

CAN-Mini-ADI4/DO4

**CAN-Modul mit
4 digitalen/analogen Eingängen
und
4 digitalen Ausgängen**

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. 205
30165 Hannover

Tel.: 0511/372 98-0
FAX : 0511/372 98-68
E-Mail: info@esd.electronics.com
Internet: www.esd-electronics.com

Dokument-Datei:	I:\texte\Doku\MANUALS\CAN\Can-Mini\adi4do4\CMIAD12H.ma6
Datum der Druckvorlagenerstellung:	08.12.2000

Platinenversion:	HVST1
-------------------------	-------

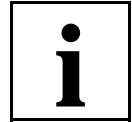
Änderungen nach Kapiteln:

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

Platinen-Version	Handbuch-Revision	Kapitel	Änderungen gegenüber Vorversion
HVST1	1.2	2.2 2.3	Änderung der Default- Konfigurationen A, B C und D

Weitere technische Änderungen vorbehalten.

Inhalt	Seite
1. Übersicht	3
1.1 Blockschaltbild	3
1.2 Gehäuseansicht mit LED- und Steckerbeschreibung	4
1.3 Gehäuseansicht	4
1.3.1 Bedeutung der LEDs Output_Error, CAN_Error, Module_Status und Run	5
1.3.2 Bedeutung der gelben LEDs	5
1.4 Zusammenfassung der technischen Daten	7
1.4.1 Allgemeine technische Daten	7
1.4.2 : Controller-Baugruppen	8
1.4.3 Digitale Ausgänge	9
1.4.4 Referenzspannungsausgang	9
1.4.5 Analog/Digitale Eingänge	10
1.5 Bestellhinweise	11
2. Hardware-Konfiguration	13
2.1 Platinenansichten	13
2.2 Default-Einstellung des Kodierschalters und der Brücken	15
2.3 Beschreibung des Kodierschalters und der Brücken	17
2.3.1 Einstellung der Bitrate, der Node-ID (Module-No.) und der Default-Konfiguration über den Kodierschalter oder die Lötbrücken LB191...LB198	17
2.3.1.1 Allgemeines	17
2.3.1.2 Default-Konfiguration	18
2.3.1.3 Node-ID	19
2.3.1.4 Bitrate	19
2.3.2 Anschluß der Kodierschalter-Bits 7 und 8 an Treiberbaustein über LA197 und LA198	20
2.3.3 Konfiguration der Analog/Digitalen-Eingänge über LB220...LB223	21
2.3.4 Anschluß der internen Referenzspannung an den I/O-Stecker P200 über LB224 ...	23
2.3.5 Auswahl des Programmspeichers des Microcontrollers über LB100	23
3. Beschreibung der Baugruppen	25
3.1 CAN-Bus-Interface	25
3.2 Digitale Ausgänge	25
3.3 Analog/Digitale Eingänge	26
3.3.1 Übersicht	26
3.3.2 Analog oder digitaler Eingang	28
3.3.3 Low-Side-Switch und Widerstandsmessung	28
3.3.4 High-Side-Switch	29
3.3.5 Referenzspannungsausgang	29
3.3.6 One-Wire-Option	30
4. Anhang	31
4.1 Steckerbelegungen	31
4.1.1 I/O-Steckverbinder X200	31



1. Übersicht

1.1 Blockschaltbild

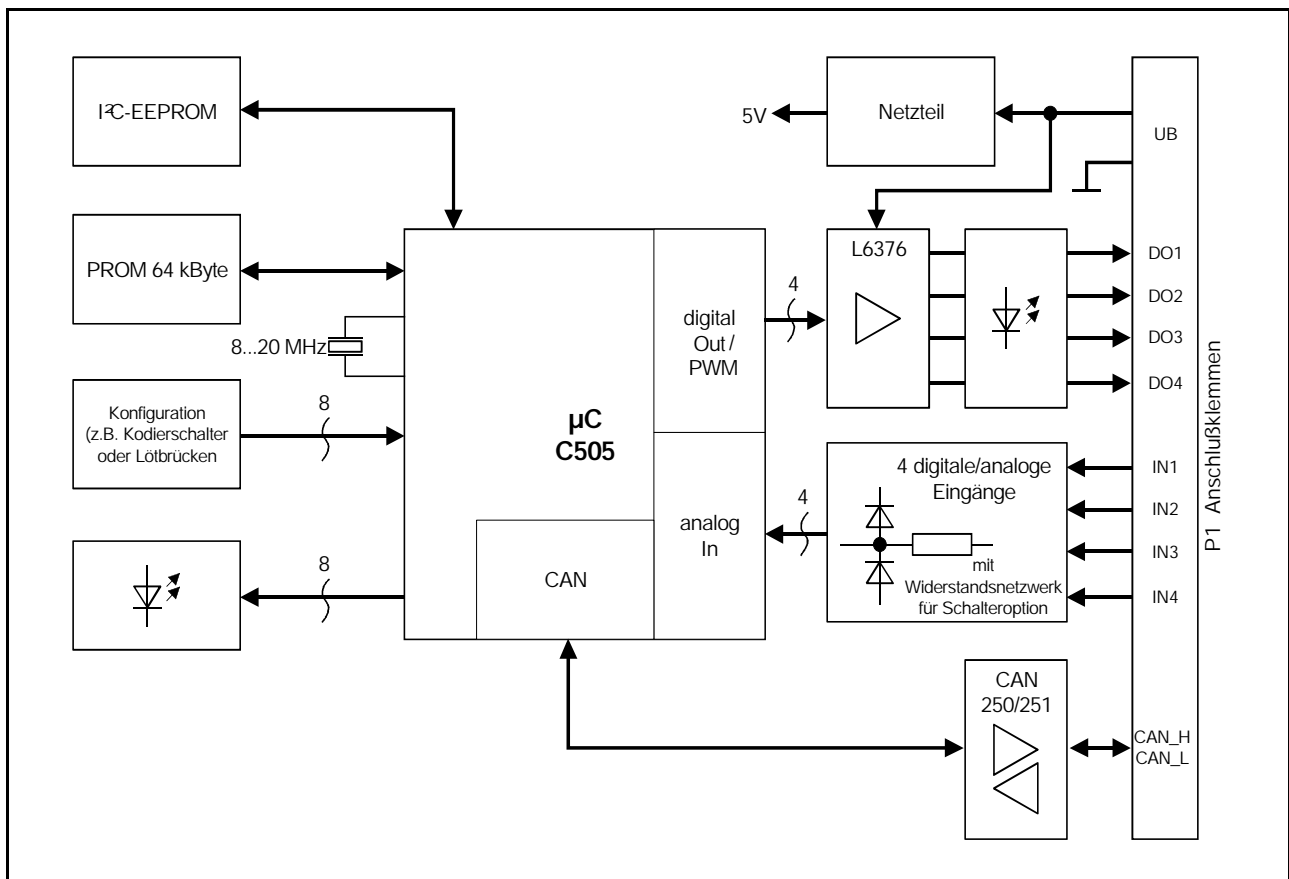


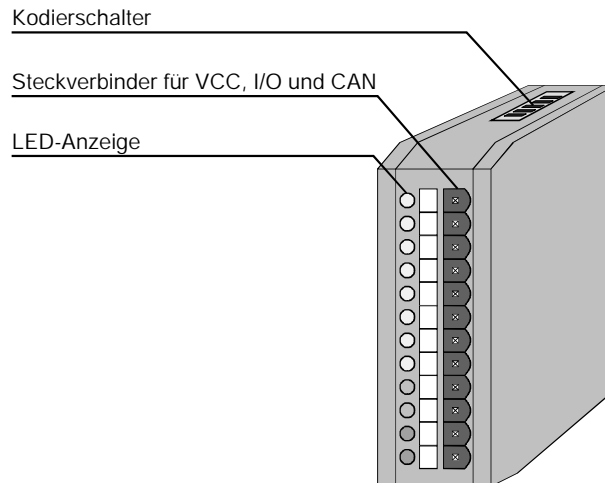
Abb. 1.1.1: Blockschaltbild des CAN-Mini-ADI4/DO



Übersicht

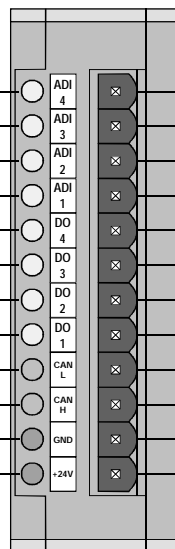
1.2 Gehäuseansicht mit LED- und Steckerbeschreibung

1.3 Gehäuseansicht



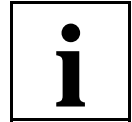
LEDs:

Analog/Digital_Input_4
Analog/Digital_Input_3
Analog/Digital_Input_2
Analog/Digital_Input_1
Digital_Output_4
Digital_Output_3
Digital_Output_2
Digital_Output_1
Output_Error
CAN_Error
Module_Status
Run



Stecker-Pins:

12 Analog/Digital_Input_4
11 Analog/Digital_Input_3
10 Analog/Digital_Input_2
9 Analog/Digital_Input_1
8 Digital_Output_4
7 Digital_Output_3
6 Digital_Output_2
5 Digital_Output_1
4 CAN_L
3 CAN_H
2 GND
1 Power Supply +24V



1.3.1 Bedeutung der LEDs Output_Error, CAN_Error, Module_Status und Run

Der aktuelle Zustand des Moduls wird über die LEDs 'Module_Status', 'Run' und 'CAN_Error' angezeigt. Die folgende Tabelle zeigt anhand der LED-Leuchtzustände den Zustand des Moduls und die CAN-Fehlermeldungen.

LED	Zustand des Moduls			CAN-Fehler *)	
	preoperational	operational	stopped	aktueller CAN-Fehler	vergängerer CAN-Fehler
CAN-Error	LED blinkt langsam (ca. 1 Hz, 50%)	LED aus	LED blinkt langsam (ca. 1 Hz, 50%)	LED blinkt schnell (5...10 Hz, 50%)	LED blinkt langsam (ca. 1 Hz, 90% grün)
Module_Status	LED aus	LED leuchtet grün	LED leuchtet grün	beliebig	beliebig
Run	LED blinkt langsam (ca. 1 Hz, 50%)	LED leuchtet grün	LED blinkt langsam (ca. 1 Hz, 50%)	LED blinkt schnell (5...10 Hz, 50%)	LED blinkt langsam (ca. 1 Hz, 90% grün)

*) Die CAN-Fehler-LEDs werden durch einen beliebigen SDO-Zugriff zurückgesetzt

Tabelle 1.2.1: Modul-Zustände und CAN-Fehler in Bezug auf die Status-LEDs

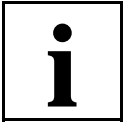
1.3.2 Bedeutung der gelben LEDs

Die Bedeutung der Anzeige der acht gelben LEDs hängt vom aktuellen Zustand des Moduls ab. Die LEDs zeigen entweder den Ein-, bzw. Ausgangspegel der I/Os an oder die aktuelle Einstellung der Kodierschalter-Bits 1...4, bzw. Lötbrücken LB191...LB194 (Node-ID).

Zustand des Moduls	Funktion der gelben LEDs 'Analog/Digital_Input' ... 'Digital_Output'
preoperational	LEDs zeigen Einstellung der Kodierschalter-Bits 1...4
operational	LEDs zeigen Zustand der I/Os
stopped	LEDs zeigen Zustand der I/Os

Tabelle 1.2.2: Status-LEDs und Funktion der gelben LEDs in Bezug auf den Modul-Zustand

Zeigen die gelben LEDs den Zustand der I/Os an, so haben die Leuchtzustände die in der folgend dargestellten Tabelle gezeigte Bedeutung (gültig für Default-Konfiguration 0, näheres hierzu siehe Seite 18).



Übersicht

Die Leuchtzustände der roten LED 'Output_Error' haben immer die unten dargestellte Bedeutung, unabhängig vom aktuellen Zustand des Moduls.

LED-Name	Farbe	Leuchtzustand	Bedeutung
Analog/Digital_Input_1 : Analog/Digital_Input_3	gelb	LED aus	kein Eingangssignal oder Signalpegel zu gering
		LED leuchtet	Eingangssignal liegt an ($U_{in} \geq 2,5V$) und Eingang ist auf PDO 'gemapped'
Analog/Digital_Input_4	gelb	LED aus	Regler nicht aktiv
		LED leuchtet	Regler aktiv (Reglerbeschreibung siehe Software-Manual)
Digital_Output_1 : Digital_Output_4	gelb	LED aus	Ausgang AUS (0 V)
		LED leuchtet	Ausgang EIN (24V)
Output_Error	rot	LED aus	kein Fehler
		LED leuchtet	Fehler des Ausgangstreibers aktiv (Übertemperatur, Strombegrenzung)

Tabelle 1.2.3: Leuchtzustände der I/O-LEDs

Zeigen die gelben LEDs den Zustand der Kodierschalter-Bits 1...4 (Lötbrücke LB191...LB194), so leuchtet die jeweilige LED, wenn das zugeordnete Bit auf '0' gesetzt ist (Kodierschalter auf 'EIN' oder Lötbrücke geschlossen).

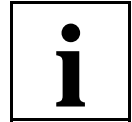
Die folgende Tabelle zeigt die Zuordnung der Hardware-Konfiguration zu den LEDs:

LED	Anzeige der Hardware-Konfiguration
Analog/Digital_Input_4	(bitrate_0) *)
Analog/Digital_Input_3	node_id_2
Analog/Digital_Input_2	node_id_1
Analog/Digital_Input_1	node_id_0

*) Die Bitrate wird durch insgesamt drei Kodierschalter-Bits bestimmt. Die anderen Bits werden nicht angezeigt.

Tabelle 1.2.4: Anzeige der Hardware-Konfiguration

Die Bedeutung der einzelnen Bits des Kodierschalters ist auf Seite 17 ausführlich beschrieben.



1.4 Zusammenfassung der technischen Daten

1.4.1 Allgemeine technische Daten

Temperaturbereich	zulässige Umgebungstemperatur: 0...50 °C
Luftfeuchtigkeit	max. 90%, nicht kondensierend
Betriebsspannung	Nennspannung: 12V, 24V, zulässiger Spannungsbereich: 9 V...27 V, Stromaufnahme (max, bei 20°C): 100 mA
Steckverbinder	P200 (12-pol. MDSTB) - Versorgungsspannung, CAN, I/Os
Gehäuse	Maße: 79 x 90,5 (+10 mm für E/A-Klemme) x 25 mm, für Tragschienenmontage NS35/7,5 DIN EN 50022
Gewicht	ca. 100g

Tabelle 1.3.1: Allgemeine Daten des CAN-Mini-ADI4/DO4

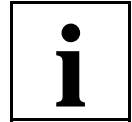


Übersicht

1.4.2 : Controller-Baugruppen

: Controller	C505C, 8 MHz, 8 Bit A/D-Wandler Option: C505CA mit 10-Bit-ADC
Flash-EPROM	zur Speicherung der Firmware, nutzbare Speicherkapazität 64 kByte
I ² C-EEPROM	zur Speicherung der Parameter, 1k Bit (optional: 256/512 Bit)
CAN-Interface	Physical Layer gemäß ISO 11898, keine galvanische Trennung, Anschluß über P200
Übertragungsrate	über Kodierschalter oder Lötbrücken auszuwählen oder programmierbar von 20 kBit/s bis 1 MBit/s
CAN-Identifizier	über Kodierschalter oder Lötbrücken einzustellen und über CAN-Bus programmierbar
LED-Anzeige	4 Status-LEDs für Microcontroller und CAN-Status

Tabelle 1.3.2: Technische Daten der : Controller-Baugruppen



1.4.3 Digitale Ausgänge

Anzahl	4
Treiberbaustein	L6376
Schaltung	P-schaltend, keine galvanische Trennung
Belastbarkeit	max. Strom/Kanal (24V): 0.5 A (25°C)
Versorgungsspannung der Ausgangsschaltung	zulässiger Spannungsbereich: $U_{VCC} = 9,5 \text{ VDC} \dots 27 \text{ VDC}$ Nennwert : $U_{VCC} = 12 \text{ VDC}$ oder 24 VDC
LED-Anzeige	4 gelbe LEDs für den Ausgangsstatus, 1 rote LED zur Fehler-Anzeige

Tabelle 1.3.3: Spezifikation der digitalen Ausgänge

1.4.4 Referenzspannungsausgang

Der Referenzspannungsausgang ist nur verfügbar, wenn die Lötbrücke LB224 gesetzt ist. Der Analog/Digital-Eingang 4 ist dann nicht mehr nutzbar.

Anzahl	1
Spannung	5 V
Verwendung	Ausgang V_{REF} für radiometrische Messungen für IN1...IN3 (Brücken, Potis, etc.)

Tabelle 1.3.4: Spezifikation des Referenzspannungsausgangs



Übersicht

1.4.5 Analog/Digitale Eingänge

Anzahl	4 (Optionen beachten)
Auswertung als analoge Eingänge 1*)	Eingangsspannungsbereich: $U_{IN_{analog}} = 0... 5 \text{ V}$ Eingangsstrom: $I_{IN} \approx 10 : \text{A}$
Auswertung als digitale Eingänge 1*)	Eingangsspannungsbereich: $U_{IN_{digital}} = 0... 30 \text{ V}$ Eingangsstrom: $I_{IN_{max(30V)}} = 25 \text{ mA}$
Option 1a, 1b (gleiche Bestückung)	1a: Schalter-Eingänge (Low-Side-Switch) Schaltstrom: $I_{max} = 0,5 \text{ mA}$ 1b: Widerstandsmessung, $R_{REF} = 10 \text{ k}\Omega$, für NTC, PTC, PT1000 (PT100 nur mit anderem R_{REF})
Option 2	Schalter-Eingänge (High-Side-Switch) Schaltstrom: $I_{max(5V)} = 0,5 \text{ mA}$ Schaltstrom: $I_{max(30V)} = 28,0 \text{ mA}$
Option 3	nur Kanal 'Analog/Digital_In_3': One-Wire-Anschluß (Dallas-I/O-Bausteine)
Option 4	nur Kanal 'Analog/Digital_In_4': Referenzspannungsausgang (siehe Seite 9)
LED-Anzeige	4 gelbe LEDs für den Eingangsstatus

1*) Diese Betriebsarten sind bei Auslieferung des Moduls ohne Änderungen der Lötbrücken oder der Bestückung möglich. Alle als 'Option' aufgeführten Eingangskonfigurationen erfordern zumindest das Setzen von Lötbrücken.

Tabelle 1.3.5: Spezifikation der Analog/Digitalen Eingänge

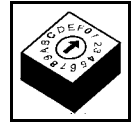


1.5 Bestellhinweise

Typ	Eigenschaften	Bestell-Nr.
CAN-Mini-ADI4/DO4	4 digitale Ausgänge, 4 Analog/Digitale Eingänge	C.2080.01
CAN-Mini-ADI4/DO4-MD (*)	Anwenderhandbuch in deutsch	C.2080.20

(*) Wird das Handbuch gemeinsam mit dem Modul bestellt, so wird es kostenlos mitgeliefert.

Tabelle 1.4.1: Bestellhinweise



2. Hardware-Konfiguration

2.1 Platinenansichten

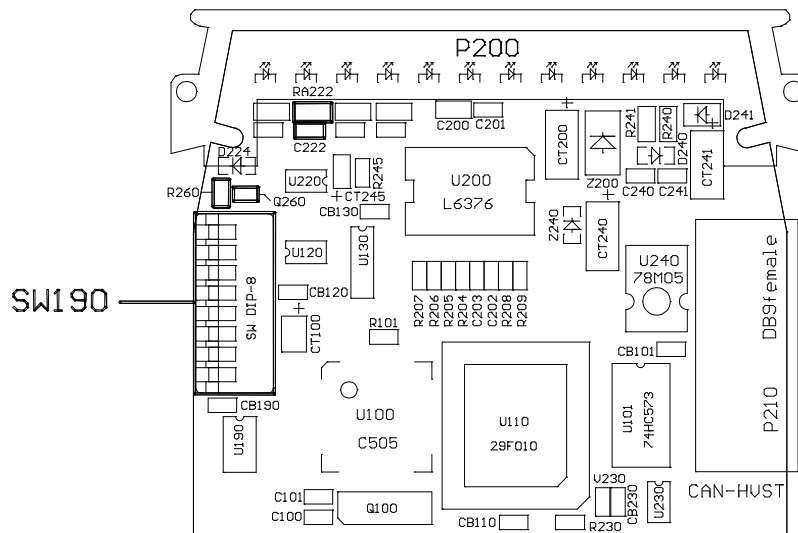


Abb. 2.1.1: Position des Kodierschalters auf der Bestückungsseite der Platine

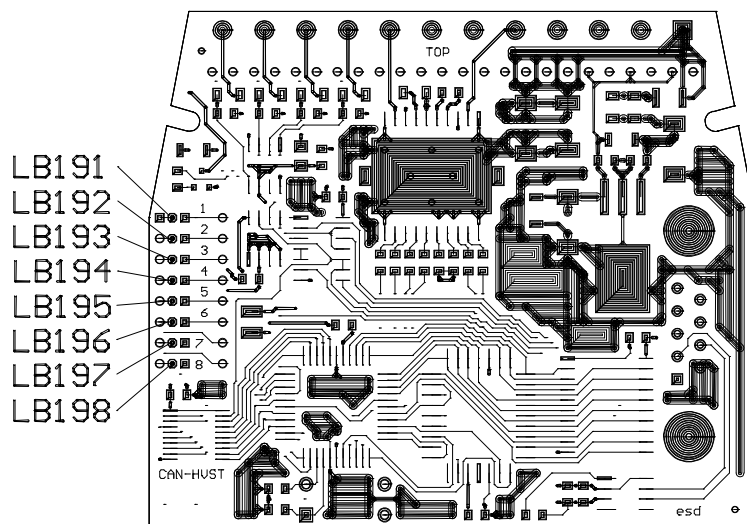


Abb. 2.1.1: Position der Lötbrücken LB191...LB198 (alternativ zu Kodierschalter) auf der Bestückungsseite



Hardware-Konfiguration

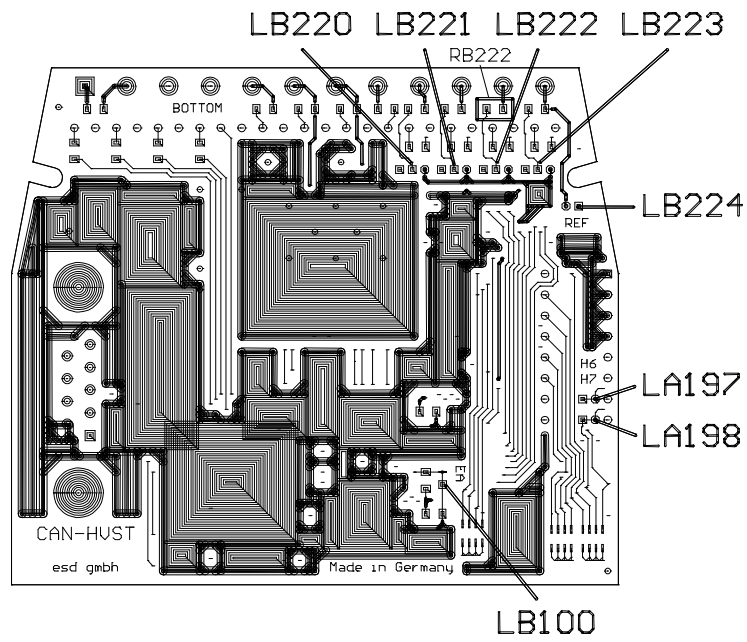
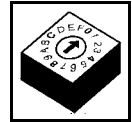


Abb. 2.1.3: Position der Lötbrücken auf der Platinenunterseite



2.2 Default-Einstellung des Kodierschalters und der Brücken

Die jeweilige Default-Einstellung des Kodierschalters und der Brücken bei Auslieferung der Karte ist in den nachfolgenden Abbildungen eingetragen.

Die Anordnung der Bauteile auf der Bestückungsseite kann den Abbildungen 2.1.1 und 2.1.2 entnommen werden. Die Bauteile sind in den anschließenden Beschreibungen so dargestellt, wie sie der Anwender betrachtet, wenn er die Platine mit dem I/O-Stecker nach oben vor sich liegen hat.

Die Lage der Lötbrücken auf der Platinenunterseite kann Abbildung 2.1.3 entnommen werden. Die Lötbrücken sind in den anschließenden Beschreibungen so dargestellt, wie sie der Anwender betrachtet, wenn er die Platine mit dem I/O-Stecker nach oben vor sich liegen hat (Ansicht Platinenunterseite).

Zusammenfassung der Default-Einstellungen bei Auslieferung des Moduls:

Kodierschalter	Funktion	Einstellung
SW190	Bitrate, Node-ID. (=Module-No.)	Bitrate = 1 MBit/s Node-ID = 6 (die Node-ID und die Bitrate müssen in jedem Fall vom Anwender mit dem vorhandenen CAN-Netz abgeglichen werden) Default-Konfiguration = Konfiguration 0

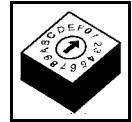
Tabelle 2.2.1: Default-Einstellung des Kodierschalters



Hardware-Konfiguration

Lötbrücke	Funktion	Einstellung
LA197, LA198	Verbindung der Kodierschalter-Bits 7 und 8 mit Treiberbaustein	geschlossen, d.h. diese Bits des Kodierschalters können ausgewertet werden
LB100	Auswahl des Programmspeichers des Microcontrollers	geschlossen, d.h. Microcontroller arbeitet mit externem Flash-EEPROM
LB191...LB198	alternativ zu Kodierschalter: Bitrate, Node-ID (=Module-No)	Bitrate = 1 MBit/s Node-ID = 6 (die Node-ID und die Bitrate müssen in jedem Fall vom Anwender mit dem vorhandenen CAN-Netz abgeglichen werden)
LB220...LB223	Konfiguration der Analog/Digitalen-Eingänge	offen, d.h. die Eingänge können als analoge Spannungseingänge oder digitale Eingänge genutzt werden
LB224	Anschluß der lokalen Referenzspannung an Pin 12 des I/O-Steckers P200	geschlossen, d.h. Referenzspannung an P200 (dadurch Analog/Digital-Eingang 4 nicht nutzbar)

Tabelle 2.2.2: Default-Einstellung der Lötbrücken



2.3 Beschreibung des Kodierschalters und der Brücken

2.3.1 Einstellung der Bitrate, der Node-ID (Module-No.) und der Default-Konfiguration über den Kodierschalter oder die Lötbrücken LB191...LB198

2.3.1.1 Allgemeines

Der Kodierschalter SW190 besitzt 8 Schaltkontakte mit denen die Default-Parameter für die Bitrate, die Node-ID und die Firmware eingestellt werden.

Ist der Kodierschalter nicht bestückt, so kann die Konfiguration über die Lötbrücken LB191 bis LB198 erfolgen. Die Lötbrücken befinden sich unterhalb des Kodierschalters und sind daher nur sichtbar, wenn der Kodierschalter nicht bestückt ist.

Der Kodierschalter, bzw. die Lötbrücken sind wie folgt belegt:

Kodierschalter-Kontakt	Lötbrücke	Belegung	Default-Einstellung	Bedeutung
1	LB191	node_id_0	Off	Node-ID = 6
2	LB192	node_id_1	Off	
3	LB193	node_id_2	Off	
4	LB194	bitrate_0	Off	1 MBit/s
5	LB195	bitrate_1	Off	
6	LB196	bitrate_2	Off	
7	LB197	config_0	Off	Default-Konfiguration 0
8	LB198	config_1	Off	

Tabelle 2.3.1: Belegung des Kodierschalters

Eine geschlossene Lötbrücke entspricht einem eingeschalteten (On) Kodierschalter-Bit.



Hardware-Konfiguration

2.3.1.2 Default-Konfiguration

Mit den Kodierschalter-Bits 7 und 8, bzw. den Lötbrücken LB197 und LB198 wird die Default-Konfiguration der lokalen Firmware ausgewählt. Über die Default-Konfiguration wird z.B. festgelegt, wie die Kodierschalter-Bits 1 bis 6 ausgewertet werden, wann die LEDs leuchten und wie die Parameter der Firmware vorbesetzt werden.

Stellung der Kodierschalter-Kontakte/Lötbrücken		Default-Konfiguration
config_1	config_0	
Off	Off	0 = CAN-HVST
Off	On	1 = CAN-Mini ADI4/DO4
On	Off	2 = CAN-Mini-RSS
On	On	3 = reserviert

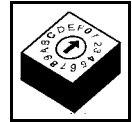
Tabelle 2.3.4: Auswahl der Default-Konfiguration

Merkmale der Default-Konfiguration 0, 1:

- Auswertung der Kodierschalter-Bits 1...6 wie in diesem Kapitel beschrieben
- Leuchtzustände der LEDs, siehe Seite 5
- Firmware:
 - Lokaler 3-Punkt-Regler mit Hysterese
 - CANopen-Implementierung
 - 4 Rx-PDOs, 2 davon vorbesetzt mit analogen Ausgängen und Regler-Parametern
 - 4 Tx-PDOs, 2 davon vorbesetzt mit analogen Eingängen und Regler-Parametern

Merkmale der Default-Konfiguration 2:

- Auswertung der Kodierschalter-Bits 1...6 wie in diesem Kapitel beschrieben
- Leuchtzustände der LEDs, siehe Seite 5
- Firmware:
 - Motor-Steuerung für Rotor-Spannsystem (RSS)



2.3.1.3 Node-ID

Stellung der Kodierschalter-Kontakte/Lötbrücken			Node-ID bei Default-Konfiguration	
node_ID_2	node_ID_1	node_ID_0	0 / 1	2
Off	Off	Off	6	12
Off	Off	On	1	15
Off	On	Off	127	16
Off	On	On	2	18
On	Off	Off	3	19
On	Off	On	7	20
On	On	Off	13	21
On	On	On	14	22

Tabelle 2.3.2: Kodierung der Node-ID (Module-No.)

2.3.1.4 Bitrate

Die folgende Tabelle zeigt die über den Kodierschalter einstellbaren Bitraten.

Die in der Tabelle angegebenen typisch erreichbaren Leitungslängen basieren auf Erfahrungswerten aus der Praxis. Die minimal erreichbaren Längen wurden aus den Angaben der Bauteilhersteller für die 'worst case'-Verzögerungszeiten ermittelt.

Bit-Rate [kBit/s]	Kodierschalter-Kontakte			typische Werte der erreichbaren Leitungslänge l_{max} [m]	minimal erreichbare Leitungslänge l_{min} [m]
	bitrate_2	bitrate_1	bitrate_0		
10	On	On	Off	3600	3600
20	On	Off	On	1400	1400
50	On	Off	Off	710	700
125	Off	On	On	570	550
250	Off	On	Off	270	250
500	Off	Off	On	130	110
800	On	On	On	59	42
1000	Off	Off	Off	37	20

Die Angaben in der Tabelle basieren auf den Grenzwerten des Bit-Timings, den Laufzeiten des lokalen esd-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 unpassende Leitungsabschlüsse, den spezifischen Widerstand, die Geometrie des Kabels oder äußere Störeinflüsse bei der Übertragung nicht miteinbezogen!

Tabelle 2.3.3: Kodierung der Bitrate



Hardware-Konfiguration

2.3.2 Anschluß der Kodierschalter-Bits 7 und 8 an Treiberbaustein über LA197 und LA198

Die Lötbrücken LA197 und LA198 verbinden die Kodierschalter-Bits 7 und 8 mit dem Datentreiberbaustein, der wiederum mit dem Microcontroller verbunden ist.

Die folgende Abbildung zeigt die Schaltung zwischen Kodierschalter, Lötbrücken und Treiberbaustein:

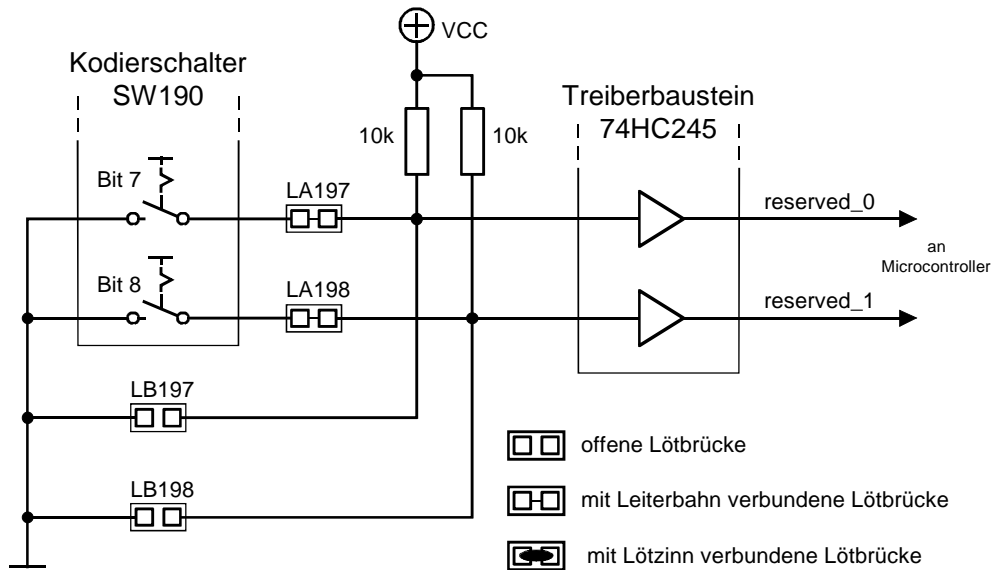
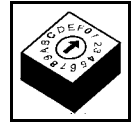


Abb. 2.3.1: Schaltung der Lötbrücken LA197 und LA198

Wie aus der Abbildung zu ersehen ist, kann die Kodierschalterstellung der Bits 7 und 8 nur ausgewertet werden, wenn die Lötbrücken LA197 und LA198 gesetzt sind. Sind die Brücken LA197 und LA198 offen, so ist nur die Stellung der Lötbrücken LB197 und LB198 von Bedeutung.

Bei Auslieferung des Moduls sind die Lötbrücken LA197 und LA198 über eine dünne Leiterbahn zwischen den Pads der Lötbrücken verbunden. Um die Lötbrücke zu unterbrechen, muß diese Leiterbahn mit einem scharfkantigen Gegenstand unterbrochen werden (z.B. mit Schraubendreher oder Messer, Achtung: Verletzungsgefahr!). Mit LötKolben und Lötzinn läßt sich die Brücke wieder schließen. Ist der Kodierschalter nicht bestückt, so ist der Zustand der Lötbrücken LA197 und LA198 ohne Bedeutung.



2.3.3 Konfiguration der Analog/Digitalen-Eingänge über LB220...LB223

Über die Lötbrücken LB220 bis LB223 wird zwischen verschiedenen Eingangsoptionen gewählt. In der Standardeinstellung sind die Brücken offen. In dieser Einstellung können die Eingänge alternativ als analoge oder als digitale Eingänge genutzt werden.

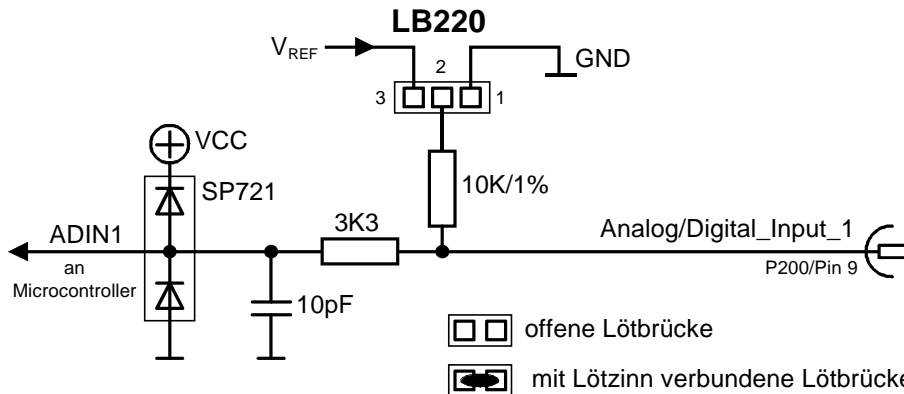


Abb. 2.3.2: Schaltung der Lötbrücken LB220...LB223 am Beispiel von LB220

Eingangskanal	Lötbrücke
1	LB220
2	LB221
3	LB222
4	LB223

Tabelle 2.3.4: Zuordnung der Lötbrücken zu den Eingangskanälen

Stellung der Lötbrücke	Zulässige Eingangskonfigurationen 1*)
offen	a) - analoge Eingangsspannungen 0..5V oder - digitaler Eingang 24V
1-2 gesetzt	b) - High-Side-Switch (Schaltkontakt gegen 24V)
2-3 gesetzt	c) - Low-Side-Switch (Schaltkontakt gegen GND) oder - Widerstandsmessung ($R_{REF}=10\text{ kOhm}$) NTC, PTC, PT1000 - One-Wire-Option (nur Kanal 3)

Tabelle 2.3.5: Einstellung der Eingangskonfiguration über LB220...LB223

1*) Die Kanäle 3 und 4 haben eine Sonderstellung. Für sie sind neben den hier aufgeführten noch zwei weitere Betriebsarten möglich (siehe Kap. Beschreibung der Baugruppen).

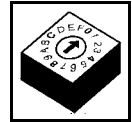


Hardware-Konfiguration

Sollen die vier Kanäle mit unterschiedlichen Eingangskonfigurationen betrieben werden, so ist folgendes zu beachten:

Die Eingangskonfigurationen a) und b) können ohne Einschränkungen miteinander kombiniert werden.

Soll die Eingangskonfiguration c) mit den Konfigurationen a) oder b) kombiniert werden, so muß der Anwender sicherstellen, daß keine Einspeisung von Spannungen $>5V$ erfolgt, da sich sonst der Wert der lokalen Referenzspannung verändern würde. Da in der Konfiguration c) jedoch keine Spannungseinspeisung vorgesehen ist, sondern nur Schalter oder Widerstände gegen GND angeschlossen werden, tritt dieser Fall bei bestimmungsgemäßem Betrieb nicht ein.



2.3.4 Anschluß der internen Referenzspannung an den I/O-Stecker P200 über LB224

Über diese Lötbrücke kann die lokale Referenzspannung nach außen geführt werden. Hierfür wird der Pin 12 des I/O-Steckers P200 verwendet, der sonst vom Analog/Digital-Eingang 4 belegt ist. Wird die Brücke gesetzt, so liegt die Referenzspannung an Pin 12 und der Analog/Digital-Eingang 4 ist nicht mehr nutzbar.

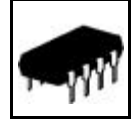
Wird das Modul in der Default-Konfiguration 0 betrieben, so wird das Modul mit gesetzter Lötbrücke LB224 ausgeliefert.

Die Schaltung der Lötbrücke ist in der Darstellung der Eingangsschaltung von Kanal 4 im Kapitel 'Beschreibung der Baugruppen' abgebildet.

2.3.5 Auswahl des Programmspeichers des Microcontrollers über LB100

Der Microcontroller ist auch als OTP-Version (One-Time-Programmable) lieferbar. In diesem Fall wird das externe Flash-EPROM nicht benötigt. Mit der Lötbrücke LB100 wird der Controller-Eingang EA* gesetzt, um festzulegen, mit welchem Speicher der Controller arbeiten soll. Standardmäßig wird ein externes Flash-EPROM verwendet, daher sind die Pads der Lötbrücke über eine dünne Leiterbahn verbunden. Wird die OTP-Version des Controllers eingesetzt, so muß die Lötbrücke offen sein.

Die Stellung dieser Lötbrücke ist durch die Bestückung der Bauteile festgelegt und darf vom Anwender nicht verändert werden!



3. Beschreibung der Baugruppen

3.1 CAN-Bus-Interface

Das CAN-Bus-Interface des CAN-ADI4/DO4-Moduls besteht im wesentlichen aus dem CAN-Transceiver 82C250, der vom internen CAN-Controller des Microcontrollers C505 angesteuert wird. Die CAN-Bus-Signale sind vom Microcontroller nicht galvanisch getrennt!

Der Anschluß der CAN-Bus-Signale erfolgt über den I/O-Stecker P200. Neben den Signalen CAN_L (Pin 4) und CAN_H (Pin 3) ist auch das Bezugspotential GND in der CAN-Leitung mitzuführen. Der GND-Anschluß erfolgt gemeinsam mit der Spannungsversorgung und den I/O-Signalen auf *einer* Klemme des Steckers P200 (Pin 2).

Die Belegung des I/O-Steckers kann dem vorangegangenen Kapitel 'Gehäuseansicht mit LED- und Steckerbeschreibung' auf Seite 4 oder dem Anhang (Seite 31) entnommen werden.

3.2 Digitale Ausgänge

Als Ausgangstreiber wird ein L6376 eingesetzt, der in einem Chip vier Ausgangskanäle integriert. Die Ausgänge werden direkt vom Treiberbaustein auf die Klemmen 5 bis 8 des I/O-Steckers P200 geführt. Die LEDs, die den Zustand der einzelnen Ausgänge anzeigen, sind über einen Vorwiderstand mit dem Ausgangssignal verbunden.

Der Ausgangstreiber besitzt Schutzschaltungen wie Abschaltung bei Unterspannung, Überstrombegrenzung und thermische Abschaltung.

Bei Übertemperatur werden alle Ausgänge abgeschaltet. Ist der Baustein abgekühlt, so werden die Ausgänge automatisch wieder eingeschaltet, sofern sie nicht zwischenzeitlich per Software ausgestellt worden sind.

Bei Erkennen einer zu geringen Treiber-Versorgungsspannung (Unterspannung) werden ebenfalls alle Ausgänge abgeschaltet. Die Ausgänge werden wieder aktiviert, wenn die Versorgungsspannung wieder oberhalb der Schaltschwelle ist, sofern sie nicht zwischenzeitlich per Software ausgestellt worden sind.

Bei Überstrom wird nur der betreffende Kanal abgeschaltet. Die Abschaltung erfolgt nicht sofort, sondern erst nach 12,8 ms, damit auch Lasten geschaltet werden können, die kurzzeitige Überströme verursachen (z.B. kapazitive Lasten, Glühlampen). Überschreitet der Strom weiterhin den Grenzwert der Strombegrenzung, so wird der Ausgang für ca. 820 ms (64 x 12,8 ms) ausgeschaltet, damit sich der Baustein abkühlen kann. Danach wird der Kanal automatisch wieder eingeschaltet. Ist der Strom weiterhin zu groß, so erfolgt wiederum die Abschaltung nach 12,8 ms und der Vorgang wiederholt sich fortlaufend.

Über einen gemeinsamen Fehlerausgang für alle vier Kanäle wird der Fehlerstatus mit einer roten LED (Output Error) angezeigt.



Beschreibung der Baugruppen

3.3 Analog/Digitale Eingänge

3.3.1 Übersicht

Die vier Analog-Digital-Eingänge bieten diverse Konfigurationsmöglichkeiten. Prinzipiell sind hierbei alle vier gleich aufgebaut. Die Kanäle drei und vier verfügen jedoch jeweils über eine weitere Betriebsart.

Die folgende Tabelle zeigt die Lage der vier Kanäle auf dem I/O-Stecker P200 und die möglichen Betriebsarten. Hierbei sind in einem Feld jeweils die Betriebsarten zusammengefaßt, die mit der gleichen Bestückung und Steckbrückenstellung erreicht werden können.

P200-Pin	Eingangskonfiguration			
12	Analog/Digital_In_4	Low-Side-Switch/ Widerstandsmessung_4	High-Side-Switch_4	Ref_Out_5V
11	Analog/Digital_In_3	Low-Side-Switch/ Widerstandsmessung_3	High-Side-Switch_3	One_Wire
10	Analog/Digital_In_2	Low-Side-Switch/ Widerstandsmessung_2	High-Side-Switch_2	-
9	Analog/Digital_In_1	Low-Side-Switch/ Widerstandsmessung_1	High-Side-Switch_1	-

Tabelle 3.3.1: Betriebsarten der Ausgänge mit Zuordnung der Anschlußklemmen

Die folgenden Abbildungen zeigen die drei verschiedenen Schaltungen der vier Kanäle (Kanal 1 und 2 sind gleich aufgebaut).

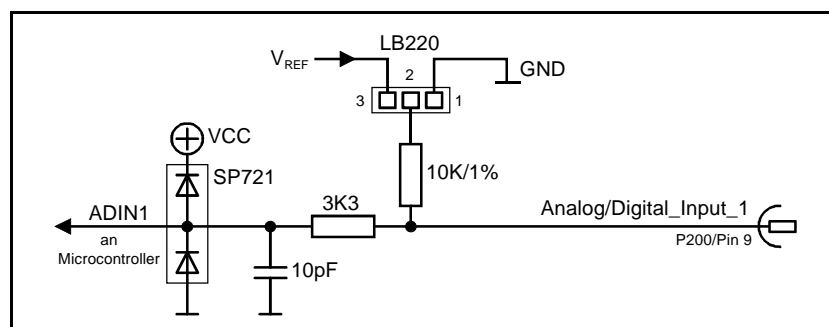


Abb. 3.3.1: Schaltung von Kanal 1 und 2 (Beispiel: Kanal 1)

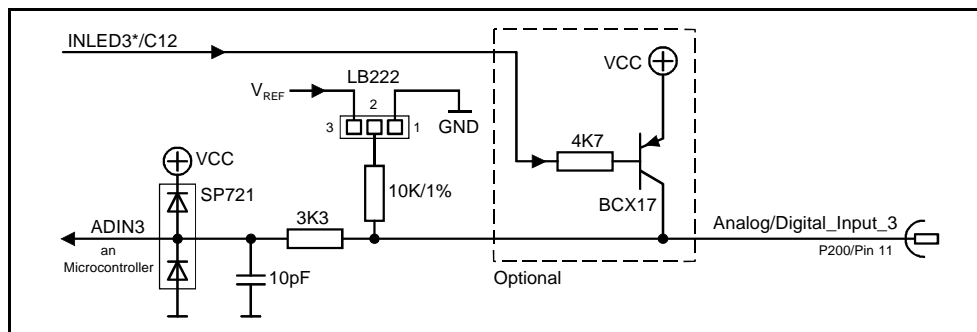


Abb. 3.3.2: Schaltung von Kanal 3

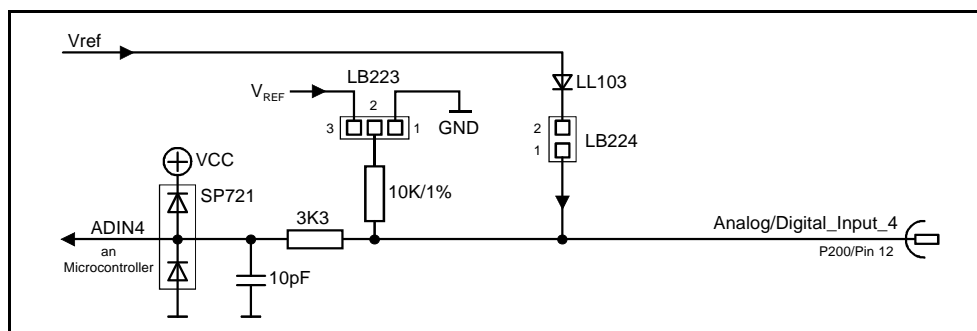


Abb. 3.3.3: Schaltung von Kanal 4

Wie die Lötbrücken für die einzelnen Konfigurationen gesetzt werden müssen, wurde bereits im vorangegangenen Kapitel beschrieben. Im folgenden werden die sich ergebenden Schaltungen nach dem Setzen der Lötbrücken dargestellt.



Beschreibung der Baugruppen

3.3.2 Analog oder digitaler Eingang

Die Schaltung für analoge Eingangsspannungen und für den digitalen Eingang ist gleich. Bei Auslieferung des Moduls ist diese Betriebsart aktiv.

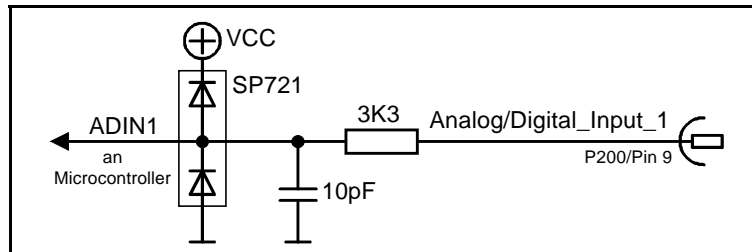


Abb. 3.3.4: Schaltung des analogen und digitalen Eingangs

3.3.3 Low-Side-Switch und Widerstandsmessung

Die Schaltung für den Anschluß eines Low-Side-Switch oder veränderlicher Widerstände (z.B. NTC, PTC, PT1000 [PT100 ist mit diesen Bauteilwerten nicht möglich]) ist gleich. Sie ergibt sich, wenn an der Lötbrücke LB22x (x=0,1,2,3) die Pads 2 und 3 verbunden werden.

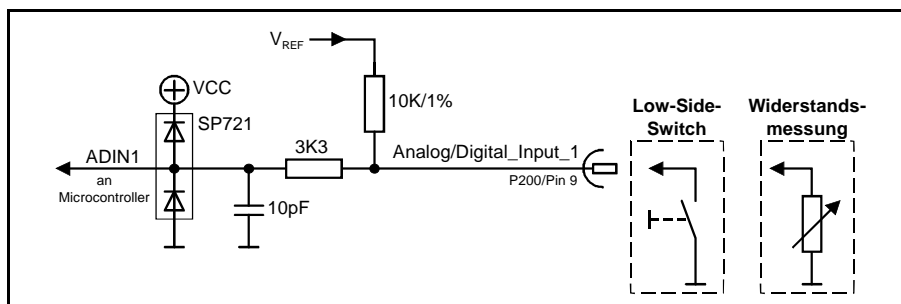


Abb. 3.3.5: Anschluß eines Low-Side-Switch oder veränderlicher Widerstände



3.3.4 High-Side-Switch

Ein High-Side-Switch kann angeschlossen werden, wenn an der Lötbrücke LB22x (x=0,1,2,3) die Pads 1 und 2 verbunden werden.

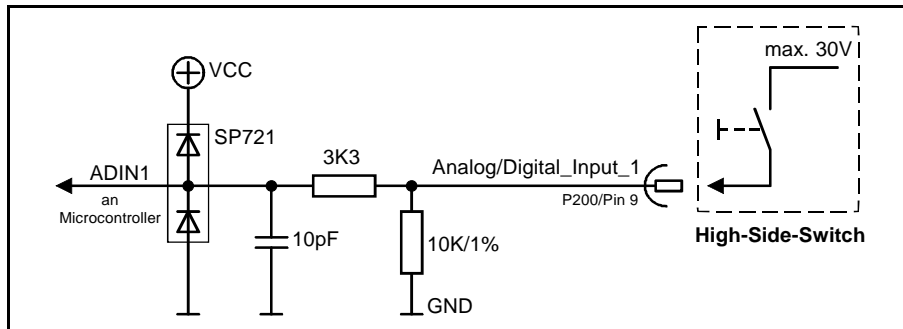


Abb. 3.3.6: Anschluß eines High-Side-Switch

3.3.5 Referenzspannungsausgang

Kanal 4 kann auch als Referenzspannungsausgang genutzt werden. Um ihn zu aktivieren, ist die Lötbrücke LB224 zu setzen. Die Bauteile der bestehenden Eingangsschaltung von Kanal 4 müssen nicht entfernt werden. Der Kanal kann nach Belegung mit der Referenzspannung natürlich nicht mehr als Eingang ausgewertet werden (Schaltung siehe Seite 27).



Beschreibung der Baugruppen

3.3.6 One-Wire-Option

Kanal 3 bietet als weitere Besonderheit die Option, ihn als One-Wire-Schnittstelle zu nutzen. Für diese Option sind jedoch weiterreichende Bestückungsänderungen vorzunehmen. Die Standardbestückung ist wie folgt zu ändern:

Standardbestückung	One-Wire-Option
C222=10nF	C222=nicht bestückt
RA222=2k4 / 5%	RA222=0 Ohm
RB222=10k / 1%	RB222=1k / 5%
R260 = nicht bestückt	R260 =4k7 / 5%
Q260 = nicht bestückt	Q260 =BCX17
Lötbrücke LB222 = offen	Lötbrücke LB222 = 2-3

Tabelle 3.3.2: Bestückungsänderungen für One-Wire-Option

Die Position der Bauteile RA222, C222, R260 und Q260 kann Seite 13 entnommen werden. RA222 und C22 liegen neben dem I/O-Stecker P200 auf der Platinenoberseite. R260 und Q260 liegen zwischen P200 und dem Kodierschalter SW190.

Die Position des Widerstands RB222 kann Seite 14 entnommen werden. Er liegt unterhalb von P200 auf der Platinenunterseite.

Mit den Bestückungsänderungen ergibt sich folgende Schaltung:

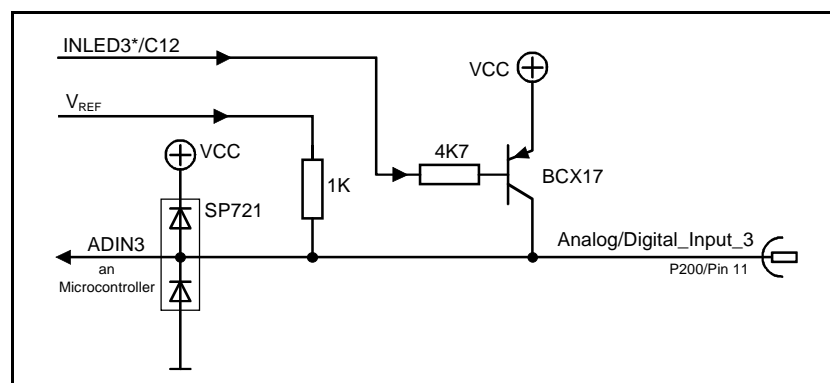
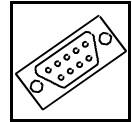


Abb. 3.3.7: Schaltung des One-Wire-Interfaces

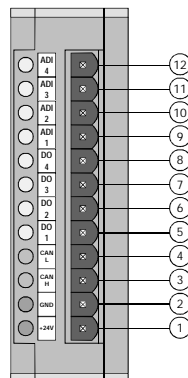


4. Anhang

4.1 Steckerbelegungen

4.1.1 I/O-Steckverbinder X200

Pin-Zuordnung:



Pin-Belegung:

Die Pins 9...12 sind je nach Hardware-Konfiguration verschieden belegt.

Pin	Signal			
12	Analog/Digital_In_4	Low-Side-Switch/ Widerstandsmessung_4	High-Side-Switch_4	Ref_Out_5V
11	Analog/Digital_In_3	Low-Side-Switch/ Widerstandsmessung_3	High-Side-Switch_3	One_Wire
10	Analog/Digital_In_2	Low-Side-Switch/ Widerstandsmessung_2	High-Side-Switch_2	-
9	Analog/Digital_In_1	Low-Side-Switch/ Widerstandsmessung_1	High-Side-Switch_1	-
8	Digital_Out_4			
7	Digital_Out_3			
6	Digital_Out_2			
5	Digital_Out_1			
4	CAN_L			
3	CAN_H			
2	GND			
1	UB (+24V)			

12-pol. MSTB

CAN-Mini-ADI4/DO4

CANopen- Software-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. 205
30165 Hannover

Tel.: 0511/372 98-0
FAX : 0511/372 98-68
E-Mail: info@esd.electronics.com
Internet: www.esd-electronics.com

Handbuch-Datei:	I:\texte\Doku\MANUALS\CAN\Can-Mini\adi4do4\CMIAD12S.ma6
Datum der Druckvorlagenherstellung:	15.12.2000

Beschriebene Software:	CANopen	CAN-Mini Kernel
Revision:	V1.kxx	-

Änderungen in den Kapiteln

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

Kapitel	Änderungen gegenüber Vorversion
-	Änderung der Default-Konfiguration A, B, C und D.
-	-

1. Allgemeines	3
1.1 Begriffsdefinition	3
1.2 NMT-Boot-up	4
1.2.1 Minimum Capability Device Boot-Up	4
1.3 Das CANopen-Objektverzeichnis	4
1.4 Kommunikationsparameter der PDOs	5
1.4.1 Zugriff auf die Kommunikationsparameter über SDO-Telegramme	5
1.4.2 Nichtflüchtiges Speichern von Parametern im EEPROM	7
1.4.3 Wiederherstellen des Default-Zustandes der Parameter	8
2. Communication Profile Area	9
2.1 Verwendete Bezeichnungen und Abkürzungen	9
2.2 Übersicht der Kommunikationsparameter	10
2.3 Beschreibung der Parameter	11
2.3.1 Error Register 1001h	11
2.3.2 Manufacturer Status Register 1002h	12
2.3.3 Manufacturer's Device Name 1008h	13
2.3.4 Manufacturer's HardwareVersion 1009h	14
2.3.5 Manufacturer's Software Version 100Ah	14
2.3.6 Guard Time (100Ch) und Life Time Factor (100Dh)	15
2.3.7 Node Guarding Identifier 100Eh	16
2.3.8 Store Parameters 1010h	17
2.3.9 Restore Default Parameters 1011h	18
2.3.10 Receive PDO Communication Parameter 1400h - 1403h	19
2.3.11 Receive PDO Mapping Parameter 1600h - 1603h	21
2.3.12 Objekt Transmit PDO Communication Parameter 1800h - 1803h	23
2.3.13 Transmit PDO Mapping Parameter 1A00h - 1A03h	25
2.4 Unterstützte Übertragungsarten nach DS-301, Tabelle 55	27
2.5 Auslösen der Sendung von Tx-PDOs	27
2.6 Node Guarding / Life Guarding	28
3. Device Profile Area	29
3.1 Übersicht	29
3.2 Beschreibung der Parameter	30
3.2.1 Lesen der digitalen Eingänge über Object 6000h	30
3.2.2 Lesen der analogen Eingänge über Object 6401h	31
3.2.3 Setzen der analogen Ausgänge über Object 6411h	33
3.2.4 Auswahl des Interrupt-Ereignisses über Object 6421h	34
3.2.5 Ermittlung der Interrupt-Quelle über Object 6422h	35
3.2.6 Enable/disable der Interrupts über Object 6423h	36
3.2.7 Spannungsdifferenz 'Input-Interrupt-Delta' der analogen Eingänge (Object 6426h)	37

Inhalt	Seite
4. Manufacturer Specific Profile Area	39
4.1 Beschreibung der Parameter des Manufacturer Specific Area	39
4.1.1 Parameter-Übersicht 2800h - 2802h	39
4.2 Reglerbeschreibung	40
4.2.1 Parameter <i>Xcycle</i>	40
4.2.2 Sollwert <i>lsoll</i>	42
4.2.3 Istwert <i>list</i>	42
4.2.4 Regler-Differenz <i>ldiff</i>	42
4.2.5 Stellgröße <i>lstell</i>	42
4.2.6 Proportional-Koeffizient <i>Xkpl</i>	43
4.2.7 Hysterese <i>Xshystl</i>	43
4.2.8 Sollwert <i>Xsollwert</i>	44
4.2.9 Regler-Timeout <i>Xreg-prld</i>	44
4.2.10 Verzögerung des Ausgangssignals (<i>outx_mode</i> , <i>t_delayx</i>)	44
4.2.10.1 Ausgangs-Mode <i>outx_mode</i>	45
4.2.10.2 Ausgangs-Verzögerung <i>t_delayx</i>	45
5. PDO-Belegung	47
5.1 Default-Konfiguration 0	47
5.2 Default-Konfiguration 1	48
6. Schnellstart	51
6.1 Konfiguration für den Einsatz im CANopen-Netzwerk	51
6.2 Tabelle der wichtigsten Identifier und Nachrichten für CANopen	53
7. Referenzen	54

1. Allgemeines

Dieses Kapitel enthält einige Grundlagen zum Thema 'CANopen' und die wichtigsten Informationen über die implementierten Funktionen.

Eine komplette CANopen-Beschreibung läge außerhalb des Rahmens dieses Handbuchs. Der interessierte Leser sei auf die nicht ganz einfache CAL / CANopen - Spezifikation (DS-xxx) verwiesen!

1.1 Begriffsdefinition

COB ...	Communication Object	
Emergency-Id...	Emergency Data Object	Notfalldaten
NMT...	Network Management (Master)	Netzwerkmanagement
Rx...	receive	empfangen
SDO...	Service Data Object	Service-Daten (Parameter)
Sync...	Sync(frame) Telegramm	Synchronisations-Identifizier
tbd ...	to be defined	Daten müssen noch festgelegt werden
Tx...	transmit	senden

PDOs (Process Data Objects)

Die PDOs dienen zur Übertragung der Prozeßdaten.

Im 'Receive'-PDO werden Prozeßdaten vom CAN-Mini-ADI4/DO4-Modul empfangen.

Im 'Transmit'-PDO sendet das CAN-Mini-ADI4/DO4-Modul (welches im CANopen-Netz selbst immer SLAVE ist !) Daten auf dem CANopen-Netz.

Die asynchrone Übertragungsart der PDOs wird durch den 'PDO transmission type' im Kommunikationsparameter des entsprechenden Prozeßdatenobjekts definiert.

Die synchrone Übertragung wird zur Zeit nicht unterstützt.

1.2 NMT-Boot-up

1.2.1 Minimum Capability Device Boot-Up

Das CAN-Mini-ADI4/DO4-Modul kann mit dem im CiA-Draft Standard 301 [2] in Kapitel 9.4.2 beschriebenen Minimal-Boot-up initialisiert werden.

Im einfachsten Fall reicht nach dem Einschalten ein Telegramm zum Umschalten vom Zustand *pre-operational* in den Zustand *operational*. Dazu ist an den CAN-Identifizier 0000h z. B. das 2-Byte-Telegramm '0100h' zu senden (= Start Remote Node all Devices).

1.3 Das CANopen-Objektverzeichnis

Das Objektverzeichnis ist im wesentlichen eine (sortierte) Gruppierung von Objekten, auf die über das Netzwerk zugegriffen werden kann. Jedes Objekt in diesem Verzeichnis wird mit einem 16-Bit-Index adressiert. In den Objektverzeichnissen wird der Index in hexadezimaler Form angegeben.

Der Index kann 16-Bit-Parameter nach der CANopen-Spezifikation (CiA-Draft DS-301) oder ein herstellerspezifischer Code sein. Anhand der höherwertigen Bits des Index wird festgelegt, zu welcher Objektklasse der Parameter gehört.

Zum Objektverzeichnis gehören unter anderem:

Index [HEX]	Objekt	Beispiel
0001 ... 009F	Definition von Datentypen	
1000 ... 1FFF	Communication Profile Area	1001 h = Fehler-Register
2000 ... 5FFF	Manufacturer Specific Profile Area (20xxh: esd-Protokoll)	2800 h, 2801 h, 2802 h = Regler (Default-Konfiguration 0)
6000 ... 9FFF	Standardised Device Profile Area	nach Anwendungsprofil DS-401
A000 ... FFFF	reserviert	

1.4 Kommunikationsparameter der PDOs

Die Kommunikationsparameter der PDOs (nach DS-301, Kap.9.2) werden als SDO (Service Data Objects) auf der ID '600h + Node-ID' übertragen (Request). Der Empfänger quittiert die Parameter auf der ID '580h + Node-ID' (Response).

Die Node-ID wird über die Lötbrücken, bzw. den Kodierschalter SW190 eingestellt. Die möglichen Einstellungen sind im Hardware-Handbuch des Moduls ausführlich beschrieben.

1.4.1 Zugriff auf die Kommunikationsparameter über SDO-Telegramme

Die SDOs (Service-Daten-Objekte) dienen zum Zugriff auf das Objektverzeichnis eines Gerätes. Ein SDO stellt daher einen 'Kanal' zum Zugriff auf die Parameter des Gerätes dar. Der Zugriff über diesen Kanal ist beim CAN-Mini-ADI4/DO4-Modul im Zustand *operational* und *pre-operational* möglich.

Die Definitionen für die Zugriffsarten können in der CiA DS-202-2 (CMS-Protocol Specification, Kap. 6: CMS Multiplexed Domain Protocols) entnommen werden.

Dieses Kapitel gibt nicht alle möglichen, sondern nur einige wichtige Übertragungsarten für das CAN-Mini-ADI4/DO4-Modul wieder.

Ein SDO ist wie folgt aufgebaut:

Identifizier	Befehls- code	Index (low)	Index (high)	Subindex	LSB	Datenfeld	MSB
--------------	------------------	----------------	-----------------	----------	-----	-----------	-----

Beispiel:

600h+ Node-ID	23h (write)	03 (Index=1403h) (Receive-PDO-Comm-Para)	14h	01 (COB-def.)	7Fh	04h	00	00
						COB = 047Fh		

Identifizier

Die Parameter werden auf der ID '600h + Node-ID' übertragen (Request).

Der Empfänger quittiert die Parameter auf der ID '580h + Node-ID' (Response).

Befehlscode

Der gesendete Befehlscode setzt sich unter anderem aus dem Command Specifier und der Länge zusammen. Häufig benötigte Kombinationen sind z.B.:

40h = 64 dez: Read Request, d.h. ein Parameter soll gelesen werden

23h = 35 dez: Write Request mit 32 Bit Daten, d.h. ein Parameter soll gesetzt werden

Das CAN-Mini-ADI4/DO4-Modul antwortet auf jedes empfangene Telegramm mit einem Antworttelegramm.

Das Antworttelegramm kann folgende Befehlscodes enthalten:

- 43h = 67 dez: Read Response, dieses Telegramm enthält den gewünschten Parameter
- 60h = 96 dez: Write Response, d.h. ein Parameter wurde erfolgreich gesetzt
- 80h = 128 dez: Error Response, d.h. das CAN-Mini-ADI4/DO4-Modul meldet einen Kommunikationsfehler.

Häufig verwendete Befehlscodes

Die folgende Tabelle gibt eine Übersicht über häufig verwendete Befehlscodes. Die Kommando-Frames müssen immer 8 Datenbytes beinhalten. Hinweise zur Syntax und weitere Befehlscodes sind in der CiA DS-202-2 (CMS-Protocol Specification, Kap. 6: CMS Multiplexed Domain Protocols) nachzulesen.

Kommando	für Anzahl Datenbytes	Befehlscode [HEX]
Write Request (Initiate Domain Download)	1	2F
	2	2B
	3	27
	4	23
Write Response (Initiate Domain Download)	-	60
Read Request (Initiate Domain Upload)	-	40
Read Response (Initiate Domain Upload)	1	4F
	2	4B
	3	47
	4	43
Error Response (Abort Domain Transfer)	-	80

Fehlercodes des SDO-Domain-Transfers

Die folgenden Fehlercodes können zur Anwendung kommen (gemäß CiA Work Draft DS-301, Tabelle 20):

Fehlercode [HEX]	CAN-Mini-ADI4/DO4-Bezeichnung	Erläuterung
0x05040001	SDO_CS_UNKNOWN	falscher Command Specifier
0x06010002	SDO_READ_ONLY	Schreibzugriff ist hier falsch
0x06020000	SDO_WRONG_INDEX	falscher Index
0x06070010	SDO_WRONG_LENGTH	falsche Anzahl Daten-Bytes
0x06070012	SDO_PARA_TO_LONG	Länge des Service-Parameters ist zu groß
0x06070013	SDO_PARA_TO_SHORT	Länge des Service-Parameters ist zu klein
0x06090011	SDO_WRONG_SUBIND	falscher Sub-Index
0x08000000	SDO_OTHER_ERROR	nicht definierte Fehlerursache

Index, Sub-Index

Der Index und der Sub-Index werden in dem folgenden Kapitel 'Kommunikationsprofil' beschrieben.

Datenfeld

Das maximal 4 Byte lange Datenfeld ist grundsätzlich nach der Regel 'niederwertiges Byte zuerst, höherwertiges Byte zuletzt' aufgebaut. Dabei steht das niederwertige Byte **immer** in 'Data 1', bei 16-Bit-Werten steht das höchstwertige Byte (Bits 8...15) in 'Data 2', und bei 32-Bit-Werten steht das MSB (Bits 24...31) in 'Data 4'.

1.4.2 Nichtflüchtiges Speichern von Parametern im EEPROM

Die eingestellten Parameter sind sofort nach der Übergabe aktiv. Das 'nichtflüchtige' Speichern der Parameter erfolgt jedoch nicht automatisch. Es muß über einen Schreibzugriff auf das Objekt 1010h veranlaßt werden und sollte nur erfolgen, wenn sich das Modul im Zustand *pre-operational* befindet.

Der Speicher-Modus wird über den Inhalt des Objektes 1010h angezeigt:

Bit 1 des Objekts 1010h, Subindex 1 ist nicht gesetzt, d.h. das CAN-Mini-ADI4/DO4-Modul speichert die Konfiguration nicht 'automatisch', sondern muß durch Schreiben der Zeichenkette 'save' (**73h 61h 76h 65h**, Reihenfolge aus CAN-Telegramm) in das Objekt 1010h, Subindex 1 extra dazu veranlaßt werden.

Read-Request zur Anzeige des aktuellen Speicher-Modus:

Identifizier	Befehls- code	Index (low)	Index (high)	Subindex	Data 1	Data 2	Data 3	Data 4
600h+ Node-ID	40h (read request)	10h	10h	01	00	00	00	00
Daten werden hier nicht ausgewertet								

Rückmeldung des CAN-Mini-ADI4/DO4-Moduls (bei Default-Einstellung):

Identifizier	Befehls- code	Index (low)	Index (high)	Subindex	Data 1	Data 2	Data 3	Data 4
600h+ Node-ID	4Fh (read response, 1 Byte)	10h	10h	01	01	00	00	00
Speichern der Parameter nur bei Request								

Kommando zum Speichern der Parameter:

Identifizier	Befehls- code	Index (low)	Index (high)	Subindex	Data 1	Data 2	Data 3	Data 4
600h+ Node-ID	23h (write, 4 Bytes)	10	10h	01	73h 's'	61h 'a'	76h 'v'	65h 'e'
= Speichern der Parameter								

1.4.3 Wiederherstellen des Default-Zustandes der Parameter

Der Default-Zustand der Parameter wird in zwei Schritten wiederhergestellt:

1. Parameter 'Restore Default-Parameter' aufrufen : Index 1011h, Subindex 1
2. Modul zurücksetzen über NMT-Service 'Reset': auf Identifizier '0' die Daten 81xx senden (xx steht für Node-ID)

2. Communication Profile Area

2.1 Verwendete Bezeichnungen und Abkürzungen

Folgende Bezeichnungen werden in den Tabellen zur Beschreibung der Kommunikationsparameter verwendet:

PDO-Map	für diesen Sub-Index des PDOs ist PDO-Mapping möglich
Save to EEPROM	der Wert dieses Parameters kann im lokalen EEPROM gespeichert werden
Access Mode	zugelassene Zugriffsarten auf diesen Parameter
ro...	read_only Dieser Parameter kann nur gelesen werden. Schreibzugriffe führen zu einer Fehlermeldung.
const....	constant Dieser Parameter kann vom Anwender nicht verändert werden. Er ist Lesbar. Schreibzugriffe führen zu einer Fehlermeldung.
rw...	read&write Dieser Parameter kann gelesen oder gesetzt werden.
rwr...	read & (write) Dieser Parameter kann gelesen oder gesetzt werden. Der gesetzte Wert wird jedoch von der lokalen Firmware überschrieben.
rw...	read & write Auf diesen Parameter können Lese- oder Schreibzugriffe erfolgen. Der geschriebene Wert wird jedoch ignoriert und nicht übernommen. Es erfolgt keine Fehlermeldung beim Schreibzugriff.
Value Range	Wertebereich des Parameters
Default-Value	Grundeinstellung des Parameters bei der angegebenen Default-Konfiguration. Die Default-Konfiguration wird über die Kodierschalter-Bits 7 und 8, bzw. über die Lötbrücken LB197 und LB198 ausgewählt. (Siehe auch Hardware-Handbuch.)
Name/Description	Name und Kurzbeschreibung des Parameters

2.2 Übersicht der Kommunikationsparameter

Das Format der Kommunikationsparameter kann der CiA DS-301, Kap. 9.1 entnommen werden. Besonderheiten der Implementierung der CAN-Mini-ADI4/DO4 sind in den anschließenden Kapiteln dieses Handbuchs aufgeführt.

Das Modul unterstützt nur die in der folgenden Tabelle aufgeführten Kommunikationsparameter.

Index [HEX]	Name	Subindex	Typ	Zugriff	Default
1000	Device Type	-	Unsigned32	ro	000C0191h
1001	Error Register	-	Unsigned8	ro	Error-Code
1002	Manufacturer Status	-	Unsigned32	ro	Status
1004	Number of PDOs	0,1,2	Unsigned32	ro	4
1008	Manufacturer's Device Name	-	Visible String 1*)	ro	esd_ADIO4 3*)
1009	Manufacturer's Hardware Version (see: DS-301)	-	Visible String 1*)	ro	0,...,3 3*)
100A	Manufacturer's Software Version	-	Visible String 1*)	ro	Vx.xxx/ CanOpen 2*)
100B	Node-ID	-	Unsigned32	ro	f (Kodier- schalter)
100C	Guard Time	-	Unsigned16	rw	0 Sek.
100D	Life Time Factor	-	Unsigned8	rw	0
100E	Node Guarding ID	-	Unsigned32	ro	700h+ Node-ID
1010	Store Parameters	0,1	Unsigned32	rw	--
1011	Restore Default Parameters	0,1	Unsigned32	rw	--
1400 - 1403	Receive PDO Communication Parameter	0,1,2,3,4	PDOCommPar	rw	--
1600 -1603	Receive PDO Mapping Parameter	0,1,2,3,4	PDOMapping	rw	--
1800 - 1803	Transmit PDO Communication Parameter	0,1,2,3,4	PDOCommPar	rw	--
1A00- 1A03	Transmit PDO Mapping Parameter	0,1,2,3,4	PDOMapping	rw	--

ro - Read Only, rw - Read/Write

1*) Achtung: Länge > 4 Bytes, d.h. Upload aufwendiger als bei anderen Parametern (siehe S. 13)

2*) abhängig vom Revisionsstand der Software

3*) abhängig von der Default-Konfiguration

2.3 Beschreibung der Parameter

2.3.1 Error Register 1001h

Das CAN-Mini-ADI4/DO4-Modul nutzt das Error-Register, um Fehlermeldungen anzuzeigen.

INDEX	1001h
Name	error register
Data Type	unsigned 8
Default Value	No

Folgende Bits des Error-Registers werden zur Zeit unterstützt:

Bit	Bedeutung
0	generic
1	over current
2	-
3	-
4	-
5	-
6	-
7	-

Die nicht unterstützten Bits (-) werden immer als '0' zurückgegeben.

Folgende Meldungen sind möglich:

- 00h keine Fehler
- 03h Überstrom der digitalen Ausgänge (= over current & generic error)

2.3.2 Manufacturer Status Register 1002h

INDEX	1002h
Name	manufacturer status register
Data Type	unsigned 32
Default Value	No

Die Bits des Status-Registers sind wie folgt belegt:

Register-Bit (Assembler)	Beschreibung	Pegelzuordnung	
D31 ... D8	Diese Bits werden immer als '0' zurückgegeben.	0 1	reserviert, kann '0' oder '1' sein
D7	New on Bus	0 1	reserviert, kann '0' oder '1' sein
D6	Default wake up	0 1	normaler Start Modul ist mit Default-Parametern aufgewacht
D5	I ² C busy	0 1	kein Zugriff auf EEPROM lokale Software greift auf I ² C-EEPROM zu
D4	EEPROM-Size	0 1	EEPROM größer als 256 Byte EEPROM kleiner/gleich 256 Byte
D3	I ² C-Error	0 1	kein I ² C-Error I ² C-Error
D2	Error on CAN	0 1	kein CAN-Bus-Fehler erkannt CAN-Bus-Fehler
D1	Suspend-Bit	0 1	Modul im Zustand <i>operational</i> Modul im Zustand <i>preoperational</i> oder <i>stopped</i>
D0	Power-Up-Reset	0 1	letzter Reset nicht durch Power-Up ausgelöst letzter Reset durch Power-Up ausgelöst

2.3.3 Manufacturer's Device Name 1008h

INDEX	1008h
Name	manufacturer's device name
Data Type	visible string
Default Value	string: 'esd_ADIO4'

Der Datenaustausch beim Upload des Strings muß wie in der folgenden Tabelle dargestellt, ablaufen (alle Zahlenwerte hexadezimal). In den übergebenen Kommandos müssen immer 8 Datenbytes übertragen werden.

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Bemerkung
Client:	40	08	10	00	00	00	00	00	upload request index 1008h, sub-index=0
ADI4/DO4:	41	08	10	00	09	00	00	00	upload ackn., len = 9
Client:	60	00	00	00	00	00	00	00	segment upload request
ADI4/DO4:	00	65	73	64	5F	41	44	49	'esd_ADI', len = 7
Client:	70	00	00	00	00	00	00	00	-
ADI4/DO4:	1B	4F	34	00	00	00	00	00	'O4', len = 2, ended

Eine ausführliche Beschreibung des Domain Uploads ist der CiA DS-202-2 (CMS-Protocol Specification) zu entnehmen.

2.3.4 Manufacturer's Hardware Version 1009h

INDEX	1009h
Name	manufacturer's hardware version
Data Type	visible string
Default Value	ASCII-string: 0, 1, 2, 3

Hardware Version [ASCII]	Description
0	Module CAN-HVST
1	Module CAN-Mini-ADI4/DO4
2	Module CAN-Mini-RSS
3	reserved

2.3.5 Manufacturer's Software Version 100Ah

INDEX	100Ah
Name	manufacturer's software version
Data Type	visible string
Default Value	string: 'Vk.xxx/CanOpen' 1*)

1*) abhängig vom Revisionsstand der Software

Das Lesen der Software-Version erfolgt ähnlich wie das Lesen des Manufacturer Device Names über das Domain Upload Protokoll. Eine ausführliche Beschreibung des Uploads ist der CiA DS-202-2 (CMS-Protocol Specification) zu entnehmen.

2.3.6 Guard Time (100Ch) und Life Time Factor (100Dh)

Guard Time und Life Time Factor werden zusammen ausgewertet. Die Multiplikation beider Werte ergibt die Life Time. Die Guard Time wird in Millisekunden angegeben.

Beim CAN-Mini-ADI4/DO4-Modul ist das Node-Guarding standardmäßig ausgeschaltet. Die Parameter Guard Time und Life Time Factor sind beide auf '0' gesetzt.

INDEX	100Ch
Name	guard time
Data Type	unsigned 16
Default Value	0 [ms]
Minimum Value	0
Maximum Value	FFFFh (65,535 s)

INDEX	100Dh
Name	life time factor
Data Type	unsigned 8
Default Value	0
Minimum Value	0
Maximum Value	FFh

2.3.7 Node Guarding Identifier 100Eh

Das Modul unterstützt nur 11-Bit-Identifizier. Der Parameter kann nur gelesen werden.

INDEX	100Eh
Name	node guarding identifier
Data Type	unsigned 32
Default Value	700 + Node-ID

Struktur des Parameters:

Bit-No.	Wert	Bedeutung
31 (MSB) 30	-	reserved
29...11	0	immer 0, da keine 29-Bit-ID unterstützt werden
10...0 (LSB)	0 x	Bit 0...10 des Node Guarding Identifiers

2.3.8 Store Parameters 1010h

Mit diesem Kommando werden die Parameter im EEPROM gespeichert. Es wird nur das Kommando 'Save all Parameters' unterstützt.

Beim Schreiben des Index muß die unten angegebene Byte-Folge gesendet werden. Das Lesen des Index gibt Informationen über die implementierten Speicher-Funktionen zurück (näheres hierzu siehe CiA DS-301).

INDEX	1010h
Name	store parameters
Data Type	unsigned 32

Index	Sub Index	PDO-mappable	Save to EEPROM	Access Mode	Value Range	Default Value	Name/Description
1400	0h	no	no	const.	unsigned 8	1h	number of entries
	1h	no	no	rw	unsigned 32	no default, write: 65 76 61 73 h (= ASCII: 'e' 'v' 'a' 's')	save all parameters

2.3.9 Restore Default Parameters 1011h

Mit diesem Kommando werden die bei der Auslieferung aktiven Default-Parameter wieder aktiviert. Alle individuellen Einstellungen, die im EEPROM gespeichert worden sind, gehen verloren. Es wird nur das Kommando 'Restore all Parameters' unterstützt.

Beim Schreiben des Index muß die unten angegebene Byte-Folge gesendet werden. Das Lesen des Index gibt Informationen über die implementierten Restore-Funktionen zurück (näheres hierzu siehe CiA DS-301).

INDEX	1011h
Name	restore default parameters
Data Type	unsigned 32

Index	Sub Index	PDO-mappable	Save to EEPROM	Access Mode	Value Range	Default Value	Name/Description
1011	0h	no	no	const.	unsigned 8	1h	number of entries
	1h	no	no	rw	unsigned 32	no default, write: 64 61 65 6C h (= ASCII: 'd' 'a' 'o' 'l')	load all default parameters

2.3.10 Receive PDO Communication Parameter 1400h - 1403h

Mit dem Objekt 'Receive PDO Communication Parameter 1400h - 1403h' werden die Eigenschaften eines Empfangs-PDOs (Rx-PDO) definiert. Das Modul belegt maximal vier Rx-PDOs.

INDEX	1400h-1403h
Name	receive PDO parameter
Data Type	PDOCommPar

Index	Sub Index	PDO-mappable	Save to EEPROM	Access Mode	Value Range	Default Value (Configuration 0/1)	Name/Description
1400	0h	no	no	const.	unsigned 8	3h	number of entries
	1h	no	yes	rw	unsigned 32	206h	COB-ID used by PDO
	2h	no	yes	rwX	unsigned 8	255d	transmission type
	3h	no	no	const.	unsigned 16	0	inhibit time
1401	0h	no	no	ro	unsigned 8	3h	number of entries
	1h	no	yes	rw	unsigned 32	306h	COB-ID used by PDO
	2h	no	yes	rwX	unsigned 8	254d	transmission type
	3h	no	no	const.	unsigned 16	0	inhibit time
1402	0h	no	no	ro	unsigned 8	3h	number of entries
	1h	no	yes	rw	unsigned 32	406h	COB-ID used by PDO
	2h	no	yes	rwX	unsigned 8	254d	transmission type
	3h	no	no	const.	unsigned 16	0	inhibit time
1403	0h	no	no	ro	unsigned 8	3h	number of entries
	1h	no	yes	rw	unsigned 32	506h	COB-ID used by PDO
	2h	no	yes	rwX	unsigned 8	254d	transmission type
	3h	no	no	const.	unsigned 16	0	inhibit time

Value range refer DS-301, table 10, 11.

Beispiel: Definition eines neuen Receive PDOs mit Index '1401h' auf dem CAN-Identifizier' 035Ah'

Zu übertragen ist die hexadezimale Bytefolge auf ID = 600 + Node-ID.

Byte	Name	Wert	Inhalt
0	Command	23	Write Request
1	Index Low	01	Neues PDO 1401h in Intel-Notation
2	Index High	14	
3	Subindex	01	COB-ID definieren
4	Data 1	5A	COB-ID 035Ah in Intel-Notation
5	Data 2	03	
6	Data 3	00	
7	Data 4	00	

2.3.11 Receive PDO Mapping Parameter 1600h - 1603h

Mit dem Objekt 'Receive PDO Mapping Parameter 1600h - 1603h' kann die Zuordnung der Empfangsdaten zu den Rx-PDOs verändert werden.

INDEX	1600h-1603h
Name	receive PDO mapping
Data Type	PDO Mapping

Die folgende Tabelle zeigt die Belegung der Receive PDO Mapping Parameter für die **Default-Konfiguration 0**:

Index	Sub Index	PDO-mappable	Save to EEPROM	Access Mode	Value Range	Default Value (Configuration 0)	Name/Description
1600	0h	no	yes	rw	unsigned 8	2h	number of entries
	1h	no	yes	rw	unsigned 32	64110310h	write_analog_16-bit, subindex 3= PWM_Out 3
	2h	no	yes	rw	unsigned 32	64110410h	write_analog_16-bit, subindex 4= PWM_Out 4
1601	0h	no	yes	rw	unsigned 8	4h	number of entries
	1h	no	yes	rw	unsigned 32	28010110h	Regler-Sollwert <i>lsoll</i>
	2h	no	yes	rw	unsigned 32	28020310h	Regler-Default-Sollwert <i>Xsollwert</i>
	3h	no	yes	rw	unsigned 32	28020110h	Regler-Kp <i>Xkpl</i>
	4h	no	yes	rw	unsigned 32	28020210h	Regler-Hysterese-Stellausgang <i>Xshyst1</i>
1602	0h	no	yes	rw	unsigned 8	0h	no entries
1603	0h	no	yes	rw	unsigned 8	0h	no entries

Value range refer DS-301, table 10, 11.

Die folgende Tabelle zeigt die Belegung der Receive PDO Mapping Parameter für die **Default-Konfiguration 1**:

Index	Sub Index	PDO-mappable	Save to EEPROM	Access Mode	Value Range	Default Value (Configuration 1)	Name/Description
1600	0h	no	yes	rw	unsigned 8	4h	number of entries
	1h	no	yes	rw	unsigned 32	64110110h	write_analog_16-bit, subindex 1= PWM_Out 1
	2h	no	yes	rw	unsigned 32	64110210h	write_analog_16-bit, subindex 2= PWM_Out 2
	3h	no	yes	rw	unsigned 32	64110310h	write_analog_16-bit, subindex 3= PWM_Out 3
	4h	no	yes	rw	unsigned 32	64110410h	write_analog_16-bit, subindex 4= PWM_Out 4
1601	0h	no	yes	rw	unsigned 8	0h	no entries
1602	0h	no	yes	rw	unsigned 8	0h	no entries
1603	0h	no	yes	rw	unsigned 8	0h	no entries

Value range refer DS-301, table 10, 11.

2.3.12 Objekt Transmit PDO Communication Parameter 1800h - 1803h

Mit diesem Objekt werden die Eigenschaften eines Sende-PDO definiert.

INDEX	1800h-1803h
Name	transmit PDO parameter
Data Type	PDOCommPar

Index	Sub Index	PDO-mappable	Save to EEPROM	Access Mode	Value Range	Default Value (Configuration 0/1)	Name/Description
1800	0h	no	no	const.	unsigned 8	5h	number of entries
	1h	no	yes	rw	unsigned 32	186h	COB-ID used by PDO
	2h	no	yes	rw	unsigned 8	255d	transmission type
	3h	no	no	const.	unsigned 16	0	inhibit time
	4h	no	no	const.	unsigned 8	6	(CMS-Prio-Group)
	5h	no	yes	rw	unsigned 16	0	Cyclic_Tx_Time [ms]
1801	0h	no	no	const.	unsigned 8	5h	number of entries
	1h	no	yes	rw	unsigned 32	286h	COB-ID used by PDO
	2h	no	yes	rw	unsigned 8	254d	transmission type
	3h	no	no	const.	unsigned 16	0	inhibit time
	4h	no	no	const.	unsigned 8	6	(CMS-Prio-Group)
	5h	no	yes	rw	unsigned 16	0	Cyclic_Tx_Time [ms]
1802	0h	no	no	const.	unsigned 8	5h	number of entries
	1h	no	yes	rw	unsigned 32	386h	COB-ID used by PDO
	2h	no	yes	rw	unsigned 8	254d	transmission type
	3h	no	no	const.	unsigned 16	0	inhibit time
	4h	no	no	const.	unsigned 8	6	(CMS-Prio-Group)
	5h	no	yes	rw	unsigned 16	0	Cyclic_Tx_Time [ms]
1803	0h	no	no	const.	unsigned 8	5h	number of entries
	1h	no	yes	rw	unsigned 32	486h	COB-ID used by PDO
	2h	no	yes	rw	unsigned 8	254d	transmission type
	3h	no	no	const.	unsigned 16	0	inhibit time
	4h	no	no	const.	unsigned 8	6	(CMS-Prio-Group)
	5h	no	yes	rw	unsigned 16	0	Cyclic_Tx_Time [ms]

Value range refer DS-301, table 10, 11.

Beispiel: Definition eines neuen Transmit PDOs mit Index '1801h' auf dem CAN-Identifizier' 047Fh'

Zu übertragen ist die hexadezimale Bytefolge auf ID = 600 + Node-ID.

Byte	Name	Wert	Inhalt
0	Command	23	Write Request
1	Index Low	01	Neues PDO 1801h in Intel-Notation
2	Index High	18	
3	Subindex	01	COB definieren
4	Data 1	7F	COB 047Fh in Intel-Notation
5	Data 2	04	
6	Data 3	00	
7	Data 4	00	

2.3.13 Transmit PDO Mapping Parameter 1A00h - 1A03h

Mit dem Objekt ‘**Transmit PDO Mapping Parameter 1A00h - 1A03h**’ kann die Zuordnung der Sendedaten zu den Tx-PDOs verändert werden.

INDEX	1A00h-1A03h
Name	transmit PDO mapping
Data Type	PDO Mapping

Die folgende Tabelle zeigt die Belegung der Transmit PDO Mapping Parameter für die **Default-Konfiguration 0**:

Index	Sub Index	PDO-mappable	Save to EEPROM	Access Mode	Value Range	Default Value (Configuration 0)	Name/Description
1A00	0h	no	yes	rw	unsigned 8	2h	number of entries
	1h	no	yes	rw	unsigned 32	64010210h	read_analog_16-bit, subindex 2= AD-In 2
	2h	no	yes	rw	unsigned 32	64010310h	read_analog_16-bit, subindex 3= AD-In 3
1A01	0h	no	yes	rw	unsigned 8	4h	number of entries
	1h	no	yes	rw	unsigned 32	28010110h	Regler-Sollwert ‘lsoll’
	2h	no	yes	rw	unsigned 32	28010210h	Regler-Istwert ‘list’
	3h	no	yes	rw	unsigned 32	28010310h	Regler-Diff. ‘ldiff’
	4h	no	yes	rw	unsigned 32	28010410h	Regler-Stell ‘lstell’
1A02	0h	no	yes	rw	unsigned 8	0h	no entries
1A03	0h	no	yes	rw	unsigned 8	0h	no entries

Value range refer DS-301, table 10, 11.

Die folgende Tabelle zeigt die Belegung der Transmit PDO Mapping Parameter für die **Default-Konfiguration 1**:

Index	Sub Index	PDO-mappable	Save to EEPROM	Access Mode	Value Range	Default Value (Configuration 1)	Name/Description
1A00	0h	no	yes	rw	unsigned 8	4h	number of entries
	1h	no	yes	rw	unsigned 32	64010110h	read_analog_16-bit, subindex 1= AD-In 1
	2h	no	yes	rw	unsigned 32	64010210h	read_analog_16-bit, subindex 2= AD-In 2
	3h	no	yes	rw	unsigned 32	64010310h	read_analog_16-bit, subindex 3= AD-In 3
	4h	no	yes	rw	unsigned 32	64010410h	read_analog_16-bit, subindex 4= AD-In 4
1A01	0h	no	yes	rw	unsigned 8	1h	number of entries
	1h	no	yes	rw	unsigned 32	60000108h	read_digital_input, In 1...4
1A02	0h	no	yes	rw	unsigned 8	0h	no entries
1A03	0h	no	yes	rw	unsigned 8	0h	no entries

Value range refer DS-301, table 10, 11.

2.4 Unterstützte Übertragungsarten nach DS-301, Tabelle 55

Transmission Type	PDO transmission					Supported by CAN-Mini-ADI4/DO4
	cyclic	acyclic	synchronous	asynchronous	RTR only	
0		X	X			NO
1...240	X		X			NO
241...251	reserved					NO
252			X		X	NO
253				X	X	NO
254				X		YES
255				X		YES

Es werden nur die Übertragungsarten 254 (Manufacturer Specific) und 255 (Profile Specific) verwendet.

2.5 Auslösen der Sendung von Tx-PDOs

Bedingung für das Senden der Tx-PDOs:

1. Es werden nur die Tx-PDOs gesendet, die über das Transmit-PDO-Mapping (Index 1A00h...1A03h) aktiviert worden sind. Dies sind z.B. in der Default-Konfiguration 0 die Tx-PDOs mit den Identifiern 186h und 286h.
2. Es muß ein 'Tx-Request' vorliegen.
Ein Tx-Request liegt vor, wenn mindestens eine der folgenden Bedingungen erfüllt ist:
 - a) Es wurde für diesen Tx-PDO ein RTR-Frame empfangen.
 - b) Für diesen Tx-PDO ist die Cyclic_Tx_Time gesetzt worden und die Zeit zum Senden des Tx-PDOs ist erreicht (über Object Transmit PDO Communication Parameter, Index 1800h...1803h).
 - c) Es liegt ein anwendungsabhängiger Request vor. Dies kann z.B., wie in der Default-Konfiguration 0, die Überschreitung der definierten Meßwertänderung 'Input-Interrupt-Delta' sein.

2.6 Node Guarding / Life Guarding

Zur gegenseitigen Überwachung von Steuerung und den angeschlossenen CAN-Modulen (insbesondere zum Erkennen von Verbindungsausfällen) dient das Node-Life-Guarding.

In der CANopen-Terminologie ist das Node-Guarding die Überwachung der NMT-Slave-Module (hier CAN-Mini-ADI4/DO4) durch den NMT-Master, das Life-Guarding läuft dagegen im NMT-Slave ab und überwacht den NMT-Master.

Node-Guarding:

Der NMT-Master ruft mit einer RTR-Message (CAN-Request) zyklisch ein besonderes Telegramm von seinen NMT-Slaves (hier CAN-Mini-ADI4/DO4) ab. Stimmt die Antwort der Slaves nicht mit der erwarteten Antwort überein, oder erfolgt die Antwort nicht innerhalb der 'guard_time', erkennt der NMT-Master einen Fehler und reagiert entsprechend. Im Antworttelegramm ist der Modulstatus und ein Toggle-Bit enthalten.

Life-Guarding:

Die NMT-Slaves überwachen, ob sie im Rahmen des Node-Guardings vom NMT-Master abgefragt werden. Bleiben diese Abfragen für die Zeitdauer der 'Life Time' aus, so wird ein lokaler RESET ausgelöst.

Die Life Time ist das Produkt aus 'Guard Time' (Index 100Ch) und dem 'Life Time Faktor' (Index 100Dh).

3. Device Profile Area

3.1 Übersicht

Die Datenformate können der CiA DS-301, Kap. 9.1 entnommen werden. Besonderheiten der Implementierung der CAN-Mini-ADI4/DO4 sind in den anschließenden Kapiteln dieses Handbuchs aufgeführt.

Das Modul unterstützt nur die in der folgenden Tabelle aufgeführten Objects.

Index [HEX]	Name	Subindex [HEX]	Typ	Zugriff	Default [HEX]
6000	Digitale Eingänge (Read_Digital_Input)	0	Unsigned 8	ro	-
		1	Unsigned 8	ro	-
6401	Analoge Eingänge (Read_Analogue_Input_16)	0	Unsigned 8	ro	4
		1, 2, 3, 4	Unsigned 16	ro	-
6411	Analoge Ausgänge ^{1*)} (Write_Analogue_Output_16)	0	Unsigned 8	ro	4
		1, 2, 3, 4	Unsigned 16	rw	-
6421	Trigger-Event auswählen (Input_Trigger_Selection)	0	Unsigned 8	ro	4
		1, 2, 3, 4		rw	4
6422	Interrupt-Quelle ermitteln (Interrupt_Source)	0	Unsigned 8	ro	1
		1	Unsigned 32	ro	-
6423	Globale Interrupt-Freigabe (Globally_Interrupt_Enable)	0	Boolean	rw	'0'
6426	Analogwertänderung für IRQ (Input_Interrupt_Delta)	0	Unsigned 8	ro	4
		1, 2, 3, 4	Unsigned 32	rw	-

ro - Read Only, rw - Read/Write, tbd. - to be defined

1*) hier physikalisch als digitale Ausgänge ausgeführt

3.2 Beschreibung der Parameter

3.2.1 Lesen der digitalen Eingänge über Object 6000h

Mit diesem Object werden die Eingangswerte der digitalen Eingänge gelesen. Das CAN-Mini-ADI4/DO4-Modul besitzt 4 Eingänge, die als digitale Eingänge ausgewertet werden können.

INDEX	6000h
Name	Read_Digital_Input

Index	Sub Index	PDO-mappable	Save to EEPROM	Access Mode	Value Range	Default Value (Configuration 1)	Name/Description
6000h	0h	no	no	ro	unsigned 8	1h	number of entries
	1h	yes	no	ro	unsigned 8	none	digital Inputs 1...4

Ein aktiver Eingang wird als '1' übergeben

Parameter-Bit:	7	6	5	4	3	2	1	0
Belegung:	don't care				digital Input 4	digital Input 3	digital Input 2	digital Input 1

3.2.2 Lesen der analogen Eingänge über Object 6401h

Mit diesem Object werden die Eingangswerte der analogen Eingänge gelesen. Die 8-Bit-Werte (mit Microcontroller C505C), bzw. 10-Bit-Werte (mit Microcontroller C505CA) werden jeweils linksbündig im Datenwort übertragen.

Unter Subindex '0' kann die Anzahl der verfügbaren analogen Eingänge gelesen werden. Unter den Sub-Indizes 1...4 können die Analogdaten gelesen werden.

INDEX	6401h
Name	Read_Analogue_Input_16

Index	Sub Index	PDO-mappable	Save to EEPROM	Access Mode	Value Range	Default Value (Configuration 0/1)	Name/Description
6401h	0h	no	no	ro	unsigned 8	4h	number of entries
	1h	yes	no	ro	unsigned 16 (left adjusted) 0000h...7F80h	none	Input_1
	2h	yes	no	ro	unsigned 16 (left adjusted) 0000h...7F80h	none	Input_2
	3h	yes	no	ro	unsigned 16 (left adjusted) 0000h...7F80h	none	Input_3
	4h	yes	no	ro	unsigned 16 (left adjusted) 0000h...7F80h	none	Input_4

Die A/D-Wandler-Bits werden beim **8-Bit-A/D-Wandler** wie folgt im Parameter übergeben:

Parameter-Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
A/D-Wandler-Bit	Sign	7 MSB	6	5	4	3	2	1	0 LSB	die Bits 6...0 werden immer als '0' gelesen						

Die folgende Tabelle zeigt die Zuordnung der gelesenen Werte zu den gemessenen Spannungen beim **8-Bit-A/D-Wandler**:

Gelesener A/D-Wert 8-Bit-A/D-Wandler [HEX]	Gemessene Spannung
0000 : 7F80	0V : 5V-(1 LSB) = 4,980V

Die A/D-Wandler-Bits werden beim **10-Bit-A/D-Wandler** wie folgt im Parameter übergeben:

Parameter-Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
A/D-Wandler-Bit	Sign	9 MSB	8	7	6	5	4	3	2	1	0 LSB	die Bits 4...0 werden immer als '0' gelesen				

Gelesener A/D-Wert 10-Bit-A/D-Wandler [HEX]	Gemessene Spannung
0000 : 7FE0	0V : 5V-(1 LSB) = 4,995V

3.2.3 Setzen der analogen Ausgänge über Object 6411h

Dieses Objekt ist in der CiA DS-401 zum Setzen von analogen Ausgängen vorgesehen. Es wird hier verwendet, um die PWM-fähigen **digitalen Ausgänge** des CAN-Mini-ADI4/DO4 anzusteuern.

Unter Subindex '0' kann die Anzahl der verfügbaren analogen Ausgänge gelesen werden. Unter den Sub-Indizes 1...4 können die digitalen Ausgänge gesetzt werden.

INDEX	6411h
Name	Write_Analogue_Output_16

Index	Sub Index	PDO-mappable	Save to EEPROM	Access Mode	Value Range	Default Value (Configuration 0/1)	Name/Description
6411h	0h	no	no	ro	unsigned 8	4h	number of entries
	1h	yes	yes	rw	unsigned 16 (left adjusted) 8xxxh....0...7FFFh	0	Output_1
	2h	yes	yes	rw	unsigned 16 (left adjusted) 8xxxh....0...7FFFh	0	Output_2
	3h	yes	yes	rw	unsigned 16 (left adjusted) 8xxxh....0...7FFFh	0	Output_3
	4h	yes	yes	rw	unsigned 16 (left adjusted) 8xxxh....0...7FFFh	0	Output_4

Die PWM-Ausgänge werden wie folgt angesteuert:

Parameter-Wert [HEX]	Zustand des digitalen Ausgangs
8xxx	konstant AUS
0000	konstant AUS
:	:
4000	Tastverhältnis 50/50
:	:
7FFF	konstant EIN

3.2.4 Auswahl des Interrupt-Ereignisses über Object 6421h

Mit diesem Object kann ausgewählt werden, bei welchem Ereignis die Sendung der Analogwerte erfolgen soll. Die Auswahl kann für jeden Kanal separat erfolgen.

Zur Zeit wird nur das Senden bei Überschreitung von 'Input-Interrupt-Delta', d.h. einer über Object 6426h zu definierenden Meßwertänderung, unterstützt.

Unter Subindex '0' kann die Anzahl der verfügbaren analogen Ausgänge gelesen werden. Unter den Sub-Indizes 1...4 wird das gewünschte Ereignis ausgewählt.

INDEX	6421h
Name	Interrupt_Trigger_Selection

Index	Sub Index	PDO-mappable	Save to EEPROM	Access Mode	Value Range	Default Value (Configuration 0/1)	Name/Description
6421h	0h	no	no	ro	unsigned 8	4h	number of entries
	1h	no	yes	rw	unsigned 8	4h	IRQ_Event_Input_1
	2h	no	yes	rw	unsigned 8	4h	IRQ_Event_Input_2
	3h	no	yes	rw	unsigned 8	4h	IRQ_Event_Input_3
	4h	no	yes	rw	unsigned 8	4h	IRQ_Event_Input_4

Die DS-401 definiert die IRQ-Events wie folgt:

Bit No.	Interrupt-Trigger	Unterstützt
0	Upper Limit exceeded	-
1	Input below lower limit	-
2	Input changed by more than delta	X
3	Input reduced by more than negative delta	-
4	Input increased by more than positive delta	-
5...7	reserved for future use	-

Beim Auftreten eines Ereignisses werden alle PDOs, in denen das entsprechende Analog-Input-Objekt 'gemapped' ist, gesendet.

3.2.5 Ermittlung der Interrupt-Quelle über Object 6422h

Über dieses Object kann ermittelt werden, welcher analoge Kanal den Interrupt ausgelöst hat.

Unter Subindex '0' kann die Anzahl der folgenden lesbaren Parameter gelesen werden. Unter dem Subindex 1 wird die Interrupt-Quelle zurückgegeben.

INDEX	6422h
Name	Interrupt_Source

Index	Sub Index	PDO-mappable	Save to EEPROM	Access Mode	Value Range	Default Value (Configuration 0/1)	Name/Description
6422h	0h	no	no	ro	unsigned 8	1h	number of entries
	1h	no	no	ro	unsigned 32	none	IRQ_Source_Bank_1

Die niederwertigen vier Bits sind den vier analogen Eingangskanälen zugeordnet. Ist ein Bit auf '1' gesetzt, so ist bei dem zugeordneten Kanal die Interrupt-Bedingung erfüllt.

Die Angabe der Interrupt-Quellen erfolgt akkumulierend, d.h. ist bei einem Kanal die Interruptbedingung bereits erfüllt und tritt die Interrupt-Bedingung bei einem weiteren Kanal ein, so werden die Bits beider Kanäle auf '1' gesetzt.

Mit jedem Lesen des Parameters werden alle Bits auf '0' zurückgesetzt.

Parameter-Bit	31 ... 4	3	2	1	0
A/D-Kanal-Zuordnung	0...0	A/D_IN_4	A/D_IN_3	A/D_IN_2	A/D_IN_1

3.2.6 Enable/disable der Interrupts über Object 6423h

Mit diesem Object können die Interrupts der analogen Eingangskanäle gemeinsam freigeschaltet werden.

INDEX	6423h
Name	Global_Interrupt_Enable

Index	Sub Index	PDO-mappable	Save to EEPROM	Access Mode	Value Range	Default Value (Configuration 0/1)	Name/Description
6423h	0h	no	yes	rw	boolean	≠ 0 (Enabled)	Global_Enable_Interrupts

Parameter-Wert [HEX]	Interrupt-Freigabe
0	IRQ disable
≠ 0	IRQ enable

3.2.7 Spannungsdifferenz 'Input-Interrupt-Delta' der analogen Eingänge (Object 6426h)

Mit diesem Object wird für jeden analogen Eingangskanal die Spannungsdifferenz Input-Interrupt-Delta eingestellt, nach der eine automatische Sendung des Analogwertes erfolgen soll. Ändert sich der gemessene Eingangsspannungswert um mehr als diese Differenz bezogen zum zuletzt gesendeten Meßwert, wird eine Sendung dieses Wertes initiiert.

Unter Subindex '0' kann die Anzahl der verfügbaren analogen Ausgänge gelesen werden. Unter den Sub-Indizes 1...4 können die Werte für Input-Interrupt-Delta übergeben werden.

INDEX	6426h
Name	Input_Interrupt_Delta

Index	Sub Index	PDO-mappable	Save to EEPROM	Access Mode	Value Range	Default Value (Configuration 0/1)	Name/Description
6426h	0h	no	no	ro	unsigned 8	4h	number of entries
	1h	no	yes	rw	unsigned 32	00.40h	Delta_Input_1
	2h	no	yes	rw	unsigned 32	00.40h	Delta_Input_2
	3h	no	yes	rw	unsigned 32	00.40h	Delta_Input_3
	4h	no	yes	rw	unsigned 32	00.40h	Delta_Input_4

Die Differenzspannungswerte werden wie folgt übergeben:

Parameter-Bit	31...16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Differenzspannungswert	immer auf '0' setzen	Sign	14 MSB	13	12	11	10	9	8	7	6	5	4	3	2	1	0 LSB

Ist ein 8-Bit-A/D-Wandler im Einsatz (im Microcontroller C505C), so sind die Bits 6...0 ohne Bedeutung, da das LSB des Wandlerwertes hier Bit 7 entspricht.

Ist ein 10-Bit-A/D-Wandler im Einsatz (im Microcontroller C505CA), so sind die Bits 4...0 ohne Bedeutung, da das LSB des Wandlerwertes hier Bit 5 entspricht.

Die folgende Tabelle zeigt die Zuordnung zwischen gewünschter Spannungsdifferenz und Parameterwert:

Differenzspannung 8-Bit-Wandler	Parameterwert [HEX]
0V	0000 0000
:	:
5V-(1 LSB) = 4,980V	0000 7F80

Differenzspannung 10-Bit-Wandler	Parameterwert [HEX]
0V	0000 0000
:	:
5V-(1 LSB) = 4,995V	0000 7FE0

4. Manufacturer Specific Profile Area

4.1 Beschreibung der Parameter des Manufacturer Specific Area

In diesem Kapitel werden die Reglerparameter beschrieben.

4.1.1 Parameter-Übersicht 2800h - 2802h

Diese Objekte werden von der lokalen Firmware verwendet, um die Regler-Parameter zu speichern und zu übertragen.

Index	Sub Index	PDO-mappable	Save to EEPROM	Access Mode	Value Range	Default Value (Configuration 0/1)	Name/Description
2800	0h	no	no	const.	unsigned 8	1h	number of entries
	1h	no	yes	rw	unsigned 16	000Ah	<i>Xcycle</i> , PWM-resol., Regler- Zyklus, CANXferMode
2801	0h	no	no	const.	unsigned 8	10h	number of entries
	1h	yes	no	rw	unsigned 16	(Xsollwert)	<i>Isoll</i> aktueller Regler-Sollwert
	2h	yes	no	rwr	unsigned 16	none	<i>Ilist</i> aktueller Regler-Istwert
	3h	yes	no	rwr	unsigned 16	none	<i>ldiff</i> aktueller Regler-Differenz
	4h	yes	no	rwr	unsigned 16	none	<i>Istell</i> aktueller Regler-Stellwert
	5h...10h	yes	no	rw	unsigned 16	0	reserved
2802	0h	no	no	const.	unsigned 8	10h	number of entries
	1h	yes	yes	rw	unsigned 16	0100h	<i>Xkp1</i> Kp für Regler-Fixed-Point
	2h	yes	yes	rw	unsigned 16	00FFh	<i>Xshyst1</i> Hysterese Stellausgang
	3h	yes	yes	rw	unsigned 16	FFFFh	<i>Xsollwert</i> Default-Sollwert für Regler
	4h	yes	yes	rw	unsigned 16	13333h	<i>Xreg-prld</i> Regler-Timeout
	13h... 16h	yes	yes	rw	unsigned 8	0	<i>outx_mode</i> Mode Ausgang
	17h... 20h	yes	yes	rw	unsigned 8	0	<i>t_delayx</i> Delay Ausgang

4.2 Reglerbeschreibung

Die folgende Abbildung zeigt den Aufbau des lokal programmierten Reglers:

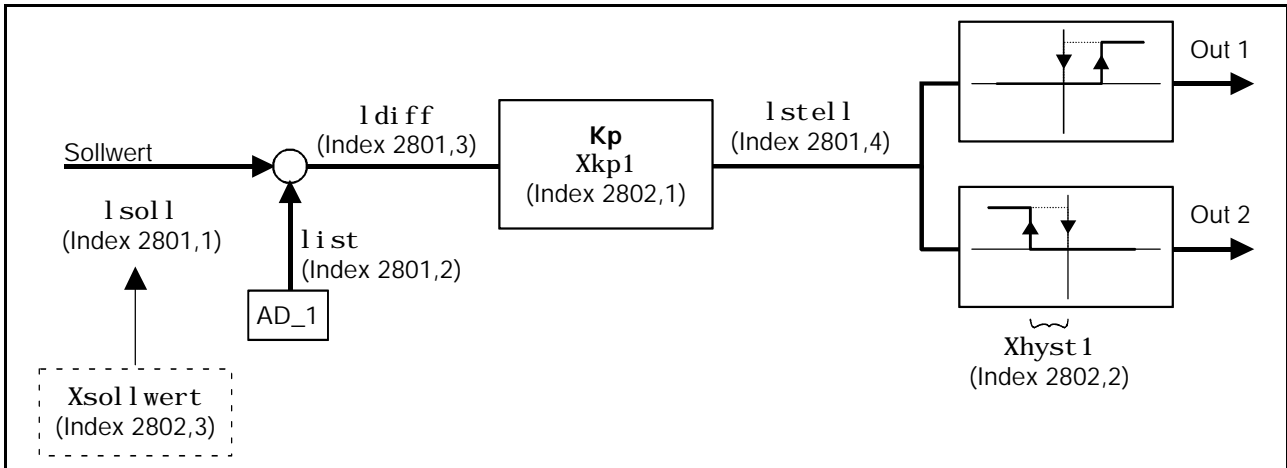


Abb 4.2.1: Reglerstruktur

4.2.1 Parameter Xcycle

Der Parameter Xcycle dient zur globalen Einstellung der System-Parameter.

Der Parameter ist wie folgt aufgebaut:

MSB														LSB	
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
x	x	x	m	p	p	p	p	r	r	r	r	r	r	r	r
keine Auswertung			Datenformat	Periodendauer der PWM-Signale				Zykluszeit des Reglers							

rrrrrrrr.... Zykluszeit des Reglers
 Wertebereich: 0...255d
 0: Regler disabled
 Default-Wert: 10 => 1.5 msec Zykluszeit

$$T_{\text{Zyklus}} = \text{rrrrrrrr} * 0.15\text{msec} [0.15 .. 38.25 \text{ ms}]$$

pppp

Auflösung/Periodendauer der PWM-Signale

Werteberich: 0...15d

0: 16-Bit PWM, $T_{\text{Periode}} = 49.152 \text{ msec}$

Default-Wert: 0

$$T_{\text{Periode}} = 0.75: \text{sec} * 2^{(16-\text{pppp})}$$

Bei pppp ungleich 0 ist nur ein PWM_Betrieb der Kanäle 2...4 möglich !

Die folgende Tabelle zeigt die sich ergebenden Periodenzeiten für verschiedene Werte von pppp:

pppp [hex]	PWM	T_{Periode} [ms]
0	16-Bit-PWM	49.152
1	15-Bit-PWM	24.576
2	14-Bit-PWM	12.288
3	13-Bit-PWM	6.144
4	12-Bit-PWM	3.072
5	11-Bit-PWM	1.536
6	10-Bit-PWM	0.768
7	9-Bit-PWM	0.384
8	8-Bit-PWM	0.192
9	7-Bit-PWM	0.096
A	6-Bit-PWM	0.048
B	5-Bit-PWM	0.024
Die folgenden Parametrierungen sind möglich, aber durch die Slew_Rate der Ausgangstreiber nicht sinnvoll, weil eine Verzerrung der Signale erfolgen würde:		
C	4-Bit PWM	0.012
D	3-Bit-PWM	0.006
E	2-Bit-PWM	0.003
F	1-Bit-PWM	0.0015

Tabelle 3.4.1: Periodendauer der PWM-Signale

m: Auswahl des PDO-Datenformates zwischen Little-Endian- (Intel) und Big-Endian- (Motorola) Format.

0: PDO_daten lsb_first (Little-Endian/Intel-Format): gemäß CanOpen-Spec.

1: PDO_daten msb_first (Big-Endian/Motorola-Format): esd_Mode

Default: 0

4.2.2 Sollwert *lsoll*

Mit *lsoll* wird der Sollwert des Reglers übergeben.

Datenformat: Signed-Integer(16)
Wertebereich: 8000h...0...7FFFh (entspricht -32768d ... 32767d)
für Sollwerte < 0 wird der Regler disabled
Default: (Xsollwert)

4.2.3 Istwert *list*

In *list* ist der Istwert des Reglers abgelegt.

Datenformat: Signed-Integer(16)
Wertebereich: 0000...7F80h (8-Bit A/D-Wandler)
0000...7FE0h (10-Bit A/D-Wandler)
Default: kein Default-Wert

4.2.4 Regler-Differenz *ldiff*

In *ldiff* ist die Regler-Differenz abgelegt:

$$ldiff = lsoll - list$$

Datenformat: Signed-Integer(16)
Wertebereich: 8000h...0...7FFFh
Default: kein Default-Wert

4.2.5 Stellgröße *lstell*

In *lstell* ist die Stellgröße des Reglers abgelegt:

$$lstell = Xkp1 * ldiff$$

Datenformat: Signed-Integer(16)
Wertebereich: 8000h ... 0 ... 7FFFh
Überschreitet das Produkt den gültigen Wertebereich, so wird die Stellgröße auf +/-Max begrenzt.
Default: kein Default-Wert

4.2.6 Proportional-Koeffizient *Xkp1*

Mit *Xkp1* wird der Proportional-Koeffizient des Reglers übergeben.

Datenformat: Signed-Fixed-Point(16)
 Value-Float = $Xkp1 / 256$ (High-Byte wird bei der Umrechnung in Dezimalzahl als Zahl vor dem Komma und Low-Byte als Zahl nach dem Komma ausgewertet.)
 Wertebereich: 8001h...0...7FFFh
 Default: 0100h => 1.00000

Xkp1 [hex]	Xkp1 [dez]
80.01h	-127.99609
...	...
FF.FFh	-0.00391
00.00h	0.00000
00.01h	0.00391
00.02h	0.00781
...	...
00.FFh	0.99609
01.00h	1.00000 (Default)
...	...
7F.FFh	127.99609

Tabelle 4.4.1: Xkp1-Umrechnung

4.2.7 Hysterese *Xshyst1*

Mit *Xshyst1* wird die Hysterese für das Setzen der Ausgänge übergeben (3-Punkt Regler mit Hysterese).

Datenformat: Signed-Integer(16)
 Wertebereich: 0000...7FFFh
 Bedingung zum Setzen der Reglerausgänge (siehe auch graphische Darstellung in Abbildung auf Seite 40):
 $Out1 = (Out1 \text{ AND } (Istell > 0)) \text{ OR } (NOT(Out1) \text{ AND } (Istell > Xshyst1))$
 $Out2 = (Out2 \text{ AND } (Istell < 0)) \text{ OR } (NOT(Out2) \text{ AND } (Istell < -Xshyst1))$
 Default: 00FF >> 1 LSB (8-Bit-A/D-Wandler) Hysterese

4.2.8 Sollwert *Xsollwert*

Mit *Xsollwert* wird der Default-Sollwert des Reglers übergeben.

Nach einem RESET wird immer *Xsollwert* nach *lsoll* übertragen. Bei *Xsollwert* < 0 ist also der Regler nach RESET abgeschaltet.

Datenformat: Signed-Integer(16)
Wertebereich: 8000h ... 0 ... 7FFFh
Default: FFFFh => Regler disabled

4.2.9 Regler-Timeout *Xreg-prld*

Xreg-prld ist das parametrierbare Regler-Timeout nach Erreichen des Sollwertes. (Gültig ab Software-Version b)

Um die Beanspruchung des Relais zu minimieren, wird der Regler nach Erreichen des Sollwertes ausgeschaltet.

Datenformat: Unsigned-Integer
Wertebereich: 0000h...FFFFh
Einheit: 150: s/Bit
Default: 13333h (entspricht 2sec.)

4.2.10 Verzögerung des Ausgangssignals (*outx_mode*, *t_delayx*)

Die Variablen *outx_mode* und *t_delayx* verzögern das Ausgangssignal

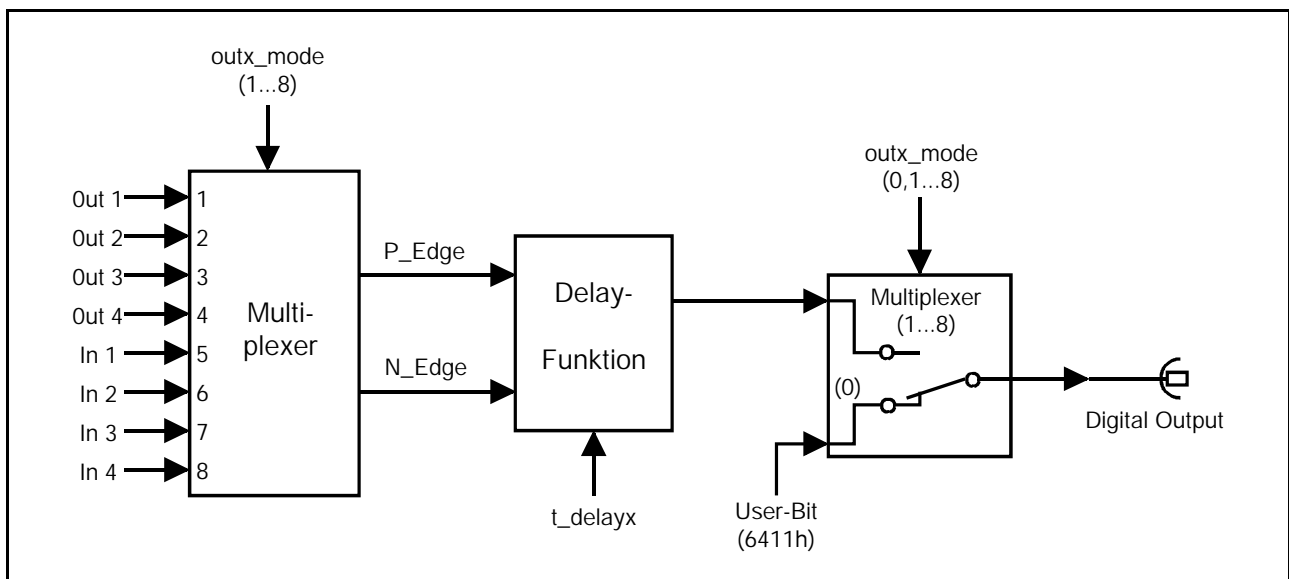


Abb.4.2.10: Funktion der Firmware

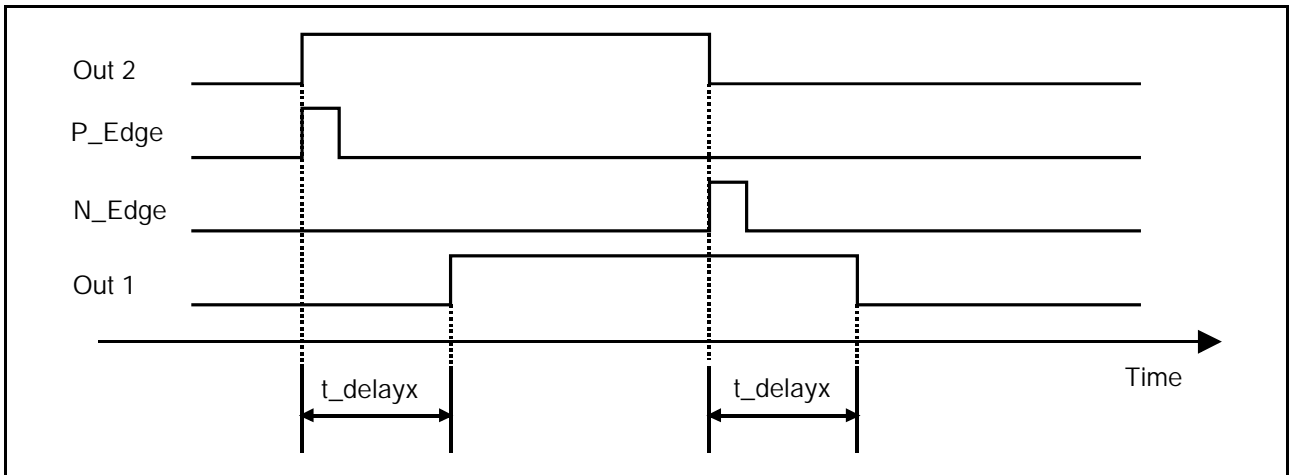


Abb.4.2.11: Verzögerung des Ausgangssignals durch t_delayx
 (Beispiel: Ausgangssignal Out2 wird verzögert auf Out1 ausgegeben)

4.2.10.1 Ausgangs-Mode $outx_mode$

Datenformat: Unsigned-Integer (8)
 Wertebereich: 0,1,...8
 Default: 0

$outx_mode$	Funktion
0	keine Verzögerung
1	Out 1
2	Out 2
3	Out 3
4	Out 4
5	In 1
6	In 2
7	In 3
8	In 4

4.2.10.2 Ausgangs-Verzögerung t_delayx

Datenformat: Unsigned-Integer (16)
 Wertebereich: 0000h...FFFFh
 Einheit: ms
 Default: 0

5. PDO-Belegung

Das Modul belegt bei Auslieferung die im folgenden angegebenen PDOs. Die Node-ID wird über Kodierschalter-Kontakte 1...3, bzw. über die Lötbrücken LB191..LB193 eingestellt (in der Default-Konfiguration 0 ist die Node-ID bei Auslieferung auf '6' eingestellt).

5.1 Default-Konfiguration 0

Alle folgenden Angaben sind nur für die Default-Konfiguration 0 gültig. Diese Konfiguration ist aktiv, wenn die Kodierschalter-Kontakte 7 und 8 und die Lötbrücken LB197 und LB198 offen sind.

CAN-Identifizier	Länge	Übertragungsrichtung
200h + Node-ID	4 Bytes	zur CAN-Mini-ADI4/DO4 (Rx/Empfangs-PDO)
300h + Node-ID	8 Bytes	
400h + Node-ID	-	disabled, no mapping
500h + Node-ID	-	disabled, no mapping
180h + Node-ID	4 Bytes	von CAN-Mini-ADI4/DO4 (Tx/Sende-PDO)
280h + Node-ID	8 Bytes	
380h + Node-ID	-	disabled, no mapping
480h + Node-ID	-	disabled, no mapping

Die acht Datenbytes der Rx-PDOs sind wie folgt belegt (mit Node-ID = 6):

Rx-PDO	Datenbytes							
	1	2	3	4	5	6	7	8
206h	Analog_Out_3 1*)		Analog_Out_4 1*)		keine Auswertung		keine Auswertung	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB
306h	Regler-Sollwert <i>lsoll</i>		Regler-Default-Sollwert <i>Xsollwert</i>		Regler-Kp <i>Xkpl</i>		Regler-Hysterese Stellausgang <i>Xshystl</i>	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB
406h	keine Auswertung		keine Auswertung		keine Auswertung		keine Auswertung	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB
506h	keine Auswertung		keine Auswertung		keine Auswertung		keine Auswertung	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB

1*)... die analogen Ausgänge werden als digitale Ausgänge gesetzt.

Das Datenformat ist jeweils 'Signed-16', 'Little-Endian' (Parameter *Xcycle*, m=0).

Die acht Datenbytes der Tx-PDOs sind wie folgt belegt:

Tx-PDO	Datenbytes							
	1	2	3	4	5	6	7	8
186h	Analog_In_2		Analog_In_3		keine Sendung		keine Sendung	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB
286h	Regler-Sollwert <i>lsoll</i>		Regler-Istwert <i>list</i>		Regler-Differenz <i>ldiff</i>		Regler-Stellgröße <i>lstell</i>	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB
386h	keine Sendung		keine Sendung		keine Sendung		keine Sendung	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB
486h	keine Sendung		keine Sendung		keine Sendung		keine Sendung	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB

5.2 Default-Konfiguration 1

Alle folgenden Angaben sind nur für die Default-Konfiguration 1 gültig. Diese Konfiguration ist aktiv, wenn die Kodierschalter-Kontakte 7 und 8 und die Lötbrücken LB197 und LB198 offen sind.

CAN-Identifizier	Länge	Übertragungsrichtung
200h + Node-ID	8 Bytes	zur CAN-Mini-ADI4/DO4 (Rx/Empfangs-PDO)
300h + Node-ID	-	disabled, no mapping
400h + Node-ID	-	disabled, no mapping
500h + Node-ID	-	disabled, no mapping
180h + Node-ID	8 Bytes	von CAN-Mini-ADI4/DO4 (Tx/Sende-PDO)
280h + Node-ID	1 Byte	
380h + Node-ID	-	disabled, no mapping
480h + Node-ID	-	disabled, no mapping

Die acht Datenbytes der Rx-PDOs sind wie folgt belegt (mit Node-ID = 6):

Rx-PDO	Datenbytes							
	1	2	3	4	5	6	7	8
206h	Analog_Out_1 1*)		Analog_Out_2 1*)		Analog_Out_3 1*)		Analog_Out_4 1*)	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB
306h	keine Auswertung		keine Auswertung		keine Auswertung		keine Auswertung	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB
406h	keine Auswertung		keine Auswertung		keine Auswertung		keine Auswertung	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB
506h	keine Auswertung		keine Auswertung		keine Auswertung		keine Auswertung	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB

1*)... die analogen Ausgänge werden als digitale Ausgänge gesetzt.

Das Datenformat ist jeweils 'Signed-16', 'Little-Endian' (Parameter *Xcycle*, m=0)

Die acht Datenbytes der Tx-PDOs sind wie folgt belegt:

Tx-PDO	Datenbytes							
	1	2	3	4	5	6	7	8
186h	Analog_In_1		Analog_In_2		Analog_In_3		Analog_In_4	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB
286h	Digital_In_1	keine Sendung	keine Sendung		keine Sendung		keine Sendung	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB
386h	keine Sendung		keine Sendung		keine Sendung		keine Sendung	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB
486h	keine Sendung		keine Sendung		keine Sendung		keine Sendung	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB

6. Schnellstart

6.1 Konfiguration für den Einsatz im CANopen-Netzwerk

Für einen schnellen Start mit der einfachsten Konfiguration sind zunächst die folgenden Schritte nötig:

Ausführung	LED-Zustand
1. CAN-Verdrahtung aufbauen (Abschlüsse nicht vergessen!)	alle aus
2. Sofern Default-Einstellung nicht gewünscht: Baudrate mit Kodierschalter in Frontplatte (bzw. Lötbrücken) setzen	alle aus
3. Sofern Default-Einstellung nicht gewünscht: Modul-Nr. mit Kodierschalter in Frontplatte (bzw. Lötbrücken) setzen	alle aus
4. Modul einschalten (sofern nicht unter 2. oder 3. erfolgt) (= Versorgungsspannung anlegen)	Solange sich das Modul im Zustand <i>pre-operational</i> befindet, wird dies durch Blinken der LEDs angezeigt ('Run' blinkt grün, 'CAN-Error' rot). Die Input/Output-LEDs zeigen Status der Kodierschalter-Bits 1...4 an.
5. Beginn der Kommunikation auf dem CAN-Bus gemäß der folgenden Tabelle (siehe Seite 52)	-

zu 5:

Jetzt müssen folgende Frames übertragen werden (Beispiel-Anwendung):

Nr.	Datenrichtung von CAN-Mini- ADI4/DO4- Modul betrachtet	Identifizier	RTR- Bit	Daten- Länge	Daten data 1... ...data8 [hex]	Bemerkung
0	Tx	0x706	0	1	00	ADI4/DO4 meldet sich an
1	Rx	0x000	0	2	0100	Startkommando an alle
2	Tx	0x206	1	0	-	ADI4/DO4 fordert Setzen der Analogwerte an
3	Tx	0x306	1	0	-	ADI4/DO4 fordert Setzen der Reglerwerte an
4	Rx	0x306	0	4	00 00 00 00	Regler-Sollwert wird auf '0' gesetzt
5	Rx	0x286	1	0	-	Anforderung der aktuellen Reglerwerte
6	Tx	0x286	0	8	00 00 00 00 00 00 00 00	ADI4/DO4 sendet aktuelle Reglerwerte
7	Rx	0x306	0	4	00 01 00 00	Regler-Sollwert wird auf 0100h gesetzt
8	Rx	0x286	1	0	-	Anforderung der aktuellen Reglerwerte
9	Tx	0x286	0	8	00 01 00 00 00 01 00 00	ADI4/DO4 sendet aktuelle Reglerwerte

6.2 Tabelle der wichtigsten Identifier und Nachrichten für CANopen

CAN-Identifier [HEX]	Bezeichnung	Länge	Daten [HEX]	Bemerkungen
0	NMT	2	02 xx	Modul geht in Zustand <i>stopped</i>
0	NMT	2	01 xx	Start (Modul geht in Zustand <i>operational</i>)
0	NMT	2	80 xx	Modul geht in Zustand <i>preoperational</i>
0	NMT	2	81 xx	Reset CAN-Mini-ADI4/DO4-Module
0	NMT	2	82 xx	Reset Communication (hier gleiche Funktion wie 81xx)
700h + Node-ID	NMT	8 Bytes	Identifier	NMT error control identifier (see DS-301)
580h + Node-ID	SDO	8 Bytes	Parameter	Quittierung der Kommunikationsparameter von CAN-Mini-ADI4/DO4 (Tx)
600h + Node-ID	SDO	8 Bytes	Parameter	Übertragung der Kommunikationsparameter zur CAN-Mini-ADI4/DO4 (Rx)
200h + Node-ID	Rx_PDO_1	0...8 Bytes	Nutzdaten	zur CAN-Mini-ADI4/DO4 (Rx/Empfangs-PDO)
300h + Node-ID	Rx_PDO_2			
400h + Node-ID	Rx_PDO_3			
500h + Node-ID	Rx_PDO_4			
180h + Node-ID	Tx_PDO_1	0...8 Bytes	Nutzdaten	von CAN-Mini-ADI4/DO4 (Tx/Sende-PDO)
280h + Node-ID	Tx_PDO_2			
380h + Node-ID	Tx_PDO_3			
480h + Node-ID	Tx_PDO_4			

xx = Node-ID

Node-ID. = 1...7Fh (Default bei Konfiguration 0: Node-ID = 6)

7. Referenzen

- [1] CiA DS-202-2 CAN Application Layer for Industrial Applications (02.1996)
- [2] CiA DS-301 CANopen Application Layer and Communication Profile V4.01 (01.06.2000)
- [3] CiA DS-401 CANopen Device Profile for Generic I/O Modules V2.0 (12.1999)