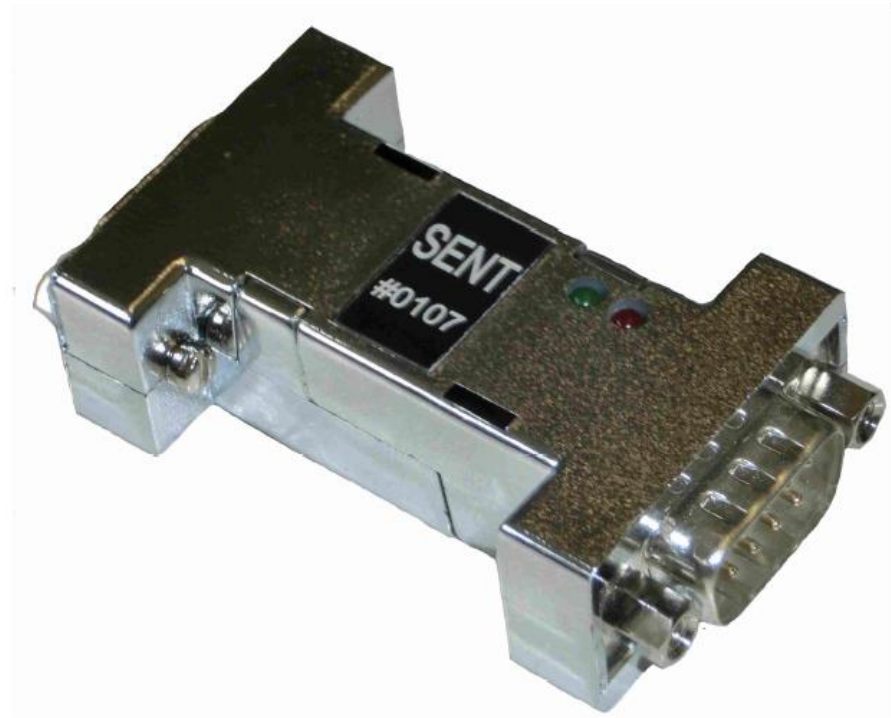


condalo

ELEKTRONIK & MECHATRONIK



SENT-To-CAN

Single-Edge-Nibble-Transmission
To
CAN

0.	Änderungsdokumentation	4
1.	Hardware	4
1.1.	Eigenschaften des SENT-To-CAN	4
1.2.	Pinbelegung	4
1.3.	Funktionen:.....	5
1.4.	Aufbau der CAN- Botschaften des SENT-To-CAN.....	8
1.4.1.	Modus1: 1 SENT-Nibble → 1 CAN-Byte.....	8
1.4.1.1.	CAN-Botschaften für die SENT-Funktion.....	8
1.4.1.1.1.	Datenbotschaft	8
1.4.1.1.2.	Enhanced-Botschaft	9
1.4.1.1.3.	Short-Botschaft	9
1.4.2.	Modus2: SENT-Nibble werden direkt in die CAN-Nachricht übernommen	10
1.4.2.1.	CAN-Botschaften für die SENT-Funktion.....	10
1.4.2.1.1.	Datenbotschaft	10
1.4.2.1.2.	Enhanced-Botschaft	11
1.4.2.1.3.	Short-Botschaft	11
1.4.3.	Modus3: SENT-Nibble ohne führende "0" in CAN-Nachricht	12
1.4.3.1.	CAN-Botschaften für die SENT-Funktion.....	12
1.4.3.1.1.	Datenbotschaft	12
1.4.3.1.2.	Enhanced-Botschaft	13
1.4.3.1.3.	Short-Botschaft	13
1.4.4.	Ausgabe für Digitaleingang	14
1.4.5.	Ausgabe für Analog/Digital-Wandler.....	14
1.4.6.	Ausgabe bei Fehler	15
1.5.	Leuchtanzeige.....	15
1.6.	Sleepmode	16
1.7.	Hinweise:.....	17
2.	Konfiguration:	18
2.1.	Software: CCOview	18
2.1.1.	Festlegung der CAN-Identifizier in einer INI-Datei	27
2.2.	Firmwareupdate:.....	28
3.	Technische Daten	29
4.	Lieferumfang SENT	30

5.	Impressum	31
5.1.	Firmwareupdates und Programmneuheiten.....	31
5.2.	Telefonische Unterstützung	31
5.3.	Fehlermeldungen und Verbesserungsvorschläge.....	31
5.4.	Anschrift.....	31

0. Änderungsdokumentation

Datum	Abschnitt	Art der Änderung	Bearbeiter	Ausgabe
29.03.11		Erste Fassung	Kulzer	0.0.1
15.04.11		Mastertrigger nicht vorgeschrieben	Kulzer	0.0.2
18.05.11		Enhanced-Botschaft hinzugefügt	Kulzer	0.0.3
19.11.12		2. Modus für die CAN-Ausgabe	Kulzer	0.0.5
08.02.13		Auto-Synchronisation für SENT-Botschaft	Kulzer	0.0.6
19.03.13		Überarbeitung für Rev.2 (Sendefunktion)	Kulzer	0.0.7
03.07.17		CAN-Ausgabemodi angepasst	Kulzer	0.0.8

1. Hardware

1.1. Eigenschaften des SENT-To-CAN

Mit diesem Gerät werden Daten der SENT (Single-Edge-Nibble-Transmission) Schnittstelle auf den CAN umgesetzt. Dadurch ist es möglich diese Daten mit einem CAN-Tool (z.B. condalo-Datenlogger) zu visualisieren, aufzuzeichnen und zu analysieren.

Mit diesem Gerät können auch Nachrichten über die SENT-Schnittstelle gesendet werden. Diese Daten können fest eingestellt oder auch Messwerte des AD-Wandlers sein. Auch das Übertragen von CAN-Daten zur SENT-Schnittstelle ist möglich.

Des Weiteren verfügt der SENT-To-CAN über zwei analoge und einen digitalen Eingang deren Spannungspegel bzw. Zustände auf dem CAN ausgegeben werden können.

1.2. Pinbelegung

Pinbelegung Sub-D Buchse:



Pin1: Analoger Eingang (Kanal 1)
Pin2: TX0- UART
Pin3: RX0- UART
Pin4: SENT Ein- bzw. Ausgang

Pin6: SENT Eingang mit Filter und Pullup(10k)
Pin7: Digitaler Eingang bzw. RTS (PC-UART)
Pin8: Analoger Eingang (Kanal 2)
Pin9: Versorgung 12 V

Pinbelegung Sub-D Stecker:

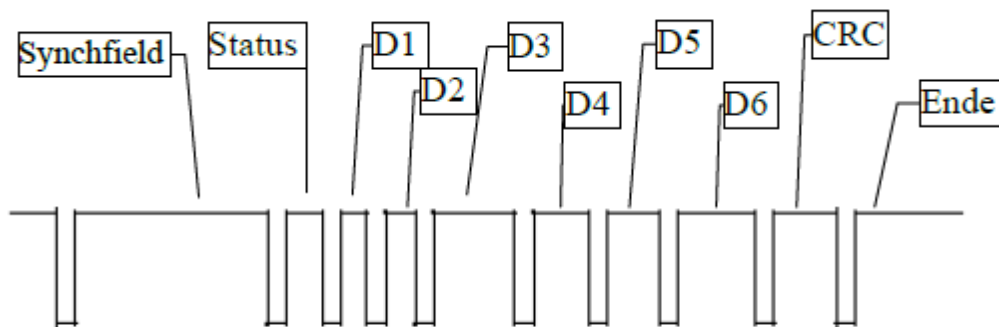


Pin1: ----	Pin6: GND
Pin2: CAN-L	Pin7: CAN-H
Pin3: GND	Pin8: ----
Pin4: SENT Ausgang mit Filterbeschaltung	Pin9: Versorgung 12 V
Pin5: GND	

1.3. Funktionen:

Der SENT-To-CAN setzt die ankommenden Signale auf dem SENT-Eingang in CAN-Botschaften um.

Eine SENT-Botschaft besteht aus einem Synchfield und 8 Daten-Nibbel (davon ein Status- und ein CRC-Nibbel). Optional kann ein Ende-Puls enthalten sein.



Ein Datennibbel besteht aus einer LOW-Phase von mindestens 5 Zeiteinheiten. Der rest des Nibbels hat einen HIGH-Pegel (ca. 7 Zeiteinheiten). Die Mindestlänge für ein DatenNibbel mit dem Wert 0 ist somit 12 Zeiteinheiten. Je nach Wert des Nibbels (0..15) kommen zu der HIGH-Phase noch je eine Zeiteinheit dazu.

D.h. bei einer Zeiteinheit von $3\mu\text{s}$ hat zum Beispiel:

- Der Wert 0 eine Gesamtlänge von $36\mu\text{s}$ zwischen zwei fallenden Flanken.
- Der Wert 15 hat eine Gesamtlänge von $81\mu\text{s}$ zwischen zwei fallenden Flanken.

Das Synchfield hat insgesamt eine Länge von 56 Zeiteinheiten wobei die LOW-Phase hier auch 5 Zeiteinheiten lang ist.

Die CAN-Botschaften sind wie folgt aufgebaut.

SENT-Daten:

Die Datenbotschaft enthält 8 Datenbytes. Hier enthält jedes Byte den Wert eines Nibbles der SENT-Botschaft.

Der Ausgabe-Identifizier für diese Botschaften können über das CCOview konfiguriert werden.

Enhanced-Botschaft:

Diese Botschaft setzt sich aus Bit2 und Bit3 des Statusnibbles aus 18 aufeinander folgenden SENT-Telegrammen zusammen.

Sie enthält 7 Datenbytes. Im ersten steht das Configuration Bit (0=>8Bit MessageID oder 1=>4Bit MessageID). Danach folgen 5 Daten-Nibbel (zuerst MessageID 8 oder 4 Bit und danach Daten 12 oder 16 Bit). Im siebten Datenbyte findet man die Checksumme (6 Bit).

Der Ausgabe-Identifizier für diese Botschaft kann über das CCOview konfiguriert werden.

Short-Botschaft:

Diese Botschaft setzt sich aus Bit2 des Statusnibbles aus 16 aufeinander folgenden SENT-Telegrammen zusammen.

Sie enthält 4 Datenbytes. Im ersten steht die MessageID (4 Bit lang). Danach folgen 2 Daten-Nibbel (je 4 Bit Daten). Im vierten Datenbyte findet man die Checksumme (4 Bit).

Der Ausgabe-Identifizier für diese Botschaft kann über das CCOview konfiguriert werden.

Der Analog/Digitalwandler misst in konfigurierbaren Zeitabständen die anliegende Spannung und gibt den Wert in einer CAN-Botschaft mit einstellbarem Identifier aus. Es ist möglich Botschaften nur auszugeben, wenn sich die Spannung um einen einstellbaren Wert seit der letzten Spannungsausgabe ändert. Die Spannung wird in mV angegeben und die Ausgabenorm (Big Endian bzw. Little Endian) ist konfigurierbar.

Der Signalzustand des digitalen Eingangs kann in konfigurierbaren Zeitabständen auf den CAN ausgegeben werden oder nur wenn eine Flankenänderung aufgetreten ist.

1.4. Aufbau der CAN- Botschaften des SENT-To-CAN

1.4.1. Modus1: 1 SENT-Nibble → 1 CAN-Byte

1.4.1.1. CAN-Botschaften für die SENT-Funktion

1.4.1.1.1. Datenbotschaft

CAN-ID: frei Wählbar (eingestellter Identifier)
Botschaft: Daten der SENT-Übertragung
Länge: 8 Byte

Parameter	Byte	Länge	Erläuterung
Status und Configuration Bit (CB)	0	1 Bit	0x00 (8Bit MessageID) 0x01 (4Bit MessageID)
Nibble 0	1	4 Bit	
Nibble 1	2	4 Bit	
Nibble 2	3	4 Bit	
Nibble 3	4	4 Bit	
Nibble 4	5	4 Bit	
Nibble 5	6	4 Bit	
CRC	7	4 Bit	Checksumme

1.4.1.1.2. Enhanced-Botschaft

CAN-ID: eingestellter Identifier
Botschaft: Enhanced – Botschaft
Länge: 7 Byte

Parameter	Byte	Länge	Erläuterung
Configuration bit (CB)	0	1 Bit	0x00 (8Bit MessageID) 0x01 (4Bit MessageID)
Nibble 0	1	4 Bit	CB = 0: MessageID (HIGH-Nibble) CB = 1: MessageID
Nibble 1	2	4 Bit	CB = 0: MessageID (LOW-Nibble) CB = 1: Daten (Bit 15 bis Bit 12)
Nibble 2	3	4 Bit	Daten (Bit 11 bis Bit 8)
Nibble 3	4	4 Bit	Daten (Bit 7 bis Bit 4)
Nibble 4	5	4 Bit	Daten (Bit 3 bis Bit 0)
CRC	6	6 Bit	Checksumme

1.4.1.1.3. Short-Botschaft

CAN-ID: eingestellter Identifier
Botschaft: Short – Botschaft
Länge: 4 Byte

Parameter	Byte	Länge	Erläuterung
MessageID	0	4 Bit	Message ID der Nachricht
Nibble 0	1	4 Bit	Daten (Bit 7 bis Bit 4)
Nibble 1	2	4 Bit	Daten (Bit 3 bis Bit 0)
CRC	3	4 Bit	Checksumme der Short-Botschaft

1.4.2. Modus2: SENT-Nibble werden direkt in die CAN-Nachricht übernommen

1.4.2.1. CAN-Botschaften für die SENT-Funktion

Zählweise in der CAN-Nachricht:

ID	DLC	D0	D1	D2	D3	D4	D5	D6	D7
11..0 oder 29..0	4..0	7..0	15..8	23..16	31..24	39..32	47..40	55..48	63..56

1.4.2.1.1. Datenbotschaft

CAN-ID: frei Wählbar (eingestellter Identifier)
Botschaft: Daten der SENT-Übertragung
Länge: 4 Byte

Parameter	Byte	Länge	Bit	Erläuterung
Nibble5 + CRC	0	8 Bit	7..4 3..0	Nibble 5 CRC
Nibble3 + Nibble4	1	8 Bit	15..12 11..8	Nibble 3 Nibble 4
Nibble1 + Nibble2	2	8 Bit	23..20 19..16	Nibble 1 Nibble 2
Status- und Config Bit(CB) + Nibble0	3	8 Bit	31..28 27..24	Status und Config-Bit (CB) Nibble 0

0	Bit7	NIBBLE 5	CRC	Bit0
1	Bit15	NIBBLE 3	NIBBLE 4	Bit8
2	Bit23	NIBBLE 1	NIBBLE 2	Bit16
3	Bit31	STATUS/CONFIG	NIBBLE 0	Bit24

1.4.2.1.2. Enhanced-Botschaft

CAN-ID: eingestellter Identifier
Botschaft: Enhanced – Botschaft
Länge: 4 Byte

Parameter	Byte	Länge	Bit	Erläuterung
CRC	0	6 Bit	7..0	Checksumme
Nibble 3 + Nibble 4	1	8 Bit	15..12 11..8	Daten (Bit 7 bis Bit 4) Daten (Bit 3 bis Bit 0)
Nibble 1 +	2	8 Bit	23..20	CB=0: MessageID (LOW-Nibble) CB=1: Daten (Bit 15 bis Bit 12)
Nibble 2			19..16	Daten (Bit 11 bis Bit 8)
Configuration Bit(CB) + Nibble 0	3	8 Bit	31..28 27..24	Status und Config-Bit (CB) CB=0: MessageID (HIGH-Nibble) CB=1: MessageID

1.4.2.1.3. Short-Botschaft

CAN-ID: eingestellter Identifier
Botschaft: Short – Botschaft
Länge: 2 Byte

Parameter	Byte	Länge	Bit	Erläuterung
Nibble 1 + CRC	0	8 Bit	7..4 3..0	Daten (Bit 3 bis Bit 0) Checksumme der Short-Botschaft
MessageID + Nibble 0	1	8 Bit	15..12 11..8	Message ID der Nachricht Daten (Bit 7 bis Bit 4)

1.4.3. Modus3: SENT-Nibble ohne führende "0" in CAN-Nachricht

1.4.3.1. CAN-Botschaften für die SENT-Funktion

Zählweise in der CAN-Nachricht:

ID	DLC	D0	D1	D2	D3	D4	D5	D6	D7
11..0 oder 29..0	4..0	7..0	15..8	23..16	31..24	39..32	47..40	55..48	63..56

1.4.3.1.1. Datenbotschaft

CAN-ID: frei Wählbar (eingestellter Identifier)
Botschaft: Daten der SENT-Übertragung
Länge: 4 Byte

Parameter	Byte	Länge	Bit	Erläuterung
Status- und Config Bit(CB) + Nibble0	0	8 Bit	3..0 7..4	Status und Config-Bit (CB) Nibble 0
Nibble1 + Nibble2	1	8 Bit	11..8 15..12	Nibble 1 Nibble 2
Nibble3 + Nibble4	2	8 Bit	19..16 23..20	Nibble 3 Nibble 4
Nibble5 + CRC	3	8 Bit	27..24 31..28	Nibble 5 CRC

1.4.3.1.2. Enhanced-Botschaft

CAN-ID: eingestellter Identifier
Botschaft: Enhanced – Botschaft
Länge: 4 Byte

Parameter	Byte	Länge	Bit	Erläuterung
Configuration Bit(CB) + Nibble 0	0	8 Bit	3..0 7..4	Status und Config-Bit (CB) CB=0: MessageID (HIGH-Nibble) CB=1: MessageID
Nibble 1 +	1	8 Bit	11..8	CB=0: MessageID (LOW-Nibble) CB=1: Daten (Bit 15 bis Bit 12)
Nibble 2			15..12	Daten (Bit 11 bis Bit 8)
Nibble 3 + Nibble 4	2	8 Bit	19..16 23..20	Daten (Bit 7 bis Bit 4) Daten (Bit 3 bis Bit 0)
CRC	3	6 Bit	31..24	Checksumme

1.4.3.1.3. Short-Botschaft

CAN-ID: eingestellter Identifier
Botschaft: Short – Botschaft
Länge: 2 Byte

Parameter	Byte	Länge	Bit	Erläuterung
MessageID + Nibble 0	0	8 Bit	3..0 7..4	Message ID der Nachricht Daten (Bit 7 bis Bit 4)
Nibble 1 + CRC	1	8 Bit	11..8 15..12	Daten (Bit 3 bis Bit 0) Checksumme der Short-Botschaft

1.4.4. Ausgabe für Digitaleingang

CAN-ID: frei Wählbar (eingestellter Identifier)
Botschaft: Zustand des Digitaleingang
Länge: 1 Byte

Parameter	Byte	Bit	Erläuterung
Pegel am Digitaleingang	0	0..7	0x01 (High-Pegel oder steigende Flanke) 0x00 (Low-Pegel oder fallende Flanke)

1.4.5. Ausgabe für Analog/Digital-Wandler

CAN-ID: frei Wählbar (eingestellter Identifier)
Botschaft: Spannung am Analogeingang
Länge: 2 Byte

Parameter	Byte	Bit	Erläuterung
Messwert	0	0..7	H-Byte bzw. L-Byte der Messung (je nach eingestellter Ausgabenorm)
“	1	8..15	L-Byte bzw. H-Byte der Messung (je nach eingestellter Ausgabenorm)

Das Messergebnis wird in mVolt ausgegeben.

1.4.6. Ausgabe bei Fehler

Bei einem Fehler (Pufferüberlauf oder CAN-Fehler) wird eine CAN-Nachricht generiert.

CAN-ID: frei Wählbar (eingestellter Identifier)
Botschaft: Fehlerausgabe
Länge: 2 Byte

Parameter	Byte	Bit	Erläuterung
Typ	0	0..7	0x01 → Pufferüberlauf 0x02 → CAN-Fehler
Daten	1	8..15	Bei Typ 0x01: Anzahl verlorener Botschaften (H-Byte) Bei Typ 0x02: Register CANGSTA des Controllers
Daten	2	16..23	Bei Typ 0x01: Anzahl verlorener Botschaften (L-Byte) Bei Typ 0x02: Register CANSTMOB des Controllers

1.5. Leuchtanzeige

Der SENT-To-CAN verfügt über zwei Leuchtdioden die den momentanen Betriebszustand anzeigen.

Die rote Leuchtdiode ist im Normalbetrieb ununterbrochen an.

Die grüne Leuchtdiode blinkt bei jeder CAN-Ausgabe, egal ob SENT-Nachrichten, Analogwerte oder Digitalpegel ausgegeben werden.

1.6. Sleepmode

Der SENT-To-CAN verfügt über einen Sleepmode.

Sleepmode:

Stellt der SENT-To-CAN für eine einstellbare Zeit kein Eingangssignal auf der SENT-Leitung fest, so wechselt er in den Standby-Mode. Durch einen Flankenwechsel auf der SENT-Leitung wacht der SENT-To-CAN auf und arbeitet mit der zuvor verwendeten Konfiguration. Die Stromaufnahme in diesem Modus beträgt ca. 450µA.

1.7. Hinweise:

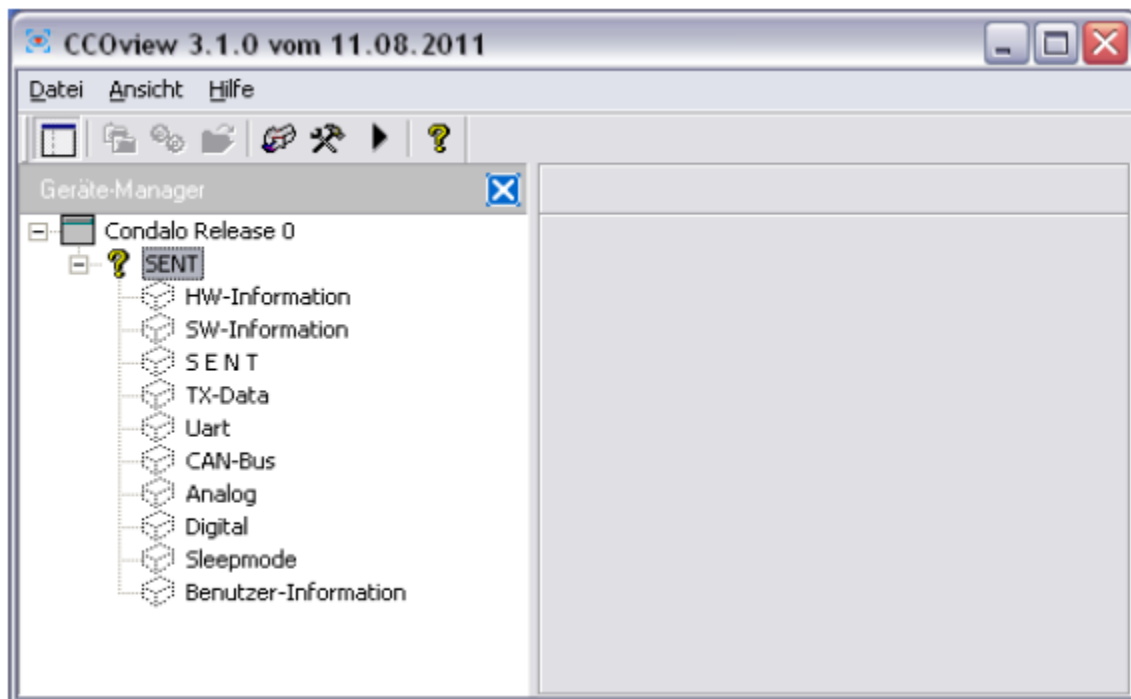
- Der SENT-To-CAN verfügt über keinen eigenen CAN- Busabschluss. Es ist deshalb darauf zu achten, dass bei einer Aufzeichnung mit dem Datenlogger mindestens ein Busabschluss von 120 Ohm zwischen CAN-High und CAN-Low geschaltet wird.
- Der CAN-Empfänger muss das Acknowledge für die CAN-Botschaft senden (sofern dies kein anderer Teilnehmer dieses Busses übernimmt), damit der SENT-To-CAN seine Botschaften absetzen kann.
- Bei einem Pufferüberlauf (nicht alle Botschaften können aufgrund zu hoher Buslast nicht rechtzeitig abgesetzt werden) wird eine CAN-Nachricht mit dem Identifier, der auch einen CAN-Error anzeigt, abgesetzt. Der Inhalt der Botschaft enthält Anzahl der verloren gegangenen Nachrichten.
- Wird für die Verbindung des PC mit dem SENT-To-CAN kein standardisiertes serielltes Kabel verwendet, so ist darauf zu achten, dass auf jeden Fall
 - die RX-Leitung des PC (PIN2),
 - die TX-Leitung des PC (PIN3),
 - der RTS (PIN7) und
 - GND (PIN5)angeschlossen sind, damit eine Konfiguration bzw. ein Firmwareupdate durchgeführt werden kann.

2. Konfiguration:

2.1. Software: CCOview

Mit Hilfe der Software „CCOview“ können die Einstellungen des SENT-To-CAN verändert und die Firmware upgedatet werden.

Ist der SENT-To-CAN mit dem seriellen Anschluss eines PCs verbunden und an eine Spannungsversorgung angeschlossen, so erscheint beim Start dieses Programms folgendes Fenster.



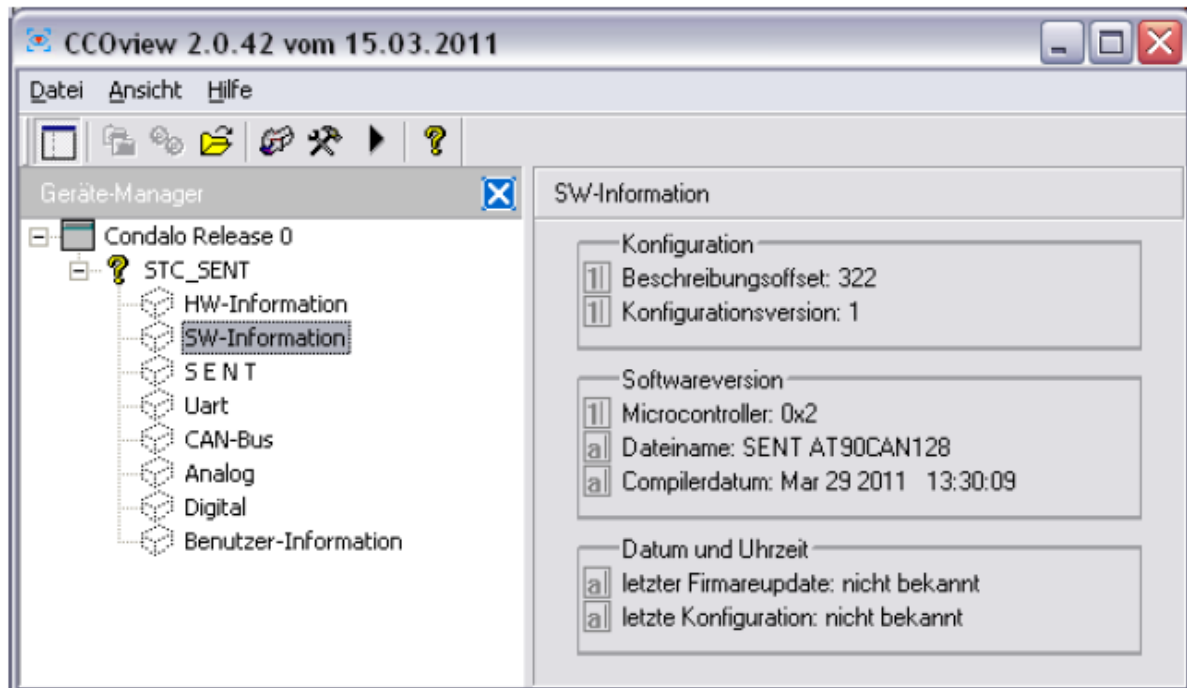
Alle momentan eingestellten Konfigurationen werden ausgelesen und angezeigt.

Die einzelnen Konfigurationsmöglichkeiten sind in den verschiedenen Zweigen des Konfigurationsbaumes untergebracht.

HW-Informationen:

Im Zweig HW-Informationen werden verschiedenen Infos Seriennummer und zur Hardware hinterlegt. Hier kann nichts verändert werden.

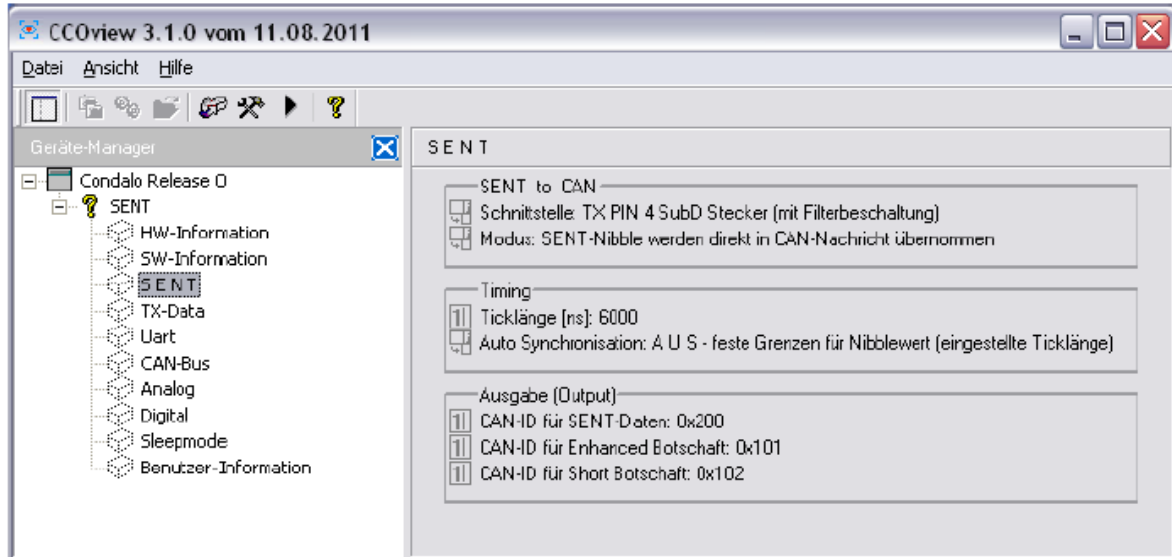
HW-Informationen:



Im Zweig SW-Informationen werden verschiedenen Infos zur Firmware hinterlegt. Hier kann nichts verändert werden.

Mögliche Einstellungen sind:

SENT:



Einstellungen:

Schnittstelle: - RX PIN4 SubD Buchse (keine Filterbeschaltung)
- TX PIN4 SubD Buchse (keine Filterbeschaltung)
- RX PIN6 SubD Buchse (mit Filterbeschaltung und 10k Pullup)
- TX PIN4 SubD Stecker (mit Filterbeschaltung)

Modus: 1 SENT-Nibble → 1 CAN-Byte (0x0?)
SENT-Nibble werden direkt in die CAN-Nachricht übernommen
SENT-Nibble ohne führende "0" in CAN-Nachricht

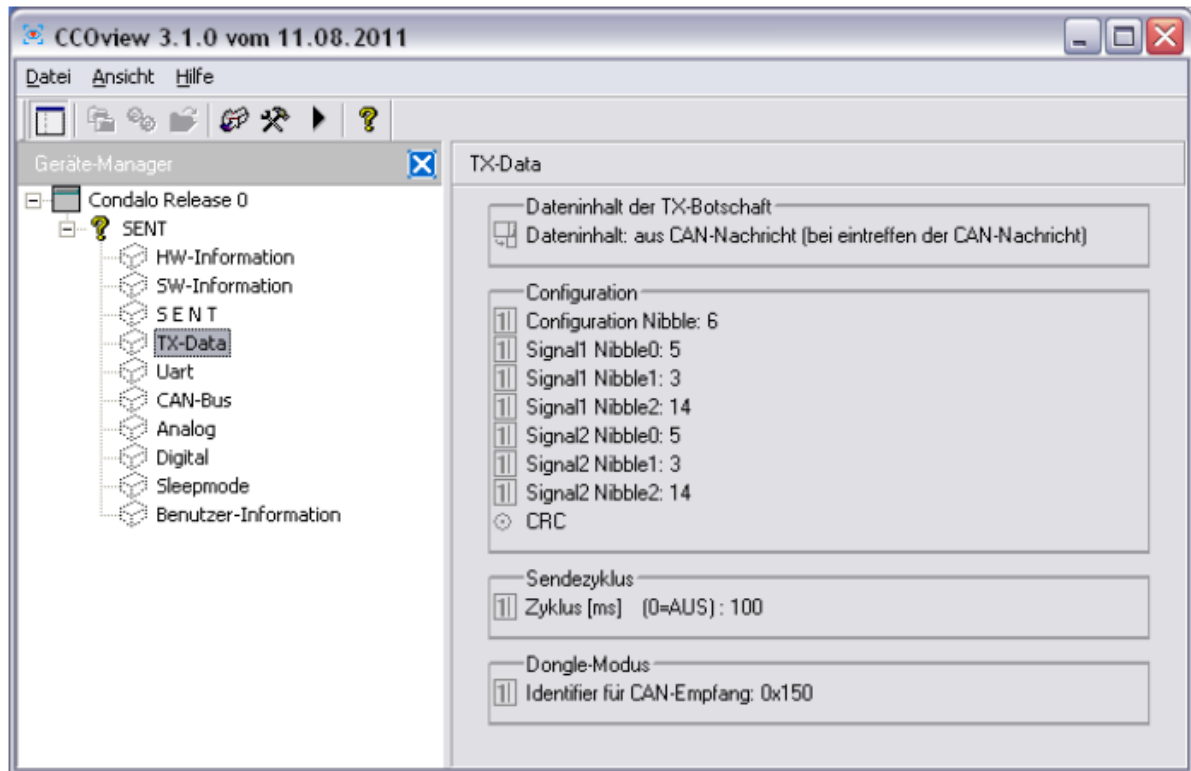
Ticklänge: Breite eines Ticks in ns
im RX-Modus wird die Ticklänge in ns für die Auswertung übernommen
im TX-Modus kann nur auf ganze µs eingestellt werden
(im µC ist die Tickbreite Ticklänge/1000)

Auto Synchronisation:
AUS-> Die Timinggrenzen für den Nibblewert werden aus der eingestellten Ticklänge berechnet.
EIN -> Die Timinggrenzen für den Nibblewert werden für jede Nachricht aus dem zugehörigen Synch berechnet.

Die Grenzen für die Entscheidung ob ein Synch angelegen hat, werden in beiden Fällen aus der vorgegebenen Ticklänge berechnet.

CAN-Identifizier für die SENT-Daten
CAN-Identifizier für die Enhanced-Botschaft
CAN-Identifizier für die Short-Botschaft

TX-Data:



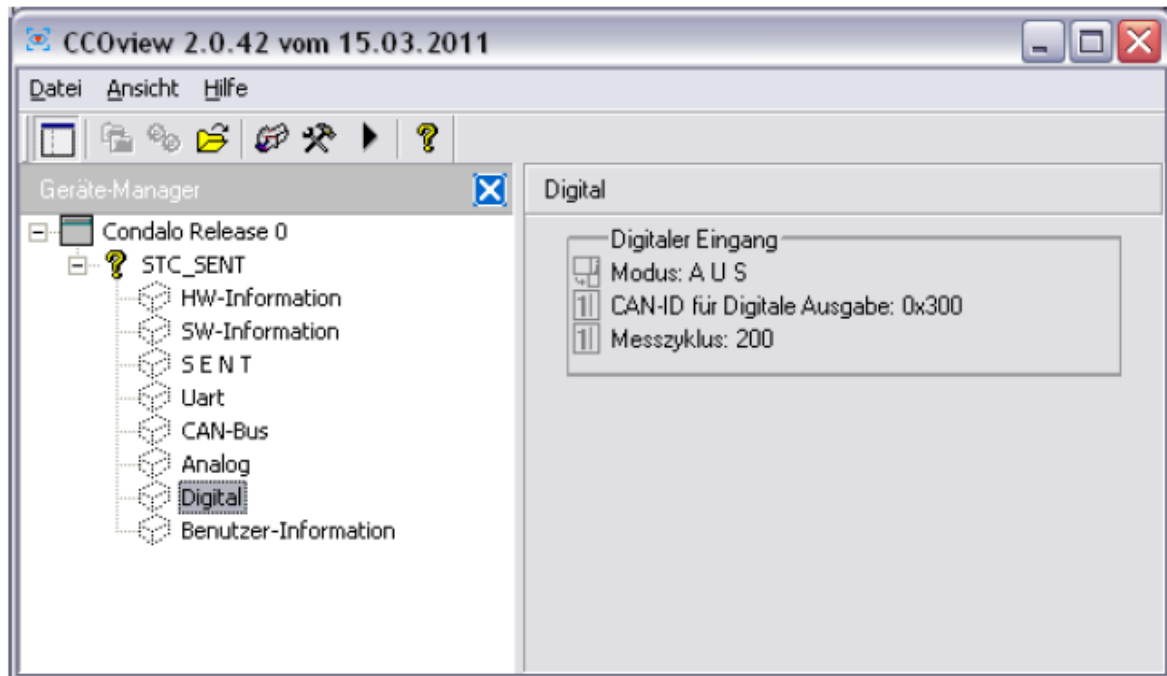
- Dateninhalt: - fester Dateninhalt (zyklisch)
- AD-Werte [Signal1=AD0 und Signal2=AD1] zyklisch
Das „Configuration Nibble“ kommt in diesem Fall von der untenstehenden Konfiguration
- aus CAN-Nachricht (bei eintreffender CAN-Nachricht)

Daten: Fester Dateninhalt kann in diesem Abschnitt eingestellt werden. Die CRC wird automatisch beim Senden berechnet.

Sendezyklus: Wiederholzyklus für die SENT-Nachricht

DongleModus: CAN-Nachrichten mit diesem Identifier werden in SENT-Nachrichten umgewandelt. Nachricht ist wie im Modus „1 SENT-Nibble → 1 CAN-Byte (0x0?)“ aufgebaut

Digitaler Eingang:



Einstellungen:

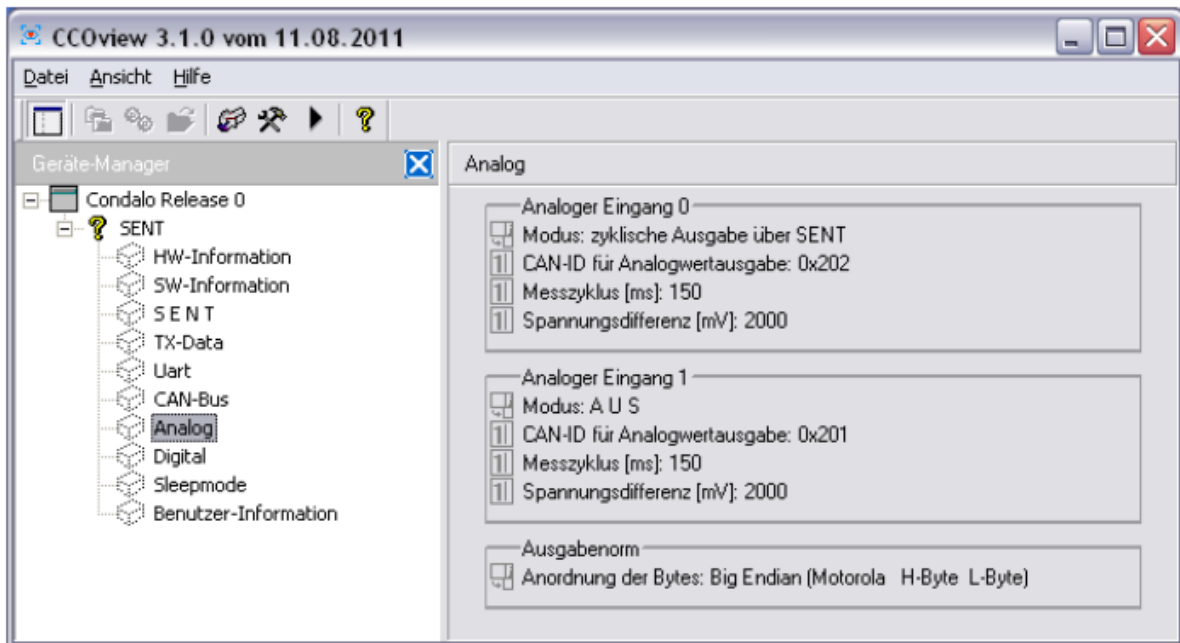
Modus:

AUS,
zyklische Ausgabe des Pegels,
steigende, fallende oder beide Flankenwechsel,

CAN-Identifizier für die Ausgabe des digitalen Pegels

Messzyklus bei zyklischer Ausgabe.

Analoger Eingang:



Einstellungen:

Modus:

- AUS
- zyklische Ausgabe des Messwertes (diese Messung ist für die CAN-Nachricht und die SENT-Nachricht verfügbar)
- Ausgabe des Messwertes nur bei einer Veränderung zur letzten Ausgabe (diese Messung ist für die CAN-Nachricht und die SENT-Nachricht verfügbar)
- zyklische Ausgabe über SENT (diese Messung ist für den CAN nicht verfügbar)

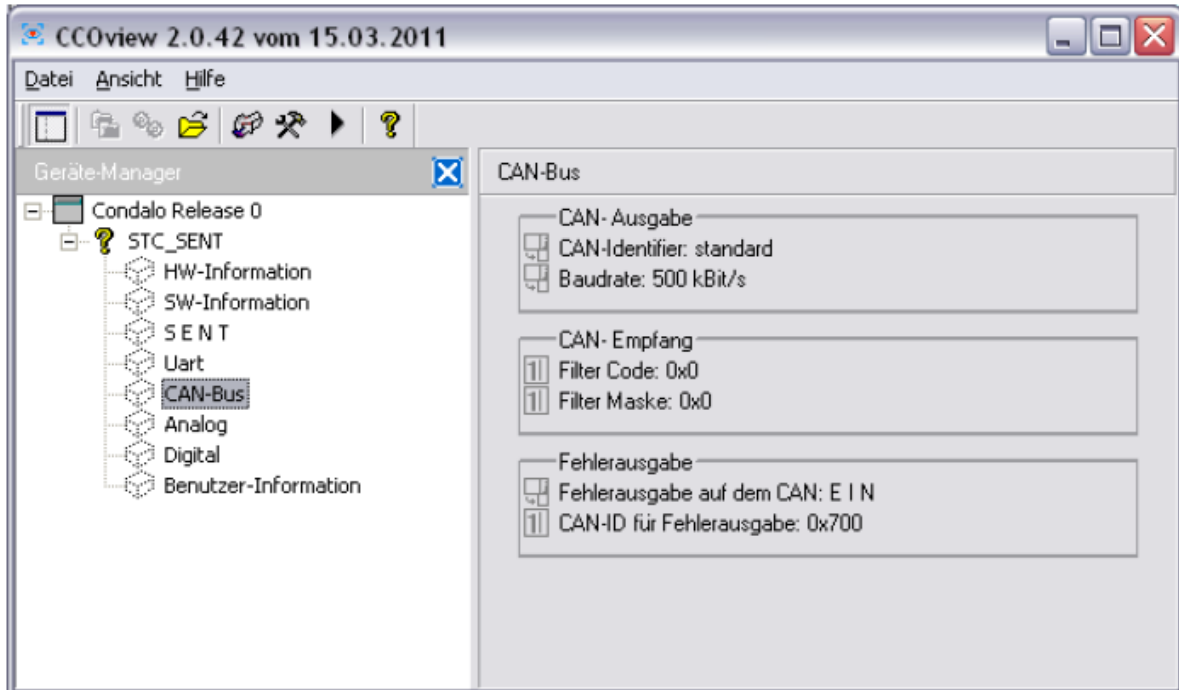
CAN-Identifizier für die Analogausgabe

Messzyklus für alle Modi

Spannungsdifferenz für den Modus, der nur bei Veränderung CAN-Daten sendet

Ausgabe-Norm für den Messwert auf dem CAN-Bus.

CAN Bus:



Einstellungen:

CAN-Ausgabe mit standard oder extended Identifier

CAN-Baudrate

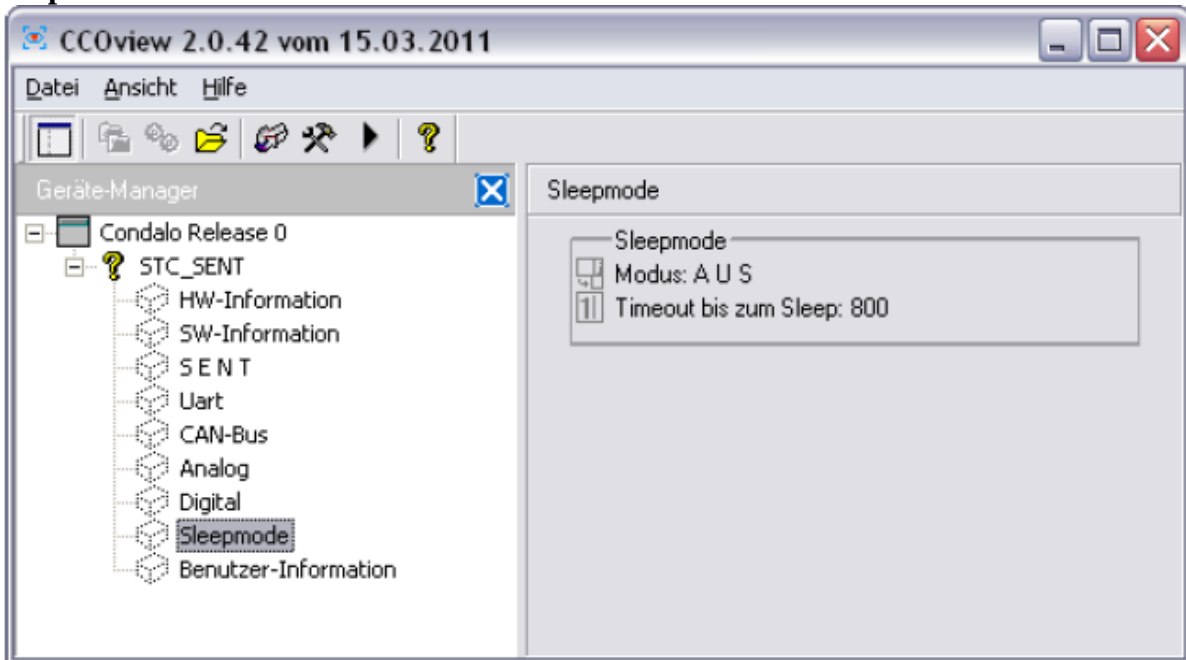
CAN-Filter-Code bei gefiltertem CAN-Empfang

CAN-Filter-Maske bei gefiltertem CAN-Empfang

Ausgabe dein Botschaft bei erkanntem Fehler

CAN-Identifizier für die Fehlerausgabe

Sleepmode:



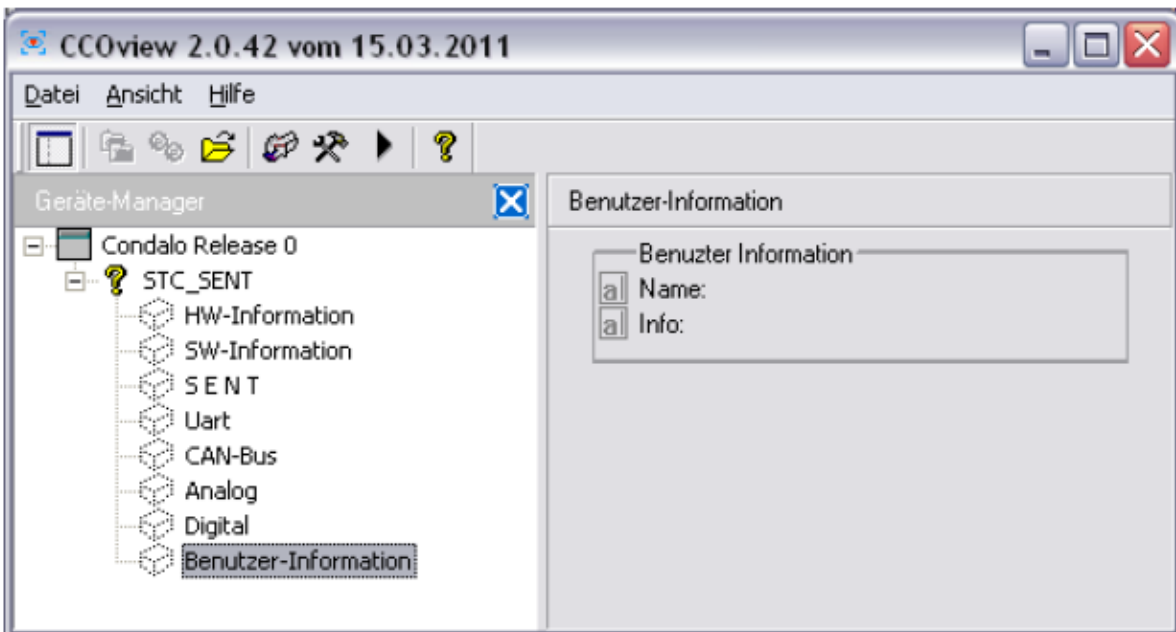
Einstellungen:

Modus:

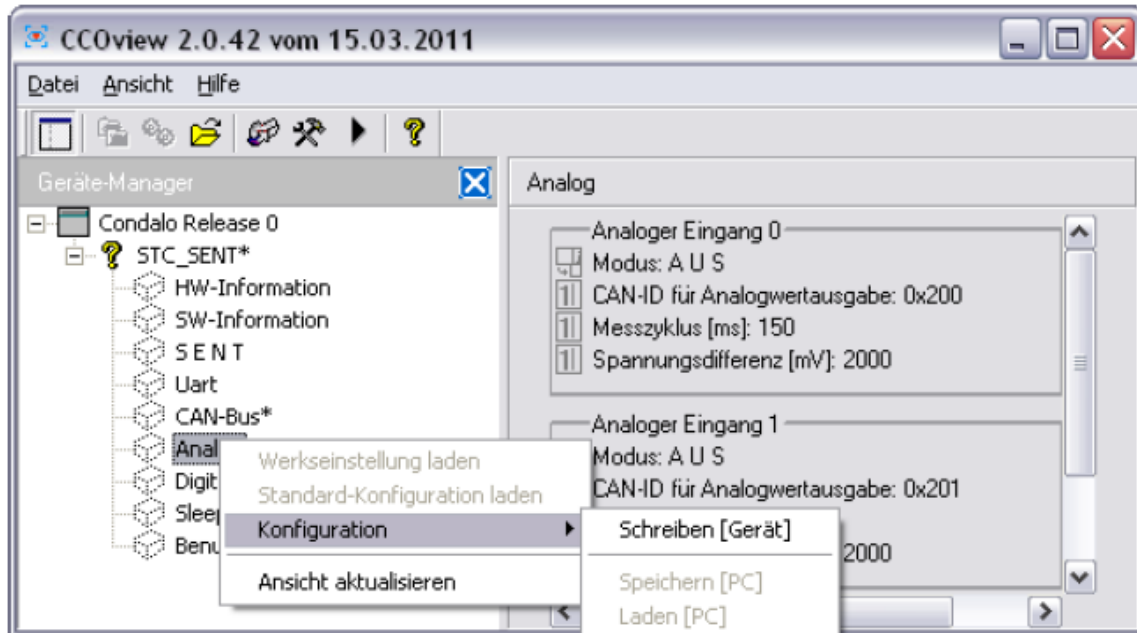
AUS,
EIN

Timeout (Busruhe) bis zum wechsel in den Sleepmode

Benutzer Infos:



Wurde eine Einstellung eines Zweiges verändert, so wird an das jeweilige Baumelement ein * - Stern angehängt. Durch den Befehl „Konfiguration → Schreiben[Gerät]“ im Kontextmenü des Baumes kann die Änderung zum STC übertragen werden.



Konfigurationen können auch auf dem PC Abgespeichert werden.
(Konfiguration → Schreiben [PC]).
Dadurch können sie auf andere Geräte übertragen werden.
(Konfiguration → Laden [PC]).

2.1.1. Festlegung der CAN-Identifizier in einer INI-Datei

Es ist möglich, symbolische Namen für CAN-Identifizier mit Hilfe einer INI-Datei zu vergeben. Befindet sich die INI-Datei im selben Ordner wie „CCOview.exe“, so können bei den CAN-Identifiern entweder die vorbelegten symbolischen Namen, sowie auch freie Eingaben verwendet werden.

Die INI-Datei ist wie folgt aufgebaut. Als Dateiname ist xTC.ini zu wählen.

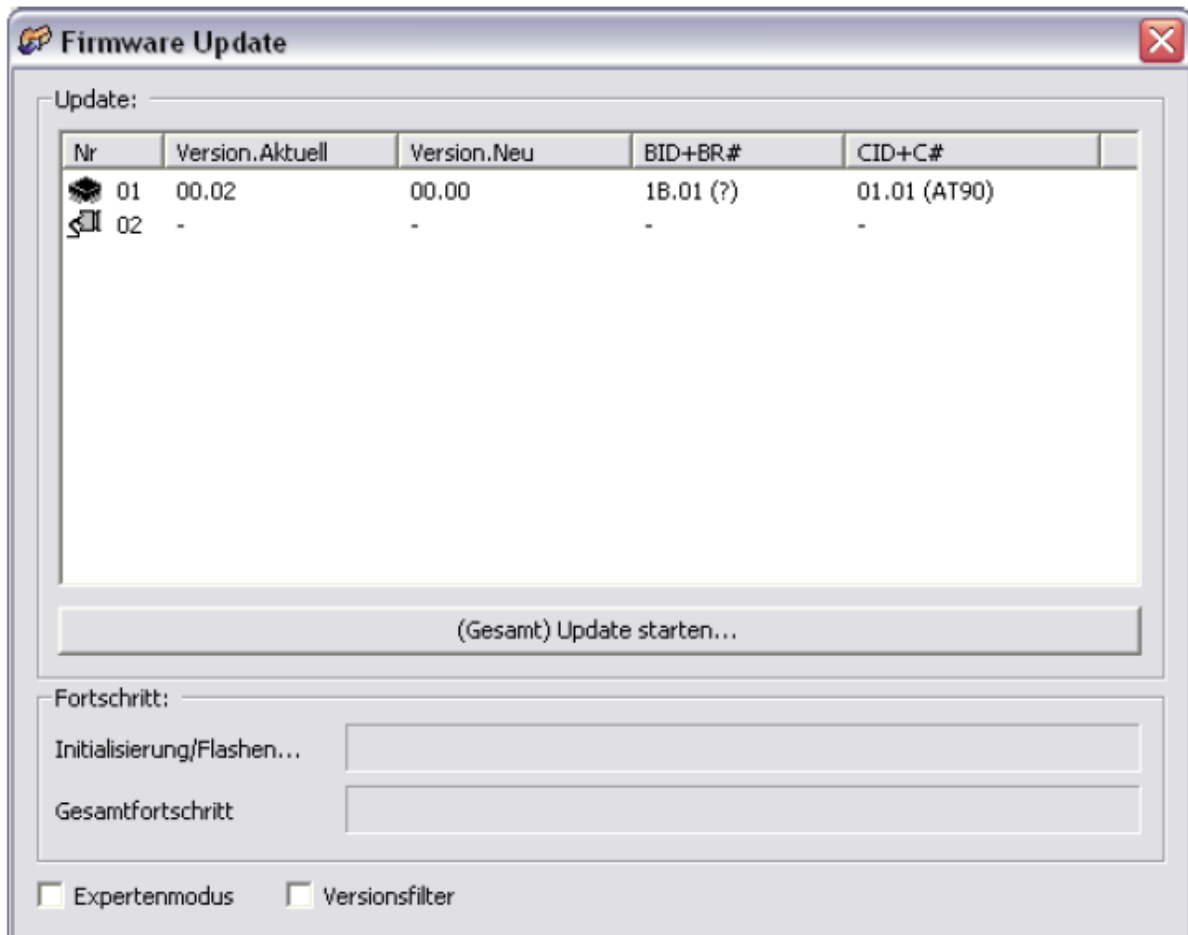
```
[Head]
Version=10
NumberOfEntries=3
[Item0]
Device=LTC
Name=Modus1 + Modus2 -- CAN-Identifizier für die LIN-Ausgabe
ID=[NameA1;0x203][NameA2;100][NameA3;200][NameA4;0x123]
[Item1]
Device=LTC
Name=Modus3 -- CAN-Identifizier Offset für die LIN-Ausgabe
ID=[NameB1;0x203][NameB2;100][NameB3;200][NameB4;0x123]

[Item2]
Device=BTC
Name=Identifizier für die BSD- Ausgabe
ID=[NameA1;0x203][NameA2;100][NameA3;200][NameA4;0x123]
```

Device =	Gerät, für welches die Einstellung gedacht ist (*.IMT)
NumberOfEntries =	Anzahl der in der INI-Datei enthaltenen ID-Datensätze
Item0; Item1; ...ItemN =	Nummerierung der einzelnen ID-Datensätze
Name =	Überschrift der CAN-ID Einstellung im CCOview
ID=	[symbolischer Name; zugehörige CAN-ID]

2.2. Firmwareupdate:

Die Firmware (*.imz) muss in den selben Order kopiert werden, in dem sich auch die CCOview.exe befindet



Falls eine neue Firmware erkannt wird, wird die Version unter „Version.Neu“ angezeigt. Durch einen klick auf „Update starten...“ wird die Firmware auf das Gerät geschrieben.

3. Technische Daten

Funktionen:

SENT nach CAN 2.0A, 2.0B Umsetzer (CAN-Highspeed-Transceiver)

Messung von zwei Spannungen 0 .. 25V

Überwachung eines digitalen Eingangs

Betriebsanzeige über zwei LEDs

Anschlüsse:

9pol Sub-D Buchse: Spannungsversorgung, RS232 (TX und RX),
2x Analog, 1x Digital, RS232 für Konfiguration

9pol Sub-D Stecker: Spannungsversorgung, CAN

Sonstiges:

Betriebsspannung: 12 V DC (7V DC - 30V DC)
Stromaufnahme max. 45mA

Temperaturbereich: -40°C bis +85°C

Abmessungen: 63mm x 17mm x 33mm

Gewicht: 22 Gramm

Material: Kunststoffgehäuse mit Metallüberzug

Analoge Eingänge: 0 - 25 V (9:1 vorgeteilt)
10 Bit Auflösung
± 2 LSB absolute Genauigkeit
Messzyklus 1ms bis 65535ms, im
Highspeed-Modus minimal 200µs

Digitaler Eingang: max. 50 V ohne zeitliche Beschränkung

Anschlusschema:

Konfiguration:

12 V (Pin 9)
GND (Pin 5)

PC
(SubD-Stecker des PC oder mindestens
Pin 2, 3, 5, 7)

SEN To CAN-Umsetzung:

CAN
Pin 2,7
(evtl. 120 Ohm
Abschluss)

12V (Pin 9)
GND (Pin 5)
SENT-RX (Pin 4)

4. Lieferumfang SENT

- 1 SENT (SENT To CAN Konverter)

5. Impressum

5.1. Firmwareupdates und Programmneuheiten

finden Sie in unserem Downloadbereich auf unserer Webseite

<http://www.condalo.de>

5.2. Telefonische Unterstützung

Zentrale 08450 - 9264 - 0

5.3. Fehlermeldungen und Verbesserungsvorschläge

info@condalo.de

5.4. Anschrift

condalo GmbH

Kohlstatt 3
86706 Lichtenau
Deutschland

Tel.: 08450 - 9264 - 0
Fax: 08450 - 9264 - 50
E-Mail: info@condalo.de
Web: www.condalo.de