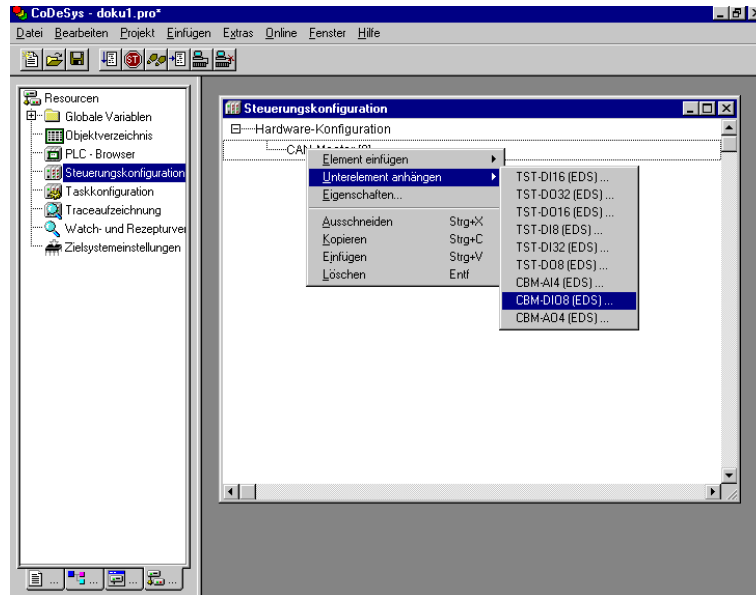
Weitere Informationen und Erläuterungen finden Sie in der CANopen Spezifikation 'CANopen CiA Draft Standard 301' Kap. 9.3.1



## Konfiguration

### 7. Unterelement einfügen:

Nach der Konfiguration des Masters wird nun das restliche CAN-Netzwerk zusammengestellt und konfiguriert. Um weitere Elemente einzufügen, klicken Sie mit der rechten Maustaste auf den eingefügten CAN-Master und gelangen über den Menüpunkt *Unterelemente anhängen* zu einer Auswahl der Module.



**Abb. 7:** Unterelemente anhängen

Klicken Sie das gewünschte Modul (hier z.B. CBM-DIO8) mit der linken Maustaste an, erscheint eine Dialogbox (siehe Abb. 8) in der Sie die Eigenschaften des ausgewählten CAN-Teilnehmers festlegen können.

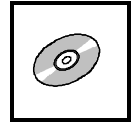
Ist für den gewünschten CAN-Teilnehmer kein Eintrag bzw. keine EDS-Datei vorhanden, kann durch die TST-Dateien die nicht verfügbare EDS-Datei ersetzt werden. Die TST-Dateien sind für einfache Anwendungen konfigurierte EDS-Dateien. Die Endung des Namens der Test-Dateien (TST-Dateien) erklärt dabei die jeweilige Funktion der Datei:

#### TST-xyyyzz

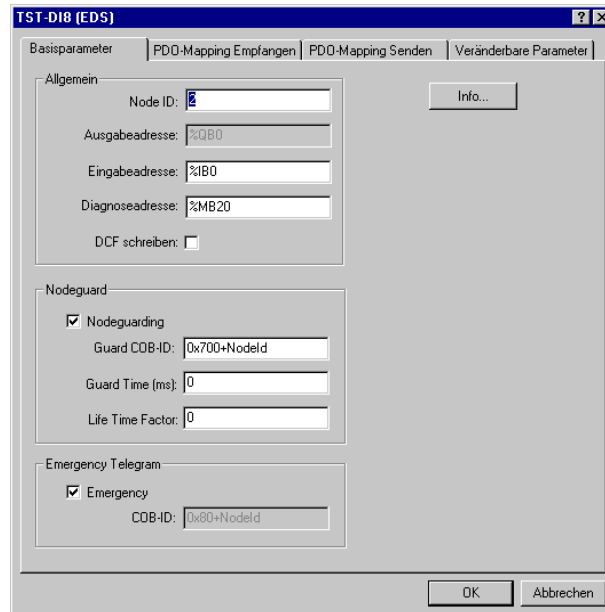
Stamm des Namens: **TST-**  
folgender Buchstabe: **xx...** D, A (Digital, Analog)  
folgende Buchstaben: **yy...** I, O, IO (Input, Output, In/Output)  
folgende Zahlen: **zz...** gibt die Anzahl der Netze  
z.B.: 8, 16, 32 oder 64

Beispiel:

TST-DI8 (EDS)      Digitaler Input, 8 Bits  
TST-DO32 (EDS)    Digitaler Output, 32 Bits



## 8. Basisparameter:



**Abb. 8:** Basisparameter der Unterelemente einstellen

Geben Sie die folgenden Parameter entsprechend Ihrer Applikation ein. Eine ausführliche Beschreibung der Parameter entnehmen Sie bitte dem Kapitel 'Basisparameter eines CAN-Moduls' der CoDeSys-Online Hilfe.

**Node ID:** Identifier des CAN Slaves

**Eingabeadresse:** Adresse unter der das Modul vom Anwenderprogramm angesprochen wird.

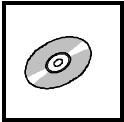
**Diagnoseadresse:** Adresse unter der die Diagnosedaten abgelegt werden.

**DCF schreiben:** Erstellen einer DCF-Datei nach dem Einfügen einer EDS-Datei, wenn aktiviert.

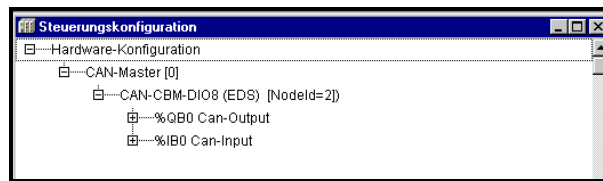
Alle Prozeßdaten im CAN-Netzwerk werden über den Eingangs- und Ausgangs-Adress-Bereich des CAN-CBM-PLC/331-1/2-Moduls gelesen oder geschrieben. (Über z.B.: %IB4 ... Eingangsbyte 4, %QB6 ... Ausgangsbyte 6). CAN-spezifische Daten z.B.: Identifier, RTR... werden im Anwendungsprogramm selber nicht verwendet.

Werden für die Überwachung der Teilnehmer die Option **Nodeguarding** und das **Emergency-Telegramm** gewünscht, aktivieren Sie diese. Weitere Informationen dazu sowie zu den Menüpunkten **PDO-Mapping empfangen**, **PDO-Mapping senden** und **Veränderbare Parameter** entnehmen Sie bitte ebenfalls der CoDeSys Online-Hilfe.

Bestätigen Sie Ihre Auswahl mit **OK**



## Konfiguration



**Abb. 9:** Steuerungskonfiguration

Das ausgewählte Modul (hier CAN-CBM-DIO8) wird nun im Fenster **Steuerungskonfiguration** in dem Konfigurationsschema als Unterelement angezeigt. Weitere Informationen über das jeweilige Element, wie die Eingabe- und Ausgabeadresse, erhalten Sie, wenn Sie mit der linken Maustaste auf das vorangestellte Pluszeichen klicken.

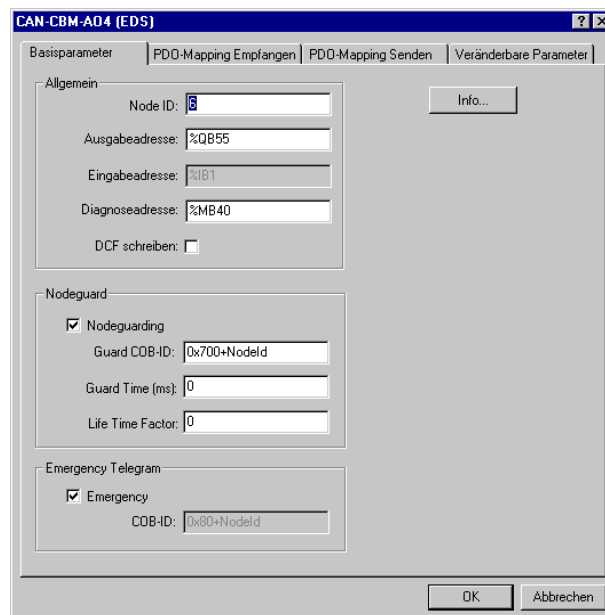
### 9. Weitere Module einfügen:

Um weitere Module einzubinden, wiederholen Sie die unter Punkt 7. und 8. beschriebenen Schritte.

#### Beispiel:

Als weiteres Modul wird das Modul CBM-AO4 (EDS) wie unter Punkt 7. 'Unterelemente anhängen' beschrieben, ausgewählt.

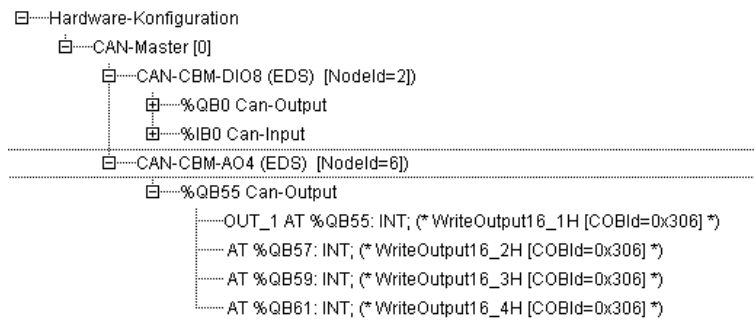
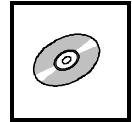
Es öffnet sich die Dialogbox Eigenschaften **CAN-CBM-AO4**:



**Abb. 10:** Beispiel CAN-CBM-AO4

Der Identifier des gewünschten Moduls hat die Node ID = 6, die Ausgabeadresse: %QB55 (das Ausgangsbyte 55), die Diagnoseadresse: %MB40 (das Byte an der Speicherstelle im Merker 40), die Guard COB-ID ergibt sich aus 0x700+Node ID (hier NodeID= 6)

Bestätigen Sie mit OK, erscheint das folgende Fenster:



**Abb. 11:** Konfigurationsbeispiel

Das Modul CAN-CBM-AO4 mit der Node ID = 6 ist nun aufgenommen. Durch Mausklick auf die Pluszeichen erhalten Sie weitere Informationen über die Konfiguration. Die Ausgabeadresse des ersten Netzes beträgt: %QB55  
 Zu jedem Netz des gewählten Moduls wird folgende Funktionsbeschreibung ausgegeben:

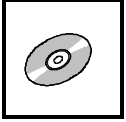
Funktions- beschreibung	<i>name</i> AT% <i>adresse</i> :Datentyp;(*Kommentar*)
Beispiel	OUT_1 AT%QB55:INT;(*WriteOutput16_1H [COBId0x306]*)

*name*: Wird ein weiteres Ausgangsmodul vor dem CAN-CBM-AO4-Modul eingefügt, erhöht sich die Ausgabeadresse entsprechend der Anzahl der Ausgänge des neu eingefügten Moduls automatisch. Um zu verhindern, dass nun alle Programme, in denen diese Adresse erscheint, ebenfalls geändert werden müssen, kann der globalen Variablen ein Name zugewiesen werden. Durch Mausklick auf AT erscheint ein kleines Eingabefenster. Hier läßt sich ein Name für die globale Variable, die auf der Adresse %QB55 liegt, eingeben. Dieser Name kann nun, unabhängig von anderen Modulen, für alle Programmierungen verwendet werden.

AT%*adresse*: ausgewählte Ausgabeadresse, %QB55

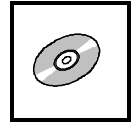
Datentyp: hier vom Typ Integer

(\*Kommentar\*): WriteOutput16\_1H: Ausgabe = 16 Bit auf Netz 1(1H), unter der COB-ID = 0x306 werden die Prozessdaten im CAN-Netzwerk gesendet.



## Konfiguration

Diese Seite ist bewußt unbedruckt.

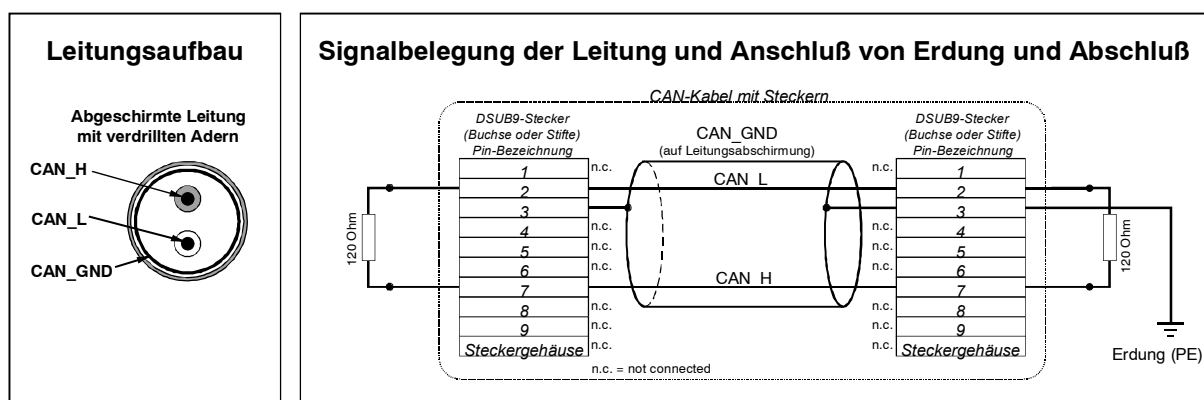


## 6. Korrekte Verdrahtung galvanisch getrennter CAN-Netze

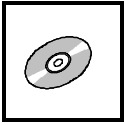
Generell sind bei der Verdrahtung sämtliche gültigen Richtlinien (DIN, VDE) bzgl. EMV-gerechtem Aufbau, Leitungsführung, Leiterquerschnitte, zu verwendende Materialien, Mindestabstände, Blitzschutz etc. zu beachten.

Die folgenden **Grundregeln** für die CAN-Bus Verdrahtung sollten unbedingt beachtet werden:

1	Ein CAN-Netz darf sich nicht verzweigen (Ausnahme: kurze Stichleitungen) und muss an beiden Enden mit dem Wellenwiderstand der Leitung (in der Regel $120 \Omega \pm 10\%$ ) abgeschlossen werden (zwischen den Signalen CAN_L und CAN_H und <b>nicht</b> gegen GND)!
2	Eine CAN-Datenleitung benötigt <b>zwei verdrehte</b> Adern (Twisted Pair) und eine Leitung zur Mitführung des Bezugspotenzials CAN_GND! Hierzu sollte die Abschirmung des Kabels verwendet werden!
3	Das mitgeführte Bezugspotenzial CAN_GND muss an <b>einem</b> Punkt mit dem Erdpotenzial (PE) verbunden werden. Es muss genau <b>eine</b> Verbindung mit Erde hergestellt werden!
4	Die Baudrate muss an die Leitungslänge angepasst werden.
5	Stichleitungen sind so kurz wie möglich zu halten ( $l < 0,3 \text{ m}$ )!
6	Bei doppelt abgeschirmten Leitungen muss der äußere Schirm an <b>einem</b> Punkt mit dem Erdpotenzial (PE) verbunden werden. Es darf nicht mehr als <b>einen</b> Anschluss an Erde geben.
7	Es ist ein geeigneter Leitungstyp (Wellenwiderstand ca. $120 \Omega \pm 10\%$ ) zu verwenden und der Spannungsabfall auf der Leitung ist zu beachten!
8	Die CAN-Leitungen sollten nicht in unmittelbarer Nähe von Störquellen verlegt werden. Lässt sich dies nicht vermeiden, so sind doppelt abgeschirmte Leitungen vorzuziehen.



**Abb.:** Aufbau und Anschluss der Leitung



## Konfiguration

## Verkabelung

- bei Geräten, die pro CAN-Netz nur einen CAN-Stecker besitzen, T-Stück und Stichleitung (kürzer als 0,3 m) verwenden (als Zubehör lieferbar)

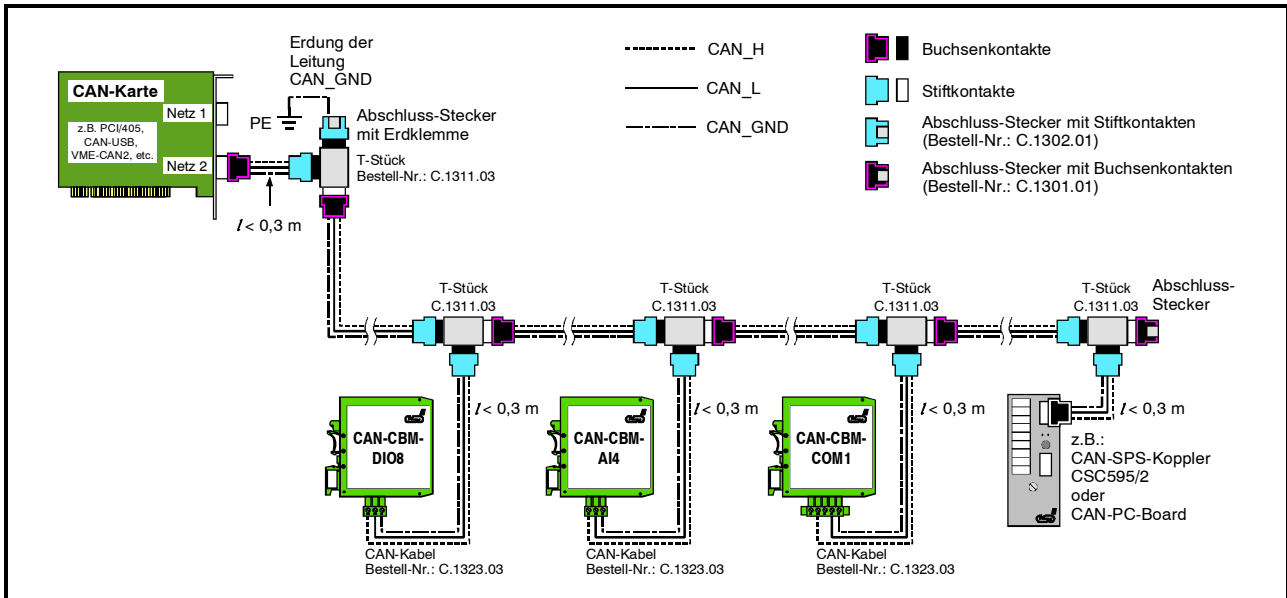


Abb.: Beispiel für korrekte Verdrahtung (bei Verwendung einfach abgeschirmter Leitungen)

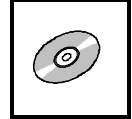
## Abschlusswiderstand

- externen Abschlussstecker verwenden, weil dieser später leichter auffindbar ist!
- 9-polige DSUB-Abschlussstecker mit Stift- oder Buchsenkontakten und Erdungsklemme sind als Zubehör erhältlich

## Erdung

- CAN\_GND muss in der CAN-Leitung mitgeführt werden, weil die einzelnen esd-Module galvanisch voneinander getrennt sind!
- CAN\_GND muss an **exakt einem** Punkt im Netz mit dem Erdpotential (PE) verbunden werden!
- jeder CAN-Teilnehmer ohne galvanisch getrenntes Interface wirkt wie eine Erdung, darum: maximal einen Teilnehmer ohne Potenzialtrennung anschließen!
- Erdung kann z.B. an einem Abschlussstecker vorgenommen werden





## Leitungslänge

- Optokoppler verzögern die CAN-Signale. Durch den Einsatz schneller Optokoppler und den Test jedes Boards bei 1 MBit/s kann esd jedoch eine erreichbare Länge von 37 m bei 1 MBit/s garantieren. Voraussetzung hierfür ist ein abgeschlossenes Netz ohne Impedanzstörungen, wie z.B. längere Stichleitungen. (Ausnahme: CAN-CBM-DIO8, -AI4, und -AO4 hier nur 10 m bei 1 MBit/s.)

Bit-Rate [kBit/s]	typische Werte der erreichbaren Lei- tungslänge mit esd- Interface $l_{\max}$ [m]	CiA-Empfehlungen (07/95) für erreichbare Leitungslängen $l_{\min}$ [m]
1000	37	25
800	59	50
666.6	80	-
500	130	100
333.3	180	-
250	270	250
166	420	-
125	570	500
100	710	650
66.6	1000	-
50	1400	1000
33.3	2000	-
20	3600	2500
12.5	5400	-
10	7300	5000

**Tabelle:** Erreichbare Leitungslängen in Abhängigkeit von der Bitrate beim Einsatz von esd-CAN-Interfaces



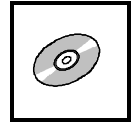
## Konfiguration

### Beispiele für CAN-Bus Leitungstypen

Hersteller	Leitungstyp
U.I. LAPP GmbH Schulze-Delitzsch-Straße 25 70565 Stuttgart <a href="http://www.lappkabel.de">www.lappkabel.de</a>	z.B. UNITRONIC ®-BUS CAN UL/CSA (UL/CSA approved) UNITRONIC ®-BUS-FD P CAN UL/CSA (UL/CSA approved)
ConCab GmbH Äußerer Eichwald 74535 Mainhardt <a href="http://www.concab.de">www.concab.de</a>	z. B. BUS-PVC-C (1 x 2 x 0,22 mm <sup>2</sup> ) Best.-Nr.: 93 022 016 (UL appr.) BUS-Schleppflex-PUR-C (1 x 2 x 0,25 mm <sup>2</sup> ) Best.-Nr.: 94 025 016 (UL appr.)
SAB Bröckskes GmbH&Co. KG Grefrather Straße 204-212b 41749 Viersen <a href="http://www.sab-brockskes.de">www.sab-brockskes.de</a>	z.B. SABIX® CB 620 (1 x 2 x 0,25 mm <sup>2</sup> ) Best.-Nr.: 56202251 CB 627 (1 x 2 x 0,25 mm <sup>2</sup> ) Best.-Nr.: 06272251 (UL appr.)



**Hinweis:** Fertig konfektionierte Leitungen in diversen Längen können bei **esd** bezogen werden.



## 7. CAN-Bus Troubleshooting Guide

Der CAN-Bus Troubleshooting Guide ist eine Anleitung zum Auffinden und Beseitigen der häufigsten Hardware-Fehlerursachen in der CAN-Bus-Verdrahtung.

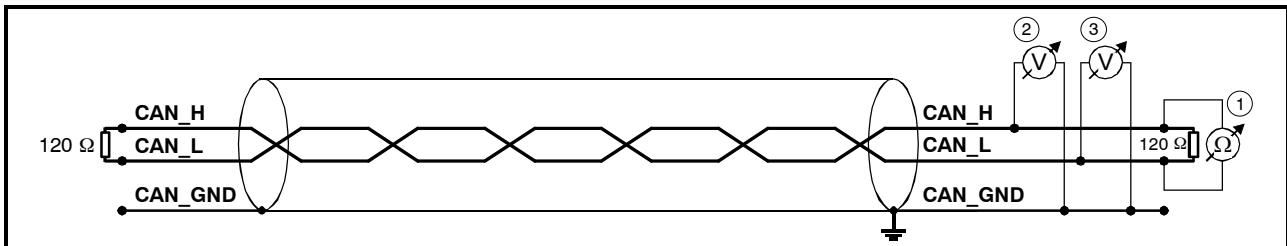


Abb. : Vereinfachtes Schaltbild eines CAN-Netzwerks

### 7.1 Bus-Abschluss

Der Bus-Abschluss wird verwendet, um den Widerstand eines Knotens an den Widerstand der verwendeten Busleitung anzupassen. Ist die Impedanz falsch angepasst, wird das gesendete Signal nicht ganz von der Last aufgenommen und zum Teil in die Übertragungsleitung zurück reflektiert. Sind die Quellen-, Übertragungsleitungs- und Last-Impedanz gleich groß, so werden die Reflexionen beseitigt. Dieser Test misst den Gesamtwiderstand der beiden CAN-Datenleitungen und des angeschlossenen Abschlusswiderstandes.

Zum Testen, verfahren Sie bitte wie folgt:

1. Schalten Sie die Versorgungsspannungen aller angeschlossenen CAN-Knoten aus.
2. Messen Sie den DC-Widerstand zwischen CAN\_H und CAN\_L in der Mitte und an den Enden des Netzwerks **1** (siehe obere Abbildung).

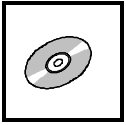
Der gemessene Wert sollte zwischen 50 und 70 Ohm liegen.

Liegt der ermittelte Wert unter 50 Ohm, stellen Sie bitte sicher, dass:

- kein **Kurzschluss** zwischen den CAN\_H- und CAN\_L-Leitungen besteht
- **nicht mehr als zwei** Abschlusswiderstände angeschlossen sind
- die Transceiver der einzelnen Knoten nicht defekt sind.

Liegt der ermittelte Wert über 70 Ohm, stellen Sie bitte sicher, dass:

- alle CAN\_H- und CAN\_L- Leitungen korrekt angeschlossen sind
- zwei Abschlusswiderstände von **je 120 Ohm** an Ihr CAN-Netzwerk angeschlossen sind (einer an jedem Ende).



## Konfiguration

### 7.2 CAN\_H/CAN\_L-Spannungen

Jeder Knoten verfügt über einen CAN-Transceiver, der differentielle Signale auf den Datenleitungen generiert. Ruht die Netzwerk-Kommunikation, betragen die CAN\_H- und CAN\_L-Spannungen etwa 2.5 V. Defekte Transceiver können diese Ruhespannungen verändern und die Netzwerk-Kommunikation unterbrechen.

Um auf defekte Transceiver zu testen, verfahren Sie bitte wie folgt:

1. Schalten Sie alle Versorgungsspannungen an.
2. Beenden sie jegliche Netzwerk-Kommunikation.
3. Messen Sie die DC-Spannung zwischen CAN\_H und GND **2** (siehe Abbildung auf vorhergehender Seite).
4. Messen Sie die DC-Spannung zwischen CAN\_L und GND **3** (siehe Abbildung auf vorhergehender Seite).

Die gemessene Spannung sollte zwischen 2.0 V und 4.0 V liegen.

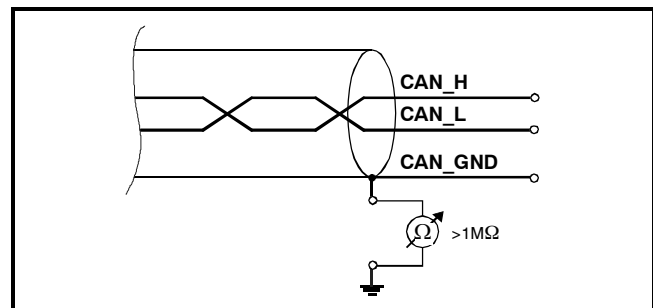
Ist die Spannung kleiner als 2.0 V oder größer als 4.0 V, ist es möglich, dass ein oder mehrere Knoten defekte Transceiver haben. Bei einer Spannung die unter 2.0 V liegt, überprüfen Sie bitte den Anschluss der CAN\_H- und CAN\_L-Leitungen. Bei einer Spannung, die oberhalb von 4.0 V liegt, überprüfen Sie bitte auf überhöhte Spannung.

Um einen Knoten mit einem defekten Transceiver zu finden, überprüfen Sie bitte den Widerstand des CAN-Transceivers (siehe Kapitel: “7.4 CAN Transceiver-Widerstands-Test”).

### 7.3 Erdung

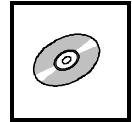
Die Abschirmung des CAN-Netzwerks darf nur an einer einzigen Stelle geerdet werden. Dieser Test zeigt an, ob die Abschirmung an mehreren Stellen geerdet ist. Zum Testen verfahren Sie bitte wie folgt:

1. Trennen Sie die Abschirmungsleitung (Shield) von dem Erdpotenzial.
2. Messen Sie den DC-Widerstand zwischen Shield und Erdpotenzial (siehe nebenstehende Abbildung).
3. Verbinden Sie die Abschirmungsleitung mit dem Erdpotenzial.



**Abb.:** Vereinfachtes Schaltbild Erdungsmessung

Der Widerstand sollte größer als ein 1 MOhm sein. Ist er kleiner, suchen Sie bitte nach zusätzlichen Erdungen der Shield-Leitung.



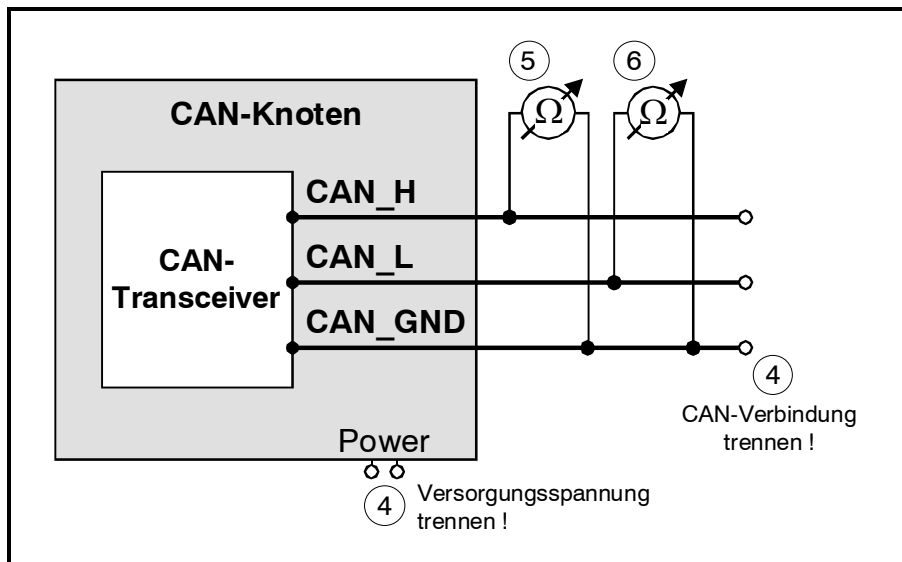
## 7.4 CAN Transceiver-Widerstands-Test

CAN Transceiver verfügen über einen Schaltkreis, der CAN\_H und CAN\_L kontrolliert. Die Erfahrung zeigt, dass elektrische Beschädigung an einem oder beiden der Schaltkreise den Leckstrom in diesen Schaltkreisen erhöhen kann.

Um den Leckstrom durch die CAN-Schaltungen zu messen, benutzen Sie bitte ein Widerstandsmessgerät und:

1. Trennen Sie den Knoten vom Netzwerk. Lassen Sie den Knoten **ausgeschaltet** **4** (siehe untere Abbildung).
2. Messen Sie den DC-Widerstand zwischen CAN\_H und CAN\_GND **5** (siehe untere Abbildung).
3. Messen Sie den DC-Widerstand zwischen CAN\_L und CAN\_GND **6** (siehe untere Abbildung).

Der Widerstand sollte zwischen 1 MOhm und 4 MOhm liegen. Liegt der Widerstand nicht in dem Bereich, ist der CAN-Transceiver möglicherweise defekt.



**Abb.:** Vereinfachtes Schaltbild eines CAN-Knotens