

# CAN-AI8

**Dok-Rev. 2.2 vom 20.10.2011**  
**Hardware-Rev. 1.0 vom 10.03.2003**

---

---

## **Inhaltsverzeichnis**

<b>1</b>	<b>Allgemeine Hinweise</b> .....	<b>4</b>
1.1	Handhabung	4
1.2	Installation	4
1.3	Erklärung	4
1.4	Reparaturen	4
<b>2</b>	<b>Technische Daten</b> .....	<b>5</b>
2.1	Umgebungsbedingungen	5
2.2	Mechanische Abmessungen	5
2.3	Technische Daten	5
<b>3</b>	<b>Inbetriebnahme</b> .....	<b>6</b>
3.1	Einbau	6
3.2	Frontansicht	6
3.3	Spannungsversorgung	6
3.4	CAN Abschlußwiderstand	6
3.4.1	Analoge Eingänge	7
3.5	Steckverbinder	7
3.5.1	I/O-Anschlüsse	7
<b>4</b>	<b>Generic CAN Firmware</b> .....	<b>9</b>
4.1	Einstellen der Identifier und der Baudrate	9
4.2	Betriebszustand	11
<b>5</b>	<b>CANopen® Firmware</b> .....	<b>12</b>
5.1	Einstellen der Identifier und der Baudrate	12
5.2	Betriebszustand	14
5.3	CANopen® Objekte	14
5.3.1	CANopen® Funktionen	14
5.3.2	AI - Objekte	15
<b>6</b>	<b>Hardwarebeschreibung</b> .....	<b>16</b>
6.1	ST4 – Serielle Schnittstelle	16
<b>7</b>	<b>Kommunikation mit der CAN-AI8</b> .....	<b>17</b>
7.1	Generic Can Kommunikation	17
7.1.1	Zyklisches Versenden der Messwerte	17
7.1.2	Mittelwertbildung	17
7.1.3	Belegung der Identifier	17

---

7.2 CANopen® Kommunikation	20
7.2.1 Implementierte CANopen®-Objekte	20
7.2.2 Kodierung der analogen Werte	20

Revisionsliste:

Rev.	Datum	Na.	Änderung
1.0	16.09.2004	Ko	Erstellung
1.1	14.12.2005	Ko	Überarbeitung
1.2	17.05.2006	Ko	Anpassung an ME-Gehäuse
1.3	01.06.2005	Ko	Es werden Standard Frames (11 Bit Identifier) genutzt
1.4	01.03.2007	Ko	Mittelwertbildung ergänzt
1.5	11.04.2008	Kr	CANopen® Beschreibung
1.6	14.04.2009	Ko	Pt100/Pt1000 Temperaturformat
1.7	21.09.2009	Ko	CANopen® Beschreibung korrigiert
1.8	23.09.2009	Kr	CANopen® Beschreibung erweitert um Objekt – Beschreibung
1.9	22.03.2010	Ko	CAN-Busanschluß liegt auf der Unterseite ergänzt
2.0	13.07.2010	Ha	Redaktionelle Änderungen
2.1	01.03.2011	Ko	® ergänzt
2.2	20.10.2011	Ko	Beschreibung 4..20mA Auswertung ergänzt

---

## **1 Allgemeine Hinweise**

### **1.1 Handhabung**

1. Lesen Sie bitte zuerst sorgfältig diese Dokumentation bevor Sie die Hardware auspacken und einschalten. Sie sparen Zeit und vermeiden Probleme.
2. Beachten Sie bitte die Vorsichtsmaßnahmen bei der Handhabung elektrostatisch gefährdeter Hardware.
3. Wenn die Hardware Batterien enthält, legen Sie sie nicht auf elektrisch leitfähige Unterlagen. Die Batterie könnte kurzgeschlossen werden und Schäden verursachen.
4. Achten Sie bitte darauf, daß der spezifizierte Temperaturbereich nicht verlassen wird.

### **1.2 Installation**

1. Überprüfen Sie, ob alle Jumper entsprechend Ihrer Anwendung gesetzt sind.
2. Schalten Sie die Spannungsversorgung der externen Anschlüsse ab, bevor Sie eine Verbindung herstellen.
3. Wenn Sie sicher sind, daß alle Verbindungen korrekt installiert sind, schalten Sie die Spannungsversorgung ein.

### **1.3 Erklärung**

Wir behalten uns das Recht vor, Änderungen, die einer Verbesserung der Schaltung oder des Produktes dienen, ohne besondere Hinweise vorzunehmen. Trotz sorgfältiger Kontrolle kann für die Richtigkeit der hier gegebenen Daten, Schaltpläne, Programme und Beschreibungen keine Haftung übernommen werden. Die Eignung des Produktes für einen bestimmten Einsatzzweck wird nicht zugesichert.

### **1.4 Reparaturen**

Sollte das Produkt defekt sein, so senden Sie es bitte frei in geeigneter Verpackung mit folgender Beschreibung an uns zurück:

- Fehlerbeschreibung
- Trat der Fehler nur unter bestimmten Bedingungen auf?
- Was war angeschlossen?
- Wie sahen die angeschlossenen Signale aus?
- Garantiereparatur oder nicht?

---

## **2 Technische Daten**

### **2.1 Umgebungsbedingungen**

Umgebungstemperatur (Betrieb)	0-50° C
Umgebungstemperatur (Lagerung)	-20-85° C
rel. Luftfeuchte	max. 95%, nicht kondensierend
Höhe	-300m bis +3000m

### **2.2 Mechanische Abmessungen**

Kartengröße	90 x 85 x 22,5 mm (T x H x B)
Anschlüsse	ME-Steckersatz

### **2.3 Technische Daten**

Versorgungsspannung:	24 Volt DC, 0.1 A
Prozessor	MB90F497
Analogeingänge:	8 Stück, 10 Bit, 0-10 / ±10 Volt optional 0/4-20 mA, PT100, usw.
CAN	1x CAN, 4 polig ME-Stecker
Firmware:	CANopen® (Profil DS402) Generic CAN

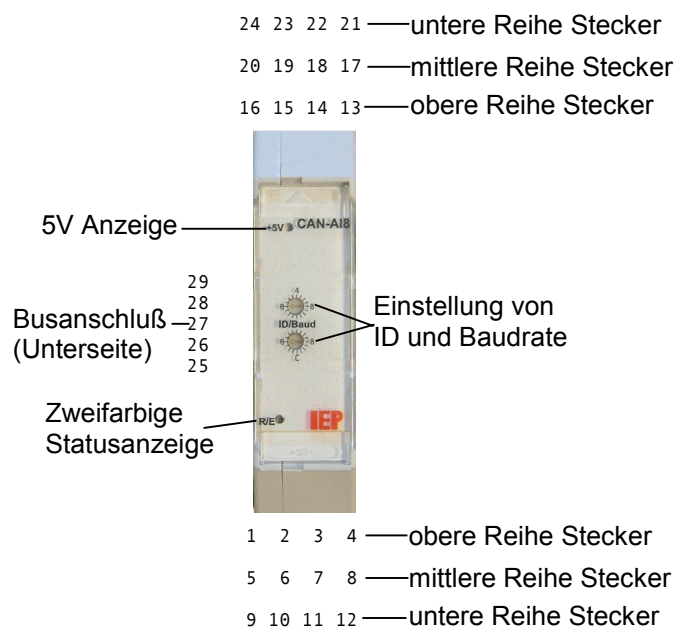
---

### 3 Inbetriebnahme

#### 3.1 Einbau

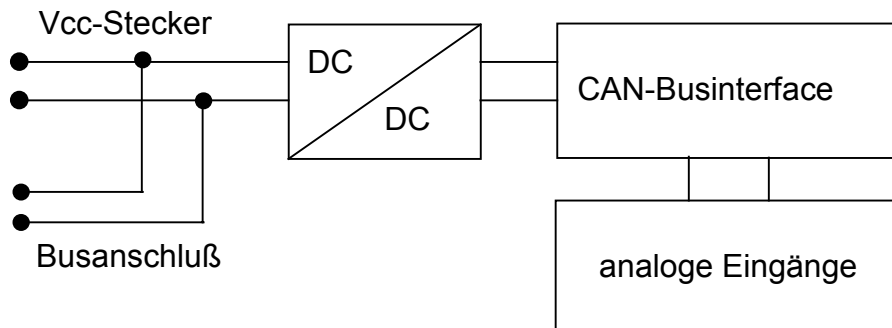
Die CAN-AI8 ist zum Einbau in Schaltschränke oder ähnliche EMV-dichte Gehäuse bestimmt. Die Verkabelung ist EMV-gerecht mit abgeschirmten Kabeln durchzuführen.

#### 3.2 Frontansicht



#### 3.3 Spannungsversorgung

Die CAN-AI8 ist galvanisch von der Versorgungsspannung getrennt. Die Versorgung kann über den Vcc-Stecker und/oder den Busanschluß erfolgen.



#### 3.4 CAN Abschlußwiderstand

Der CAN-Bus muß an jedem Ende mit 120  $\Omega$  abgeschlossen werden.

---

---

### 3.4.1 Analoge Eingänge

Die analogen Eingänge sind gemeinsam galvanisch von der Versorgungsspannung getrennt. Es stehen 0-10 Volt,  $\pm 10$  Volt, 0/4-20 mA, PT100 sowie weitere Eingangsspannungsbereiche zur Verfügung.

## 3.5 Steckverbinder

### 3.5.1 I/O-Anschlüsse

Die I/O-Anschlüsse werden über 4 polige ME-Steckverbinder geführt:

Spannungseingänge		
Stecker		Signal
1	1	+V_IN1
	2	-V_IN1
	3	+V_IN2
	4	-V_IN2
2	5	+V_IN5
	6	-V_IN5
	7	+V_IN6
	8	-V_IN6
3	9	+24V <sub>VCC</sub>
	10	-24V <sub>VCC</sub>
	11	PE
	12	PE

Spannungseingänge		
Stecker		Signal
4	13	+V_IN3
	14	-V_IN3
	15	+V_IN4
	16	-V_IN4
5	17	+V_IN7
	18	-V_IN7
	19	+V_IN8
	20	-V_IN8
6	21	-24V <sub>VCC</sub>
	22	GND <sub>CAN</sub>
	23	CAN_L
	24	CAN_H

Busanschluß (Unterseite)	
	Signal
25	-24V <sub>VCC</sub>
26	+24V <sub>VCC</sub>
27	GND <sub>CAN</sub>
28	CAN_L
29	CAN_H

---

Werden mehrere Module eingesetzt, kann die Spannungsversorgung und der CAN-Bus über den 5 poligen Tragschienen-Busverbinder geführt werden (Phoenix-Contact: Typ ME 22,5 TBUS 1,5/ 5-ST-3,81 KMGY).

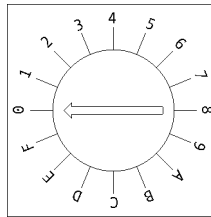


---

## 4 Generic CAN Firmware

### 4.1 Einstellen der Identifier und der Baudrate

Von der CAN-AI8 werden 4 aufeinander folgende Identifier (Standard: 11 Bit Länge) belegt. Die Identifier und die Baudrate werden über die beiden Hex-Drehschalter auf der Frontplatte eingestellt.



Der untere Hex-Drehschalter stellt den ID-Bereich und die Baudrate ein, mit dem oberen wird die ID gewählt.

Die Bereiche der ID wurden so gelegt, dass immer ein PDOx Bereich aus dem CANopen® Protokoll genutzt wird, so dass ein problemloser Parallelbetrieb möglich ist, wenn der entsprechende CANopen® Bereich nicht genutzt wird.

Schalter	Baudrate	CANopen® PDO	Adressbereich
0	50 KBaud	PDO1	384 - 639 (\$180 - \$27F)
1	50 KBaud	PDO2	640 - 895 (\$280 - \$37F)
2	50 KBaud	PDO3	896 - 1151 (\$380 - \$47F)
3	50 KBaud	PDO4	1152 - 1407 (\$480 - \$57F)
4	125 KBaud	PDO1	384 - 639 (\$180 - \$27F)
5	125 KBaud	PDO2	640 - 895 (\$280 - \$37F)
6	125 KBaud	PDO3	896 - 1151 (\$380 - \$47F)
7	125 KBaud	PDO4	1152 - 1407 (\$480 - \$57F)
8	500 KBaud	PDO1	384 - 639 (\$180 - \$27F)
9	500 KBaud	PDO2	640 - 895 (\$280 - \$37F)
A	500 KBaud	PDO3	896 - 1151 (\$380 - \$47F)
B	500 KBaud	PDO4	1152 - 1407 (\$480 - \$57F)
C	1 MBaud	PDO1	384 - 639 (\$180 - \$27F)
D	1 MBaud	PDO2	640 - 895 (\$280 - \$37F)
E	1 MBaud	PDO3	896 - 1151 (\$380 - \$47F)
F	1 MBaud	PDO4	1152 - 1407 (\$480 - \$57F)

---

Bitte beachten Sie, dass das Standard Frameformat verwendet wird, d.h. die Identifier sind 11 Bit lang.

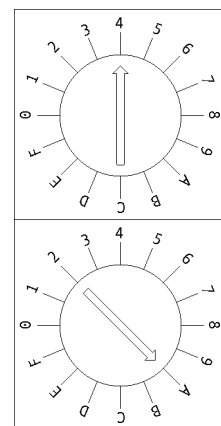
Mit dem oberen Hex-Drehesalter wird die Adresse der CAN-AI8 im jeweiligen Adressbereich definiert. Es ergeben sich folgende Offsets:

Schalter	Offset
0	168 (\$A8)
1	172 (\$AC)
2	176 (\$B0)
3	180 (\$B4)
4	184 (\$B8)
5	188 (\$BC)
6	192 (\$C0)
7	196 (\$C4)
8	200 (\$C8)
9	204 (\$CC)
A	208 (\$D0)
B	212 (\$D4)
C	216 (\$D8)
D	220 (\$DC)
E	224 (\$E0)
F	228 (\$E4)

Die Basisadresse ergibt sich aus der Addition von Adressbereich und Offset:

$$\text{BasisID} = \text{Adressbereich} + \text{Offset}$$





Stehen die Hex-Drehesalter z.B. auf 4A, so ergibt sich als Adressbereich PDO3 (896) und als Offset 184. Die Adresse des Moduls ist also 1080 (0x438). Die Baudrate beträgt 500 Kbaud.



---

## **4.2 Betriebszustand**

Die +5V LED zeigt an, dass die Betriebsspannung des Moduls vorhanden ist. Die Leuchtdioden R und E signalisieren den Betriebszustand des Moduls:

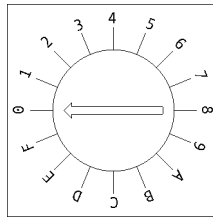
<b>Zustand</b>	<b>LED R</b>	<b>LED E</b>
Einschalten		
Datenaustausch		

---

## 5 CANopen® Firmware

### 5.1 Einstellen der Identifier und der Baudrate

Die CAN-AI8 arbeitet nach den CiA®-Standards DS301 V4.0.2 und DS401 V3.0. Die NodeID und die Baudrate werden über die beiden Hex-Drehschalter auf der Frontplatte eingestellt.



Der untere Hex-Drehschalter stellt den NodeID-Offset und die Baudrate ein, mit dem oberen wird nur die NodeID gewählt.

Schalter	Baudrate	NodeID_Offset
0	10 Kbaud	0
1	20 Kbaud	0
2	50 Kbaud	0
3	125 Kbaud	0
4	250 Kbaud	0
5	500 Kbaud	0
6	800 Kbaud	0
7	1 Mbaud	0
8	10 Kbaud	16
9	20 Kbaud	16
A	50 Kbaud	16
B	125 Kbaud	16
C	250 Kbaud	16
D	500 Kbaud	16
E	800 Kbaud	16
F	1 Mbaud	16

Mit dem oberen Hex-Drehschalter wird die NodeID der CAN-AI8 definiert. Es ergeben sich folgende Offsets:

Schalter	NodeID
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
A	10
B	11
C	12
D	13
E	14
F	15

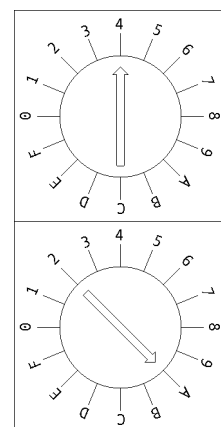
Sollte der Bereich der NodeID von 0-31 nicht ausreichen, kann über Lötjumper im CAN-AI8 Modul noch ein Offset von 32 bzw. 64 addiert werden. Damit können NodeID's von 0 bis 127 eingestellt werden. Wird Löt-Jumper ST2 geschlossen, addiert sich 32 auf die NodeID, mit Lötjumper ST1 werden noch 64 addiert.

Die tatsächliche NodeID ergibt sich aus der Addition von NodeID und NodeID\_Offset:

$$NodeID = NodeID + NodeID\_Offset$$

Stehen die Hex-Drehschalter z.B. auf 4A, so ergibt sich als NodeID\_Offset 16 und als NodeID 4. Die Adresse des Moduls ist also NodeID=20. Die Baudrate beträgt 50 Kbaud.







Würde jetzt noch ST1 geschlossen, betrüge die NodeID=84.



---

## **5.2 Betriebszustand**

Die +5V LED zeigt an, dass die Betriebsspannung des Moduls vorhanden ist. Die Leuchtdioden R (grün) und E (rot) signalisieren den Betriebszustand des Moduls:

<b>Zustand</b>	<b>LED R</b>	<b>LED E</b>
Init		
PreOperational (R blinkt)		
Operational		

## **5.3 CANopen® Objekte**

Es werden von dem Modul derzeit die folgenden CANopen® Objekte unterstützt.

### **5.3.1 CANopen® Funktionen**

Der Anwender hat die Wahl, ob er Heartbeat oder Nodeguarding nutzen möchte.

Bei Nodeguarding besteht die Möglichkeit zur lokalen Überwachung der Nodeguardtelegramme mit Hilfe des LifeTime-Faktors; falls die Überwachungszeit überschritten wird, wird ein Emergency-Telegramm vom Modul ausgelöst.

Die I/O Funktionalität ist für alle AI-Kanäle auf asynchrone Betriebsart mit einer InhibitTime von 50 ms und einer EventTime von 0 voreingestellt.

Vom Modul wird auch die synchrone Betriebsart (mit SYNC bzw. RTR Übertragung) unterstützt, diese Konfiguration muss dann im entsprechenden Konfigurationstool des CANopen®-Master eingestellt werden.

### 5.3.2 AI - Objekte

Objekt	Subindex	Datenrichtung	Wertebereich	Bedeutung
6401H	1..8	Read	0..7FE0	AD-Wert Kanal 1..8
6421H	1..8	Write	0,1,2,3,\$FF	AI Triggerselection Kanal 1..8
6424H	1	Write	0..7FE0	AI UpperLimit ( alle Kanäle)
6425H	1	Write	0..7FE0	AI LowerLimit ( alle Kanäle)
6426H	1	Write	0..7FE0	AI delta ( alle Kanäle )

Alle AD-Werte sind linksbündig im 10Bit Format abgelegt und wie folgt kodiert:

Eingangswert	CANopen®-Value
0 Volt	0000H
5 Volt	3FE0H
10 Volt	7FE0H
0,009766 Volt = 9,766 mV	1 digit = 0020H

Mit der Triggerselection kann für jeden Kanal die Limitüberwachung eingeschaltet werden.

Trigger-selection	Bedeutung
0	Keine Trigger ( default )
1	Trigger bei Überschreiten von UpperLimit bzw. Unterschreiten von LowerLimit
2	Trigger bei Wertänderung von mehr als AI delta
3	Wie 1+2, beide Triggerauslösungen
\$FF = 255	Jeden Wert melden

---

## **6 Hardwarebeschreibung**

### **6.1 ST4 – Serielle Schnittstelle**

Die serielle Schnittstelle dient nur der Programmierung der Controller. Sie hat folgende Belegung:

ST8	PIN	PIN	ST8
Vcc	1	2	MD2
Rx	3	4	Tx
GND	5	6	MD1

Nach dem Einschalten gibt die CAN-AI8 eine Versions-Meldung über die serielle Schnittstelle mit 38400 Baud aus. **ACHTUNG:** Die Schnittstelle stellt nur TTL-Pegel zur Verfügung. Der Anschluß einer normalen RS-232-Schnittstelle führt zur Zerstörung des Moduls.





---

## **7 Kommunikation mit der CAN-AI8**

### **7.1 Generic Can Kommunikation**

Die CAN-AI8 arbeitet als Slave. Der Master muß entweder eine Anfrage stellen, die entsprechend beantwortet wird, oder den zyklischen Versand der Daten anfordern. Von der CAN-AI8 werden 4 aufeinander folgende Identifier belegt. Die Basisadresse wird über die Hex-Drehschalter eingestellt (siehe cap. 4.1).

#### **7.1.1 Zyklisches Versenden der Messwerte**

Bei der Initialisierung kann der Master die CAN-AI8 so konfigurieren, dass sie die aktuellen Messwerte zyklisch verschickt. Die Zeit wird im Konfigurationstelegramm übertragen. Sie ist in ms anzugeben, das Raster beträgt 20 ms. Sollen die aktuellen Messwerte nicht selbsttätig verschickt werden, ist die Zeit auf 0 zu setzen.

#### **7.1.2 Mittelwertbildung**

Das CAN-AI8 Modul kann so parametrieren, dass es automatisch Mittelwerte der erfaßten Analogwerte bildet. Die Anzahl der Werte, die zur Mittelwertbildung herangezogen werden, kann parametrieren. Immer wenn die entsprechende Anzahl Werte gemessen wurde, wird ein neuer Mittelwert gebildet, der dann ggf. als aktueller Messwert übertragen wird.

Eine Messung dauert ca. 11  $\mu$ s pro Kanal. Das Modul misst laufend alle Kanäle und bildet ggf. die Mittelwerte. Die Messungen werden nur durch die Kommunikation über den CAN-Bus unterbrochen.

#### **7.1.3 Belegung der Identifier**

Der Offset der Identifier bezieht sich auf die BasisID (Einstellung siehe Seite 9, Kapitel 4.1). Bei den Telegrammen des Masters ist der Identifier grau hinterlegt.

---

### 7.1.3.1 Initialisierung

Kennung des Moduls abfragen:

Identifizier	R/W	Länge	Inhalt
0	W	1	01

M->S: Schick mir deine Kennung?

Antwort des Slave:

Identifizier	R/W	Länge	Inhalt
1	R	3	01 xx xx

xxxx = 0021 CAN-AI8 Modul mit 10 Bit Auflösung

xxxx = anderes Modul

Mittelwertbildung und zyklisches Versenden:

Identifizier	R/W	Länge	Inhalt
0	W	8	02 ii ii ii mm mm ss ss

iiiiii Basis-Identifizier des Moduls

mmmm Mittelwerte, 0 = keine Mittelwertbildung

ssss zyklisches Versenden der Messwerte in ms, 0=kein Versenden

### 7.1.3.2 Abfrage der analogen Werte

Analoge Werte des Moduls abfragen:

Identifizier	R/W	Länge	Inhalt
0	W	1	03

M->S: Schick mir deine Werte?

Antwort des Slave:

Identifizier	R/W	Länge	Inhalt
2	R	8	K1H K1L K2H K2L K3H K3L K4H K4L

jeder Kanal belegt ein Wort, hier Kanal 1-4

Identifizier	R/W	Länge	Inhalt
3	R	8	K5H K5L K6H K6L K7H K7L K8H K8L

jeder Kanal belegt ein Wort, hier Kanal 5-8

---

Die analogen Werte werden im Motorola-Format übertragen, d.h. zuerst das High-Byte, dann das Low-Byte.

Die beiden Telegramme werden unmittelbar aufeinander folgend verschickt. Wurde die Betriebsart "zyklisches Versenden" gewählt, so werden die beiden Telegramme ebenfalls im angegebenen Zeitabstand verschickt.

### 7.1.3.3 Kodierung der analogen Werte

Je nach Bestückung der Eingänge ergeben sich folgende Auflösungen:

Binärwert	Unipolar			Bipolar
	0-20 mA	4-20 mA	0-10 V	±10 V
0	0,0000 mA	3,91566 mA	0 V	-10,000 V
1FF	9,9805 mA	11,97229 mA	4,902 V	-0,019 V
200	10,0000 mA	11,98806 mA	5,000 V	0 V
3FF	19,9805 mA	20,04469 mA	9,902 V	9,980 V
LSB	19,53 $\mu$ A	15,7664 $\mu$ A	9,77 mV	19,53 mV

Bei CAN-AI8-Modulen mit Temperaturmesseingängen (Pt100, PT1000, KTY) wird direkt die Temperatur in °C mit Faktor 10 skaliert übertragen, d.h die Temperatur wird mit 1/10°C Auflösung ausgegeben.

Beispiel: Messwert = 0x007B = 123 heißt die Temperatur beträgt 12,3°C

Messwert = 0xFFE4 = -28 heißt die Temperatur beträgt -2,8°C

---

## 7.2 CANopen® Kommunikation

### 7.2.1 Implementierte CANopen®-Objekte

Wie aus der zugehörigen EDS-Datei zu ersehen ist, hat das CAN-AI8 Modul die Mandatory-Implementierung des DS-402 Profils:

- Device Type
- Error Register
- Emergency ID
- Producer Heartbeat Timer
- Identity
- Read Analog Input 16 Bit signed

Es werden folgende PDO-Transfer Modi unterstützt.

- SYNC – Transfer Anzahl der SYNCs
- Timed Transfer mit Event – Time

Zusätzlich wird die Einstellung der Inhibit-Time unterstützt, so dass die Häufigkeit der Can-Telegramm beschränkt werden kann.

### 7.2.2 Kodierung der analogen Werte

Da die AI8 einen 10 Bit Wandler hat, werden die sich ergebenden 10 Bit vorzeichenrichtig in den 16 Bit CANopen® Wert eingefügt:

- Unipolarer Mode : 0xxxxxxxxx00000
- Bipolarer Mode: VZxxxxxxxxx000000

Binärwert	Unipolar		
	0-20 mA	4-20 mA	0-10 V
0	0,0000 mA	4,0000 mA	0 V
3FE0	9,9805 mA	11,9844 mA	4,902 V
4000	10,0000 mA	12,0000 mA	5,000 V
7FE0	19,9805 mA	19,9844 mA	9,902 V

Binärwert	Bipolar
	±10 V
FFC0	-10,000 V
8000	-0,019 V
0000	0 V
7FC0	9,980 V