



# STC

Serial To CAN

## Inhaltsverzeichnis

|  |    |
|--|----|
| 0. Änderungsdokumentation.....                                 | 3  |
| 1. Hardware.....   | 4  |
| 1.1. Eigenschaften des STC.....                                | 4  |
| 1.2. Pinbelegung.....  | 4  |
| 1.3. Ausführungen:.....  | 5  |
| 1.4. Funktionen:.....  | 6  |
| 1.5. Aufbau der CAN- Botschaften des STC.....                  | 7  |
| 1.5.1. CAN-Botschaften für die RS232-Funktion des STC.....     | 7  |
| 1.5.1.1. Datenbotschaft .....                                  | 7  |
| 1.5.1.2. Frameerror.....                                       | 7  |
| 1.5.2. Ausgabe für Digitaleingang.....                         | 7  |
| 1.5.3. Ausgabe für Analog/Digital-Wandler.....                 | 8  |
| 1.5.4. Ausgabe bei CAN-RESET.....                              | 8  |
| 1.5.5. CAN-Dongle-Funktion.....                                | 9  |
| 1.6. Serielle Ausgabe.....                                     | 12 |
| 1.7. Leuchtanzeige.....  | 13 |
| 1.8. Sleepmode.....  | 14 |
| 1.9. Hinweise:.....  | 15 |
| 2. Konfiguration:.....   | 16 |
| 2.1. Software: CCOview.....                                    | 16 |
| 2.1.1. Festlegung der CAN-Identifizier in einer INI-Datei..... | 24 |
| 2.1.2. Konfiguration abspeichern.....                          | 25 |
| 2.2. Firmwareupdate:.....                                      | 26 |
| 3. Technische Daten.....                                       | 27 |
| 4. Lieferumfang STC.....                                       | 28 |
| 5. Impressum.....  | 29 |
| 5.1. Firmwareupdates und Programmneuheiten.....                | 29 |
| 5.2. Telefonische Unterstützung.....                           | 29 |
| 5.3. Fehlermeldungen und Verbesserungsvorschläge.....          | 29 |
| 5.4. Anschrift.....  | 29 |

## 0. Änderungsdokumentation

| Datum    | Abschnitt | Art der Änderung  | Bearbeiter | Ausgabe |
|----------|-----------|---|------------|---------|
| 30.09.05 |           | STC 1.0, STC-Konfig 1.0.10  | Kulzer     | 1.0.0   |
| 08.11.05 |           | STC 2.0, STC-Konfig 1.0.10  | Kulzer     | 1.0.1   |
| 14.11.05 |           | Hinweise hinzugefügt  | Kulzer     | 1.0.2   |
| 17.03.06 |           | Änderungen für neues Konfigurationsprogramm   | Kulzer     | 1.0.3   |
| 07.08.06 |           | Änderung Firmenlogo   | Kulzer     | 1.0.4   |
| 22.05.07 |           | CAN-Buslastmessung und Dongle-Funtion   | Kulzer     | 1.0.5   |
| 17.07.07 |           | Überarbeitete Dongle-Funktion, Automatisches Senden über CAN                                      | Kulzer     | 1.0.6   |
| 20.11.07 |           | Donglefunktion und Automatisches Senden gleichzeitig möglich, UART für Donglefunktion umschaltbar | Kulzer     | 1.0.7   |
| 03.02.10 |           | Bidirektionaler STC-Modus   | Kulzer     | 1.0.8   |
| 19.07.10 |           | Baudrate im Donglemodus über RS232 einstellbar  | Kulzer     | 1.0.9   |
|          |           |   |            |         |
|          |           |   |            |         |
|          |           |   |            |         |
|          |           |   |            |         |
|          |           |   |            |         |

## 1. Hardware

### 1.1. Eigenschaften des STC

Mit diesem Gerät werden serielle Daten (z.B. RS232) auf den CAN umgesetzt. Dadurch ist es möglich RS232-Botschaften mit einem CAN-Tool (z.B. condalo-Datenlogger) zu visualisieren, aufzuzeichnen und zu analysieren.

Der STC verfügt über einen seriellen Anschluss, dessen Daten auf den CAN weitergegeben werden. Es ist auch möglich diese Daten parallel über einen zweiten seriellen Anschluss mit Hilfe des Konfigurationsprogrammes „CCOview“ auf einem PC zu visualisieren. Auch ein CAN-RS232-Dongle für geringe CAN-Buslast kann dadurch realisiert werden.

Des weiteren verfügt der STC über zwei analoge und einen digitalen Eingang deren Spannungspegel bzw. Zustände auf dem CAN bzw. seriell ausgegeben werden können.

### 1.2. Pinbelegung

#### Pinbelegung Sub-D Buchse:



|                                  |  |
|----------------------------------|--|
| Pin1: Analoger Eingang (Kanal 1) | Pin6: TX1- UART                            |
| Pin2: TX0- UART                  | Pin7: Digitaler Eingang bzw. RTS (PC-UART) |
| Pin3: RX0- UART                  | Pin8: Analoger Eingang (Kanal 2)           |
| Pin4: RX1- UART(Serial_IN)       | Pin9: Versorgung 12 V                      |
| Pin5: GND                        |  |

(RX0, TX0 für Konfiguration bzw. serielle Ausgabe und  
RX1, TX1 Eingang des umzuwandelnden Signals)

#### Pinbelegung Sub-D Stecker:



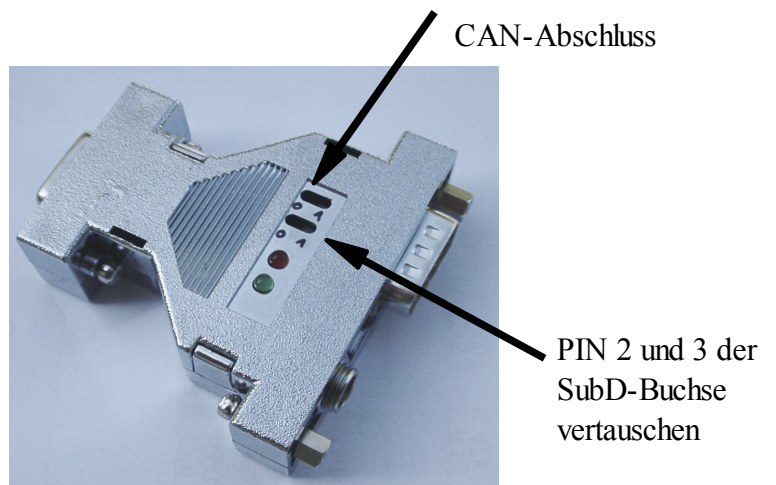
|             |                       |
|-------------|-----------------------|
| Pin1: ----  | Pin6: GND             |
| Pin2: CAN-L | Pin7: CAN-H           |
| Pin3: GND   | Pin8: ----            |
| Pin4: ----  | Pin9: Versorgung 12 V |
| Pin5: GND   |                       |

### 1.3. Ausführungen:

LTC: Pinbelegung wie beschrieben



STC2: Pinbelegung wie beschrieben



- zusätzlichem Spannungsanschluss über Kleinspannungsstecker:



- zuschaltbarer CAN-Busabschluss bzw. Pintauch an UART0:

Schalter in Stellung „1“:

- kein CAN-Busabschluss

- Standardbelegung für PIN 2 und 3 an der SUBD-Buchse (RX, TX von UART0)

Schalter in Stellung „0“:

- 120 Ohm CAN-Busabschluss mit Symmetrierung

- Pin 2 und 3 bei der SUBD-Buchse sind vertauscht (RX, TX von UART0)

## **1.4. Funktionen:**

Der STC setzt alle ankommenden seriellen Signale in CAN-Botschaften um. Dabei kann die Polarität der ankommenden Signale umgeschaltet werden. Die CAN-Botschaften können mit zwei verschiedenen Modi aufgebaut werden.

### MODUS 1:

Es werden immer 8 Bytes zusammengefasst und in einer CAN-Botschaft ausgegeben. Es ist möglich eine neue Botschaft nach einem konfigurierbaren Zeichen zu beginnen auch wenn noch keine 8 Datenbytes zusammengefasst sind. Die letzte Botschaft wird, sofern keine 8 Bytes gesammelt werden konnten, nach einem einstellbaren Timeout ausgegeben.

### MODUS 2:

Hier wird versucht jedes ankommende Byte in einer eigenen CAN-Botschaft zu senden. Kann eine CAN-Botschaft aufgrund zu hoher Busauslastung nicht sofort abgesetzt werden, so werden die ankommenden Bytes solange gesammelt, bis der CAN-Bus wieder frei ist. Die gesammelten Bytes werden danach zusammengefasst in einer CAN-Botschaft ausgegeben.

Der STC kann auch als CAN-RS232-Dongle verwendet werden. Es ist also möglich über die RS232-Schnittstelle CAN-Nachrichten zu erstellen, sowie CAN-Nachrichten mit einem Terminalprogramm am PC darzustellen. Dabei ist aber wegen der geringeren Geschwindigkeit der seriellen Schnittstelle, keine 100% Buslast auf dem CAN möglich.

Es kann auch die aktuelle CAN-Buslast über die serielle Schnittstelle ausgegeben werden. Dabei wird die Buslast in Prozent sowie die Anzahl der Standard-, Extended- sowie Errorframes pro Sekunde angezeigt.

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 zwischen zwei Messungen 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.

## 1.5. Aufbau der CAN- Botschaften des STC

### 1.5.1. CAN-Botschaften für die RS232-Funktion des STC

#### 1.5.1.1. Datenbotschaft

CAN-ID: frei Wählbar (eingestellter Identifier)  
Botschaft: Daten der RS232-Schnittstelle  
Länge: 0..8 Byte

| Parameter   | Byte | Bit   | Erläuterung                       |
|-------------|------|-------|-----------------------------------|
| Dateninhalt | 0..8 | 0..63 | Datenbyte der RS232-Schnittstelle |

#### 1.5.1.2. Frameerror

CAN-ID: eingestellter Identifiziert + 4  
Botschaft: Daten der RS232-Schnittstelle  
Länge: 0..8 Byte

| Parameter   | Byte | Bit   | Erläuterung                       |
|-------------|------|-------|-----------------------------------|
| Dateninhalt | 0..8 | 0..63 | Datenbyte der RS232-Schnittstelle |

### 1.5.2. 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)<br>0x00 (Low-Pegel oder fallende Flanke) |

### 1.5.3. 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<br>(je nach eingestellter Ausgabenorm) |
| “         | 1    | 8..15 | L-Byte bzw. H-Byte der Messung<br>(je nach eingestellter Ausgabenorm) |

Das Messergebnis wird in mVolt ausgegeben.

### 1.5.4. Ausgabe bei CAN-RESET

Bei einem Fehler auf dem CAN (Acknowledge, Bitstuffing) wird ein Reset des CAN-Controllers durchgeführt. Damit festgestellt werden kann, wann und wie oft solch ein Fehler vorkommt, wird bei jedem CAN-Reset eine CAN-Botschaft abgesetzt.

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

| Parameter | Byte | Bit   | Erläuterung |
|-----------|------|-------|-------------|
| Byte 1    | 0    | 0..7  | 0x00        |
| Byte 2    | 1    | 8..15 | 0x01        |



### 1.5.5. CAN-Dongle-Funktion

#### CAN-Baudrate über RS232 ändern (HEX):

Byte1: „b“ (0x62) für Baudrate ändern.  
Byte2-5: neue Baudrate HB...LB

Beispiel: 500 kBit/s = 500 000 = 0x07 A1 20  
=> 0x62 0x00 0x07 0xA1 0x20

1 MBit/s = 1 000 000 = 0x0F 42 40  
=> 0x62 0x00 0x0F 0x42 0x40

nach dem letzten Datenbyte wird automatisch die CAN-Baudrate eingestellt.

Die hier eingestellte Baudrate wird nicht abgespeichert. Nach einem Reset des Gerätes muss dieses Kommando erneut gesendet werden

#### CAN-Botschaft über RS232 senden (HEX-Modus):

Byte1: „s“ (0x73) bzw. „e“ (0x65) für standard oder extended CAN-Frame  
Byte2-5: CAN-ID (HighByte .... LowByte) immer 4 Byte lang!!!  
Byte 6: Länge (Anzahl der Datenbytes)  
Byte 7-x: Datenbytes

Beispiel: ID:0x150 Len:3 Daten: 0x01 0x12 0x23  
=> 0x73 0x00 0x00 0x01 0x50 0x03 0x01 0x12 0x23

nach dem letzten Datenbyte wird automatisch die CAN-Nachricht generiert.

#### CAN-Botschaft über RS232 senden (ASCII-Modus):

für jedes Byte der CAN-Nachricht, werden 2 Zeichen benötigt!!!

Zeichen 1: „s“ bzw. „e“ für standard oder extended CAN-Frame  
Zeichen 2-9: CAN-ID (HighByte .... LowByte) immer 8 Zeichen lang!!!  
Zeichen 10-11: Länge  
Byte 7-x: Datenbytes

Beispiel: ID:0x150 Len:3 Daten: 0x01 0x12 0x23  
=> „s0000015003011223“

nach dem letzten Zeichen wird automatisch die CAN-Nachricht generiert.

### Empfangene CAN-Botschaften über RS232 ausgeben (HEX-CCOview):

Byte 1: Kanal (für OnlineViewer im CCOview)  
Byte 2: Anzahl der nachfolgenden Bytes  
Byte 3-6: CAN-ID (HighByte .... LowByte)  
Byte 7: Länge: (Anzahl der Datenbytes)  
Byte 8-x: Datenbytes

Das oberste Bit der ID zeigt an, ob ein STD- bzw. ein EXT-Identifizier empfangen wurde.

### Empfangene CAN-Botschaften über RS232 ausgeben (ASCII-Terminal):

Bsp: STD: 150 3 01 12 23  
EXT: 00012345 3 01 12 23

### Empfangene CAN-Botschaften über RS232 ausgeben (HEX-Terminal):

Bsp: 0x0A 0x65 0x00 0x00 0x12 0x23 0x04 0x01 0x02 0x03 0x04

Byte 1: Anzahl der nach diesem Byte folgenden Bytes (für dieses CAN-Telegramm)  
Byte 2: „s“ bzw. „e“ (0x73 bzw. 0x65) für standard bzw. extended Frame  
Byte 3-6: Identifizier der CAN-Nachricht (immer 4 Byte lang)  
Byte 7: Länge der CAN-Nachricht  
Byte 8-x: Datenbytes

Aufgrund der niedrigeren Datenrate der RS232-Schnittstelle kann es zu Telegrammverlusten kommen. In diesem Fall wird eine Botschaft über die Serielle Schnittstelle ausgegeben.

### Ausgabe bei Botschaftsverlust (nur über RS232):

CAN-ID: 0x1FFFFFFF  
Botschaft: Telegrammverlust CAN zu RS232  
Länge: 4 Byte

| Parameter       | Byte | Bit   | Erläuterung                                |
|-----------------|------|-------|--|
| Byte 0          | 0    | 0..7  | 0xAA                                       |
| Byte 1          | 1    | 8..15 | 0xBB                                       |
| Zähler (H-Byte) | 2    | 16-23 | Anzahl der verlorenen Botschaften (H-Byte) |
| Zähler (L-Byte) | 3    | 24-31 | Anzahl der verlorenen Botschaften (L-Byte) |

### Ausgabe bei CAN-Fehler (nur über RS232):

CAN-ID: 0x1FFFFFFF  
Botschaft: CAN-Fehler detektiert  
Länge: 4 Byte

| Parameter       | Byte | Bit   | Erläuterung           |
|-----------------|------|-------|-----------------------|
| Byte 0          | 0    | 0..7  | 0x11                  |
| Byte 1          | 1    | 8..15 | 0x22                  |
| Zähler (H-Byte) | 2    | 0     | 1=Error Passiv Mode   |
|                 |      | 1     | 1=Bus Off Mode        |
|                 |      | 2..7  | keine Bedeutung       |
|                 |      |       |                       |
| Zähler (L-Byte) | 3    | 0     | 1 = Acknowledge Error |
|                 |      | 1     | 1 = Form Error        |
|                 |      | 2     | 1 = CRC Error         |
|                 |      | 3     | 1 = Stuff Error       |
|                 |      | 4     | keine Bedeutung       |
|                 |      | 5     | keine Bedeutung       |
|                 |      | 6     | Bus Off               |
|                 |      | 7     | keine Bedeutung       |

Bei einer Aufzeichnung mit dem condalo-Datenlogger II setzt sich eine Botschaft wie folgt zusammen:

Eine Botschaft im CCO-File hat immer 16 Byte. Jede Botschaft hat  
5 Byte Zeitstempel,  
1 Byte Datenlogger- Kanal,  
2 Byte CAN-ID mit Datenlänge,  
maximal 8 Byte Dateninhalt.

Ist die Datenlänge kleiner als 8 Byte so sind nur die Bytes bis zur Datenlänge gültig.

## 1.6. Serielle Ausgabe

Es ist möglich die gleichen Daten wie auf dem CAN-Bus auch über die serielle Schnittstelle auf dem PC darzustellen.

Im Onlineviewer der mitgelieferten PC-Software „CCOview“ werden die Daten in korrekter zeitlicher Abfolge mit einem Zeitstempel versehen angezeigt.

Neben dem Zeitstempel enthält die dargestellte Botschaft den Kanal (woher sie kommt), die Länge und die darzustellenden Informationen.

| #     | Zeitstempel | Kanal | Länge | Daten   |
|-------|-------------|-------|-------|---|
| 10... | 2194.192850 | Hex   | 16    | 70 71 72 73 74 75 76 77 78 79 7A 30 31 32 33 34 : p q r s t u v w x y z 0 1 2 3 4 |
| 10... | 2194.192854 | Hex   | 16    | 35 36 37 38 39 41 42 43 44 45 46 47 48 49 4A 4B : 5 6 7 8 9 A B C D E F G H I J K |
| 10... | 2194.192858 | Hex   | 16    | 4C 4D 4E 4F 50 51 52 53 54 55 56 57 58 59 5A 61 : L M N O P Q R S T U V W X Y Z a |
| 10... | 2194.192862 | Hex   | 16    | 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 : b c d e f g h i j k l m n o p q |
| 10... | 2194.192866 | Hex   | 16    | 72 73 74 75 76 77 78 79 7A 30 31 32 33 34 35 36 : r s t u v w x y z 0 1 2 3 4 5 6 |
| 10... | 2194.192870 | Hex   | 16    | 37 38 39 41 42 43 44 45 46 47 48 49 4A 4B 4C 4D : 7 8 9 A B C D E F G H I J K L M |
| 10... | 2194.192874 | Hex   | 16    | 4E 4F 50 51 52 53 54 55 56 57 58 59 5A 61 62 63 : N O P Q R S T U V W X Y Z a b c |
| 10... | 2194.192878 | Hex   | 16    | 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 : d e f g h i j k l m n o p q r s |
| 10... | 2194.192881 | Hex   | 16    | 74 75 76 77 78 79 7A 30 31 32 33 34 35 36 37 38 : t u v w x y z 0 1 2 3 4 5 6 7 8 |
| 10... | 2194.192885 | Hex   | 16    | 39 41 42 43 44 45 46 47 48 49 4A 4B 4C 4D 4E 4F : 9 A B C D E F G H I J K L M N O |
| 10... | 2194.192889 | Hex   | 16    | 50 51 52 53 54 55 56 57 58 59 5A 61 62 63 64 65 : P Q R S T U V W X Y Z a b c d e |
| 10... | 2194.192893 | Hex   | 16    | 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 : f g h i j k l m n o p q r s t u |
| 10... | 2194.192897 | Hex   | 16    | 76 77 78 79 7A 30 31 32 33 34 35 36 37 38 39 41 : v w x y z 0 1 2 3 4 5 6 7 8 9 A |
| 10... | 2194.192901 | Hex   | 16    | 42 43 44 45 46 47 48 49 4A 4B 4C 4D 4E 4F 50 51 : B C D E F G H I J K L M N O P Q |
| 10... | 2194.192905 | Hex   | 16    | 52 53 54 55 56 57 58 59 5A 61 62 63 64 65 66 67 : R S T U V W X Y Z a b c d e f g |
| 10... | 2194.192909 | Hex   | 16    | 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 : h S T U V W X Y Z a b c d e f g |
| 10... | 2194.192913 | Hex   | 16    | 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 : h i j k l m n o p q r s t u v w |
| 10... | 2194.192917 | Hex   | 16    | 78 79 7A 30 31 32 33 34 35 36 37 38 39 41 42 43 : x y z 0 1 2 3 4 5 6 7 8 9 A B C |
| 10... | 2194.192921 | Hex   | 16    | 44 45 46 47 48 49 4A 4B 4C 4D 4E 4F 50 51 52 53 : D E F G H I J K L M N O P Q R S |
| 10... | 2194.192925 | Hex   | 16    | 54 55 56 57 58 59 5A 61 62 63 64 65 66 67 68 69 : T U V W X Y Z a b c d e f g h i |
| 10... | 2194.192929 | Hex   | 16    | 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 78 79 : j k l m n o p q r s t u v w x y |
| 10... | 2194.192933 | Hex   | 16    | 7A 30 31 32 33 34 35 36 37 38 39 41 42 43 44 45 : z 0 1 2 3 4 5 6 7 8 9 A B C D E |
| 10... | 2194.192937 | Hex   | 16    | 46 47 48 49 4A 4B 4C 4D 4E 4F 50 51 52 53 54 55 : F G H I J K L M N O P Q R S T U |
| 10... | 2194.192941 | Hex   | 16    | 56 57 58 59 5A 61 62 63 64 65 66 67 68 69 6A 6B : V W X Y Z a b c d e f g h i j k |
| 10... | 2194.192945 | Hex   | 16    | 6C 6D 6E 6F 70 71 72 73 74 75 76 77 78 79 7A 30 : l m n o p q r s t u v w x y z 0 |
| 10... | 2194.192949 | Hex   | 16    | 31 32 33 34 35 36 37 38 39 41 42 43 44 45 46 47 : 1 2 3 4 5 6 7 8 9 A B C D E F G |
| 10... | 2194.192953 | Hex   | 16    | 48 49 4A 4B 4C 4D 4E 4F 50 51 52 53 54 55 56 57 : H I J K L M N O P Q R S T U V W |
| 10... | 2194.192957 | Hex   | 16    | 58 59 5A 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D : X Y Z a b c d e f g h i j k l m |
| 10... | 2194.192961 | Hex   | 16    | 6E 6F 70 71 72 73 74 75 76 77 78 79 7A 30 31 32 : n o p q r s t u v w x y z 0 1 2 |
| 10... | 2194.192964 | Hex   | 16    | 33 34 35 36 37 38 39 41 42 43 44 45 46 47 48 49 : 3 4 5 6 7 8 9 A B C D E F G H I |
| 10... | 2194.192968 | Hex   | 16    | 4A 4B 4C 4D 4E 4F 50 51 52 53 54 55 56 57 58 59 : J K L M N O P Q R S T U V W X Y |
| 10... | 2194.192973 | Hex   | 16    | 5A 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F : Z a b c d e f g h i j k l m n o |
| 10... | 2194.192976 | Hex   | 16    | 70 71 72 73 74 75 76 77 78 79 7A 30 31 32 33 34 : p q r s t u v w x y z 0 1 2 3 4 |
| 10... | 2194.192980 | Hex   | 16    | 35 36 37 38 39 41 42 43 44 45 46 47 48 49 4A 4B : 5 6 7 8 9 A B C D E F G H I J K |

Aufgrund der internen Rechenzeiten können ankommende Signale mit 115 kBits/s nicht ohne Verluste wieder seriell ausgegeben werden. Die Übermittlung der Daten vom Eingang auf den Ausgang erfolgt hier mit Hilfe eines FIFO-Puffers. Wenn der Puffer voll ist, wird dessen Eingang 100 ms lang abgeschaltet. Die ankommenden Botschaften während dieser Zeit gehen somit für die serielle Ausgabe verloren. Die Ausgabe der Daten auf den CAN ist davon aber nicht betroffen.

## 1.7. Leuchtanzeige

Der STC verfügt über zwei Leuchtdioden die den momentanen Betriebszustand anzeigen.

Die rote Leuchtdiode ist im Normalbetrieb ununterbrochen an.

Werden Fehler auf dem CAN festgestellt, gibt diese einen Blinkcode (kurze Unterbrechungen des Leuchtsignals) aus.



Ist für eine einstellbare Zeit kein RS232-Byte empfangen worden, so blinkt die Leuchtdiode zweimal kurz nach einer langen Pause auf.



Sind die „EIN“ und „AUS“-Phase der roten LED gleich lang, so ist die Standardkonfiguration im EEprom gespeichert.



Die grüne Leuchtdiode blinkt bei je nach konfigurierter Ausgabe entweder bei einer ankommenden RS232-Botschaft, bei der CAN-Ausgabe, bei einer digitalen oder analogen Messung auf.

## 1.8. Sleepmode

Der STC verfügt über zwei verschiedene Sleepmodes.

### **Sleepmode 1:**

In diesem Modus misst der STC die Spannung am Analogeingang 1. Fällt die Spannung unter einen konfigurierbaren Wert, gibt der STC CAN-Botschaften mit der anliegenden Spannung aus. Die Stromaufnahme in diesem Modus beträgt ca. 1500 – 1800µA.

### **Sleepmode 2:**

Stellt der STC für eine einstellbare Zeit kein RS232-Signal fest, so wechselt er in den Standby-Mode. Durch einen Flankenwechsel auf der RS232-Leitung wacht der STC auf und arbeitet mit der zuvor verwendeten Konfiguration. Die Stromaufnahme in diesem Modus beträgt ca. 350 – 450µA.

Während aktiver RS232-Kommunikation und solange die Konfigurationssoftware (CCOview) in Betrieb ist, wechselt der STC **nicht** in den Sleepmode.

## 1.9. Hinweise:

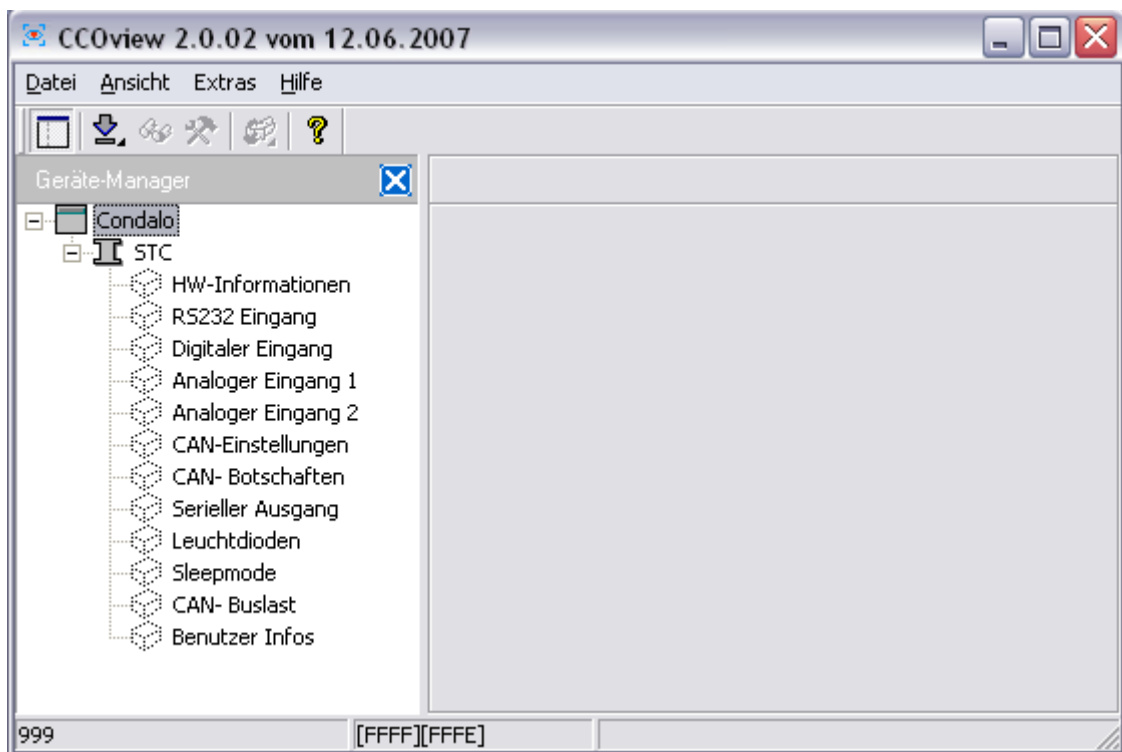
- Der STC verfügt über keinen eigenen Busabschluss. Es ist deshalb darauf zu achten, dass bei einer Aufzeichnung mit dem Datenlogger ein Busabschluss von 120 Ohm zwischen CAN-High und CAN-Low geschaltet wird.
- Der CAN-Empfänger muss das Acknowledge für die CAN-Botschaft senden (sofern dies kein anderer Teilnehmer dieses Busses übernimmt), damit der STC seine Botschaften absetzen kann.
- Wird von der Empfangs-UART des STC ein Frameerror festgestellt, so wird eine CAN-Botschaft mit einem Identifier, der um 4 höher ist als der eingestellte, ausgegeben.
- Bei einem Pufferüberlauf (nicht alle Botschaften können aufgrund zu hoher Buslast nicht rechtzeitig abgesetzt werden) wird eine CAN-Nachricht mit einem Identifier, der um 2 höher ist als der eingestellte mit der Anzahl der verlorenen Bytes ausgegeben.
- Wird für die Verbindung des PC mit dem STC kein standardisiertes serielles 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.
- Die fehlerfreie Nutzung eines USB-RS232-Dongle ist erst ab der Firmwareversion 2.0 gewährleistet.

## 2. Konfiguration:

### 2.1. Software: CCOview

Mit Hilfe der Software „CCOview“ können die Einstellungen des STC verändert, die Firmware upgedatet, und die laufende Kommunikation auf dem Bildschirm mitverfolgt werden.

Ist der STC 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.

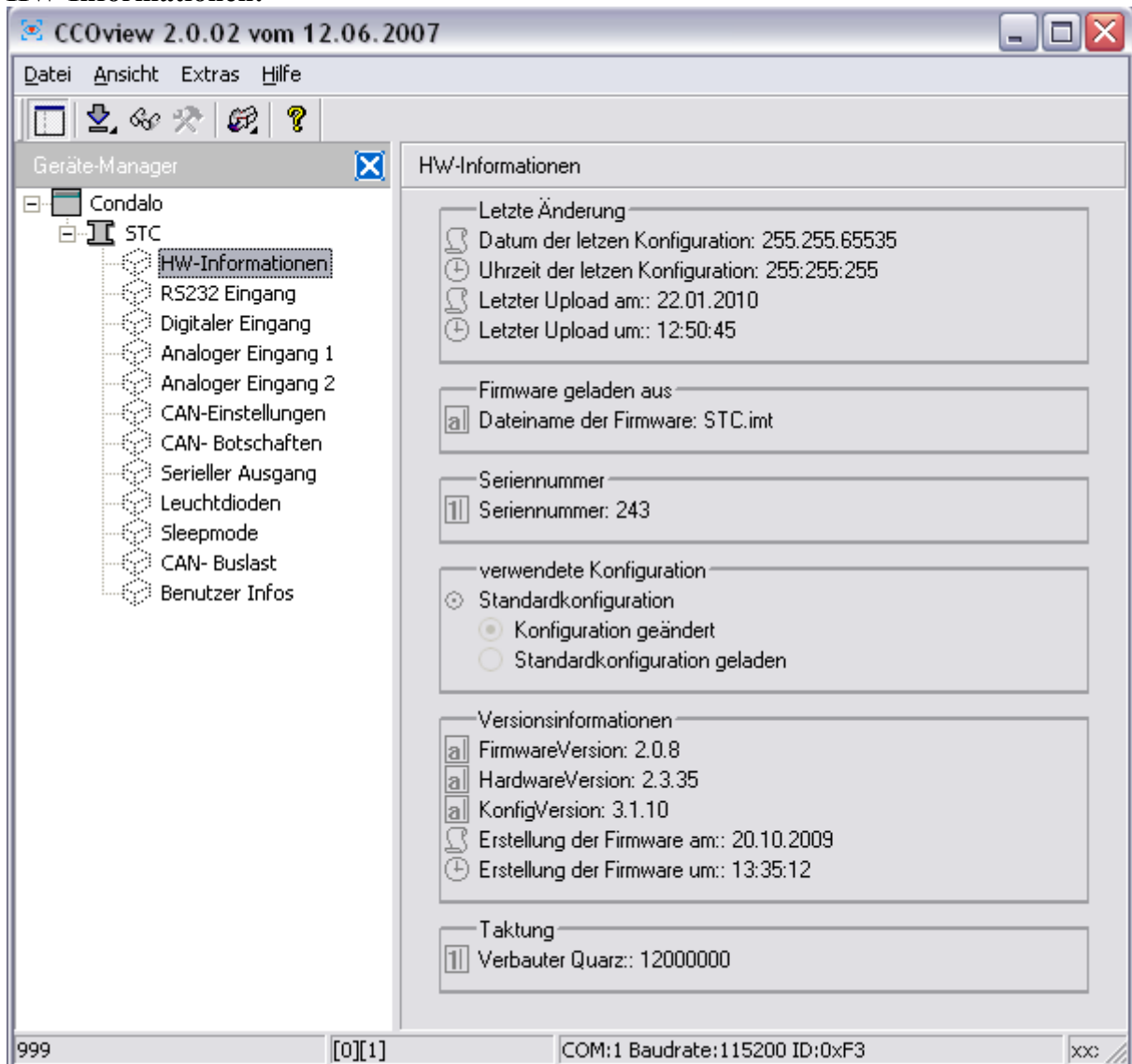
Die einzelnen Konfigurationsmöglichkeiten sind in den verschiedenen Zweigen des Konfigurationsbaumes untergebracht.

An diese Software können gleichzeitig mehrere Geräte angeschlossen werden. Dazu werden die verfügbaren COM-Ports immer wieder abgefragt.

Die dabei nicht benötigten COM-Ports, können im Menü unter „Extras → Einstellungen“ ab- bzw. ausgewählt werden.



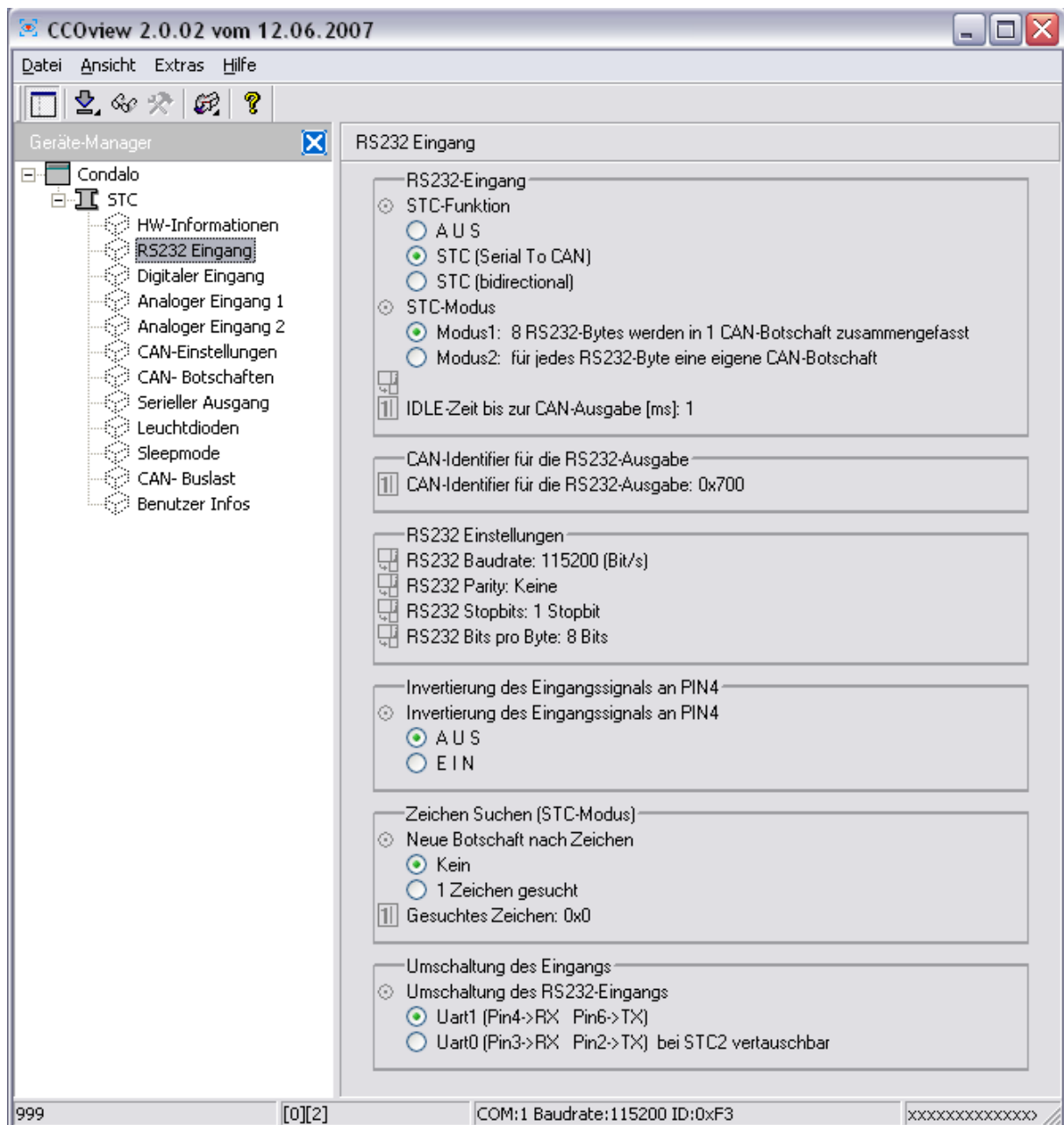
## HW-Informationen:



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

Mögliche Einstellungen sind:

### RS232 Eingang:



Einstellungen:

Funktion:

AUS,

STC (ankommende RS232-Bytes werden in die Datenbytes einer CAN-Nachricht mit konfigurierbarer CAN-ID kopiert und versendet) und

STC-bidirektional (Datenbytes, die in ankommenden CAN-Nachrichten enthalten sind, werden auf die RS232 gesendet → keine CAN-ID und Längenangabe)

Modus (in welcher Form die Daten auf den CAN gesendet werden)

Timeout (Zeit, nach der das Ende einer RS232-Botschaft erkannt wird -> Senden der CAN-Botschaft auch wenn z.B. noch keine 8 Datenbytes gesammelt wurden)

Identifizier für die CAN-Ausgabe

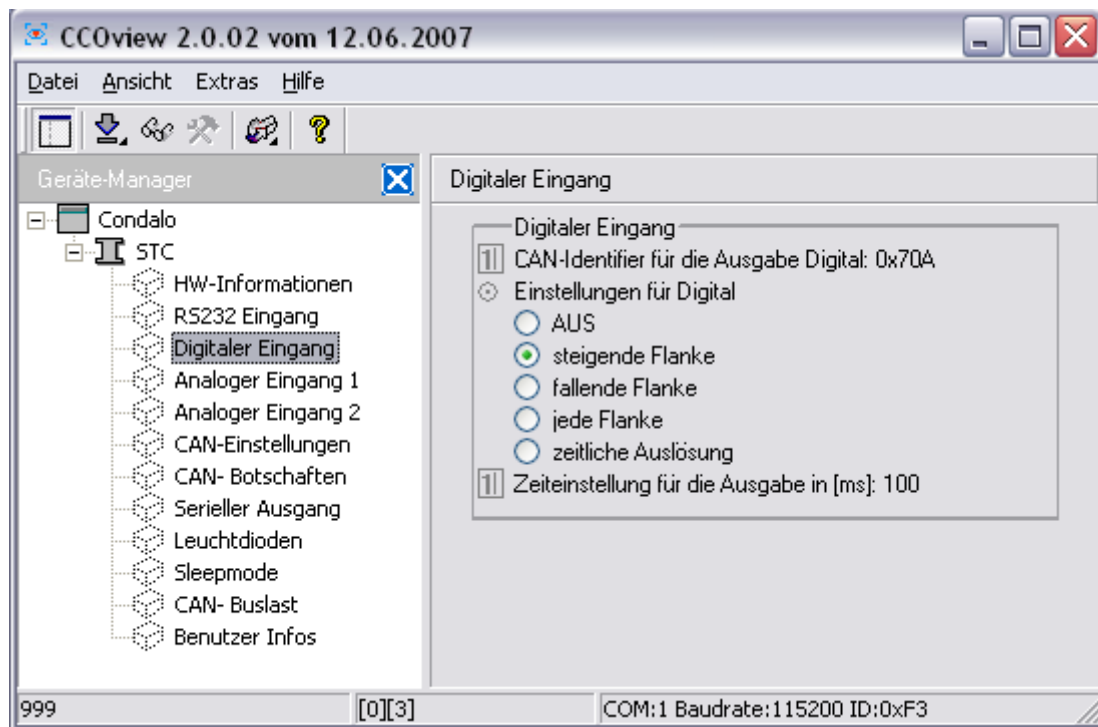
RS232-Baudrate, RS232-Parität, RS232-Stopbits, RS232 Bits pro Byte

Invertierung des Eingangssignals(z.B. für ein Signal ohne RS232-Transceiver)

Beginnen einer neuen CAN-Botschaft nach einem gesuchten Zeichen

Umschaltung der Aufzeichnungs-UART (Aufzeichnung auch auf der Konfigurations-UART möglich; Zeitverzug bis zum Beginn der Aufzeichnung beachten; keine Invertierung des Eingangssignals möglich)

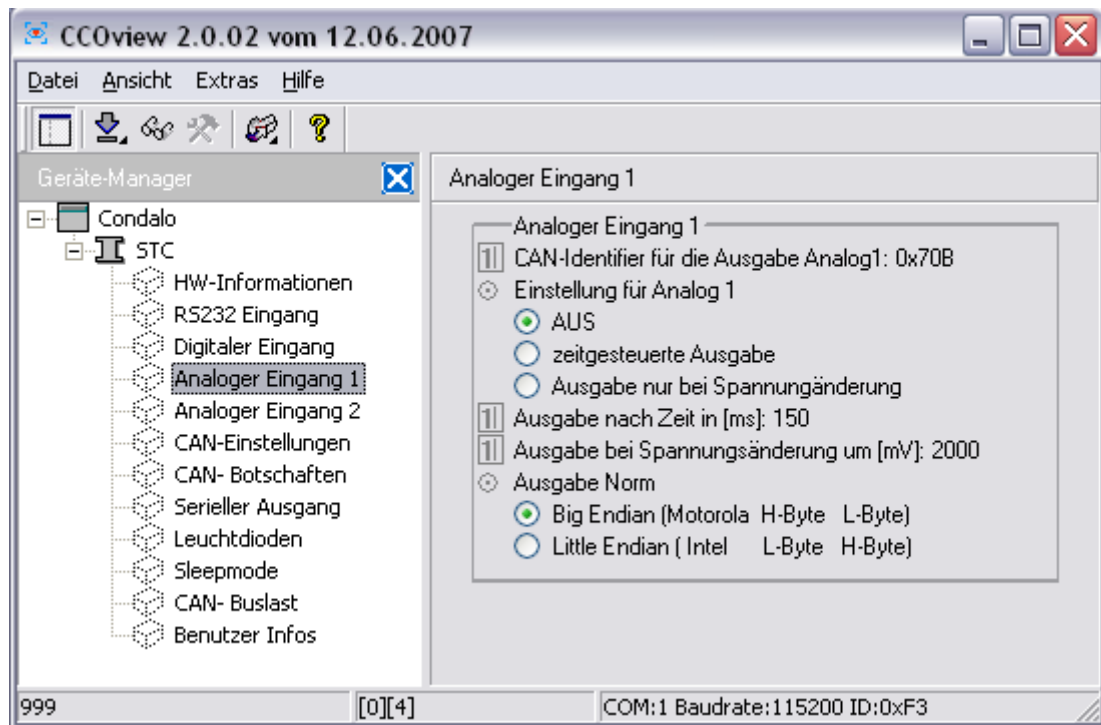
## Digitaler Eingang:



Einstellungen:

Identifizier für die CAN-Ausgabe,  
Auslöseereignis für die Ausgabe,  
Zeit in ms für die zyklische Messung

## Analoger Eingang:



## Einstellungen:

Identifizier für die CAN-Ausgabe des Analogwertes

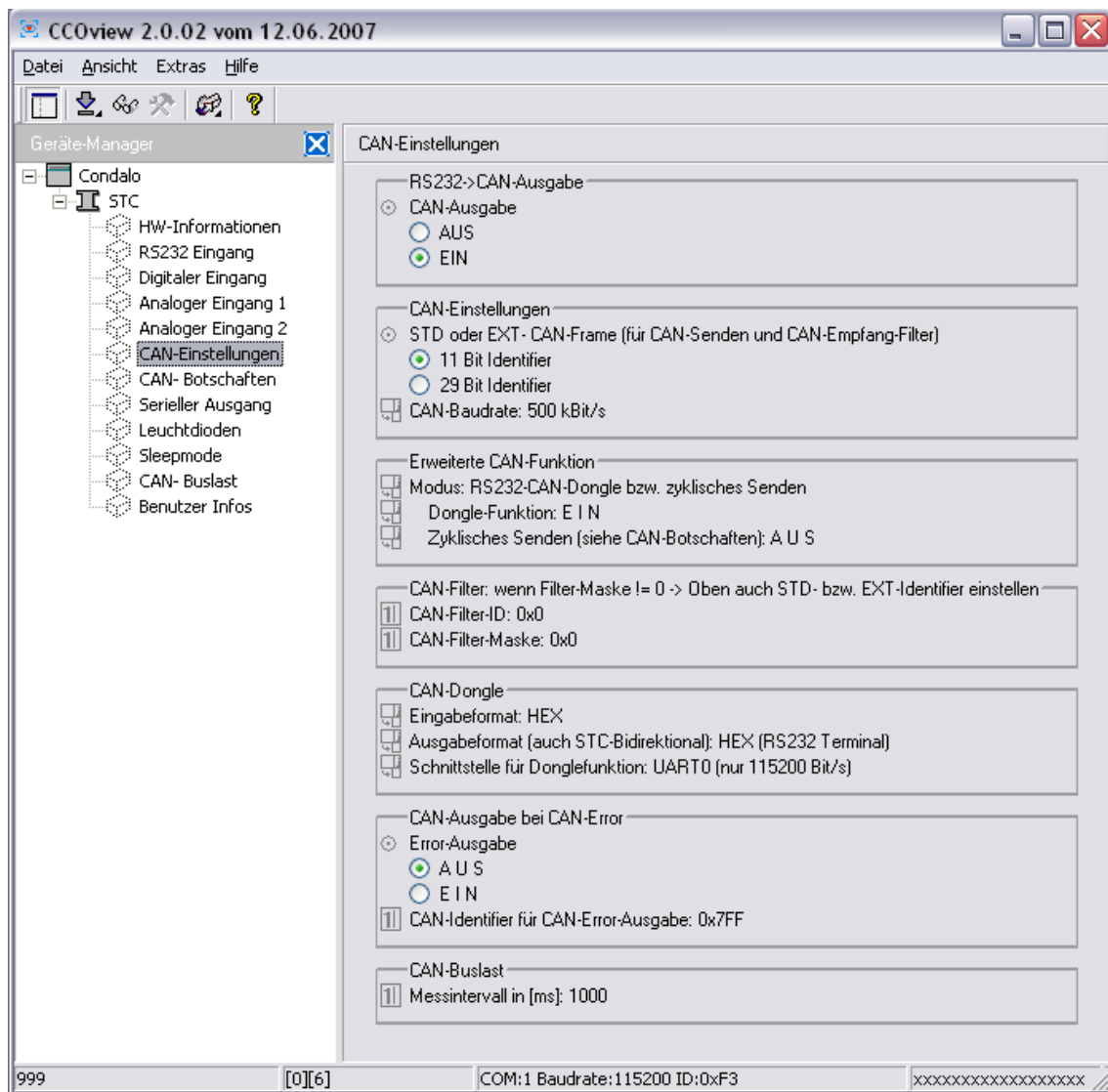
Auslöseereignis für die Ausgabe

Zeit in ms für die zyklische Messung

Spannung in mV, bei deren Änderung zur vorherigen Ausgabe eine CAN-Ausgabe erfolgen soll.

Ausgabe-Norm für den Messwert.

## CAN Ausgang:



## Einstellungen:

CAN-Ausgabe EIN bzw. AUS

Normaler oder extended Identifier

CAN-Baudrate

Erweiterte CAN-Funktion: STC, Dongle u. automatisches Senden oder CAN-Buslast.

Getrenntes EIN/AUS-schalten für Dongle- und AutoSenden-Modus

FilterID für die Dongle-Funktion (Code)

FilterID für die Dongle-Funktion (Maske)

Eingabeformat (wie die Daten dem Dongle eingegeben werden sollen → HEX oder ASCII)

Ausgabeformat (wie die Daten des Dongle ausgegeben werden sollen → HEX oder ASCII)

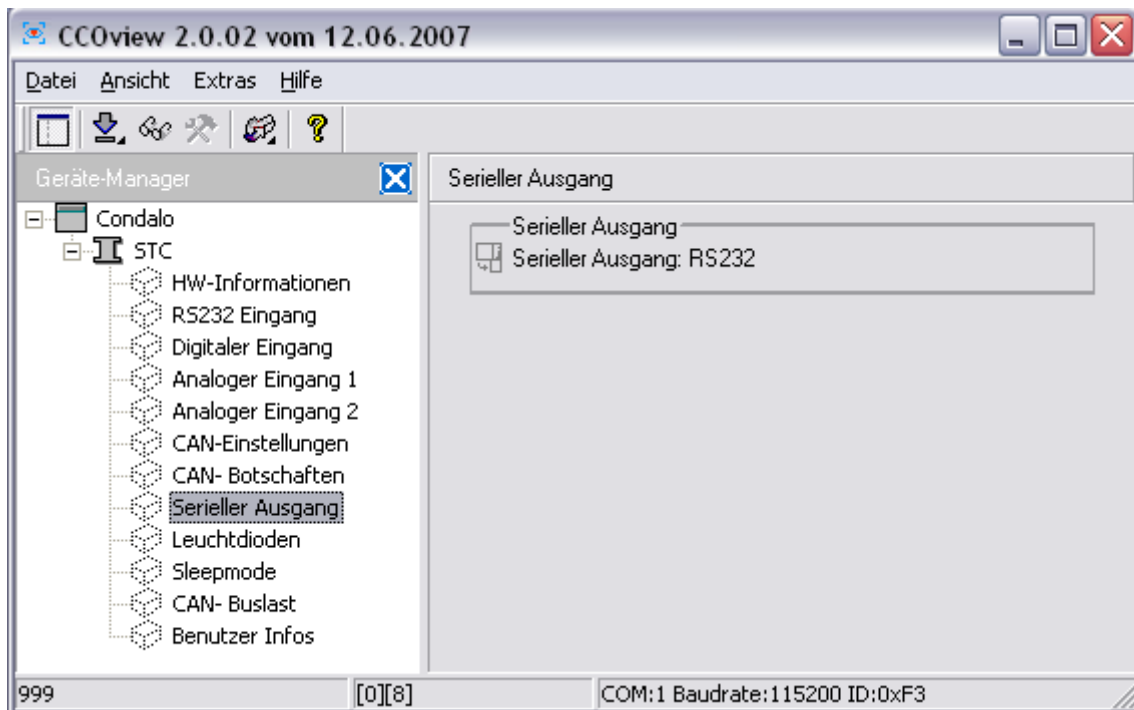
RS232-Schnittstelle für die DongleFunktion bzw. Buslastausgabe.

CAN-Ausgabe einer gesonderten Botschaft, nachdem ein Fehler auf dem CAN festgestellt wurde.

Identifier für die CAN-Error-Ausgabe

Messintervall für die Buslastmessung

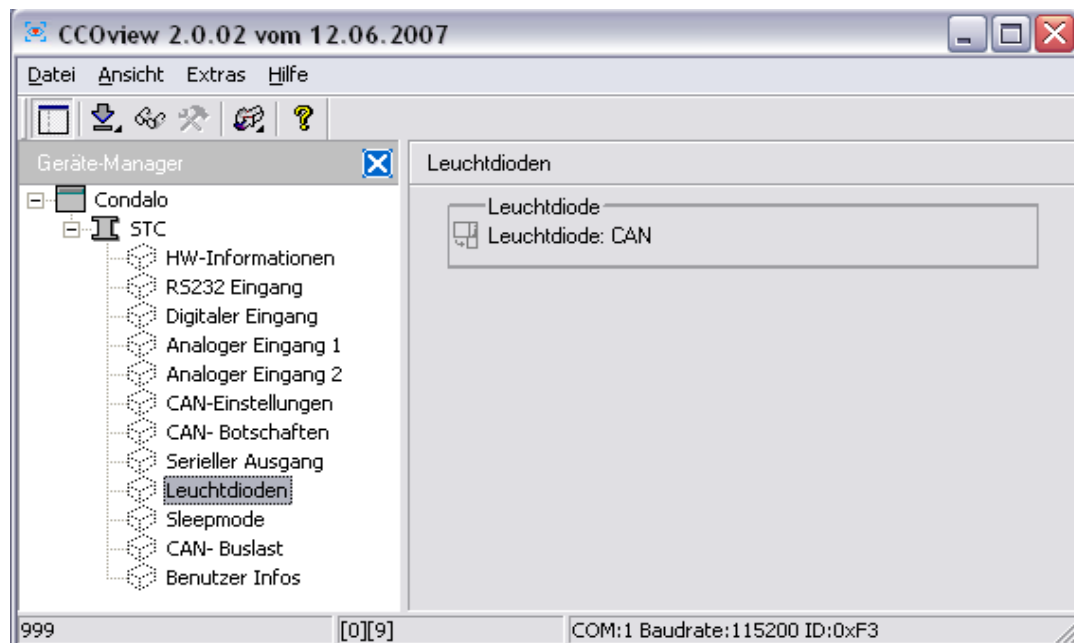
## Serieller Ausgang:



## Einstellungen:

Im seriellen Trace können RS232, Digitale oder Analoge Werte dargestellt werden.

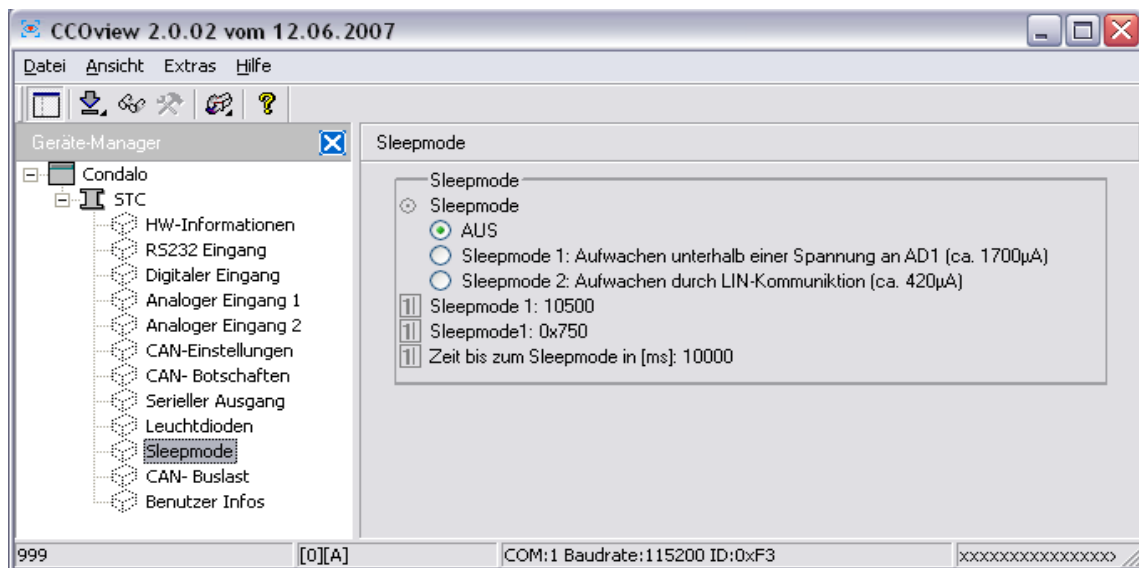
## Leuchtdioden:



## Einstellungen:

Ereignis, bei dem die Grüne LED aufleuchten soll.

## Sleepmode:



## Einstellungen:

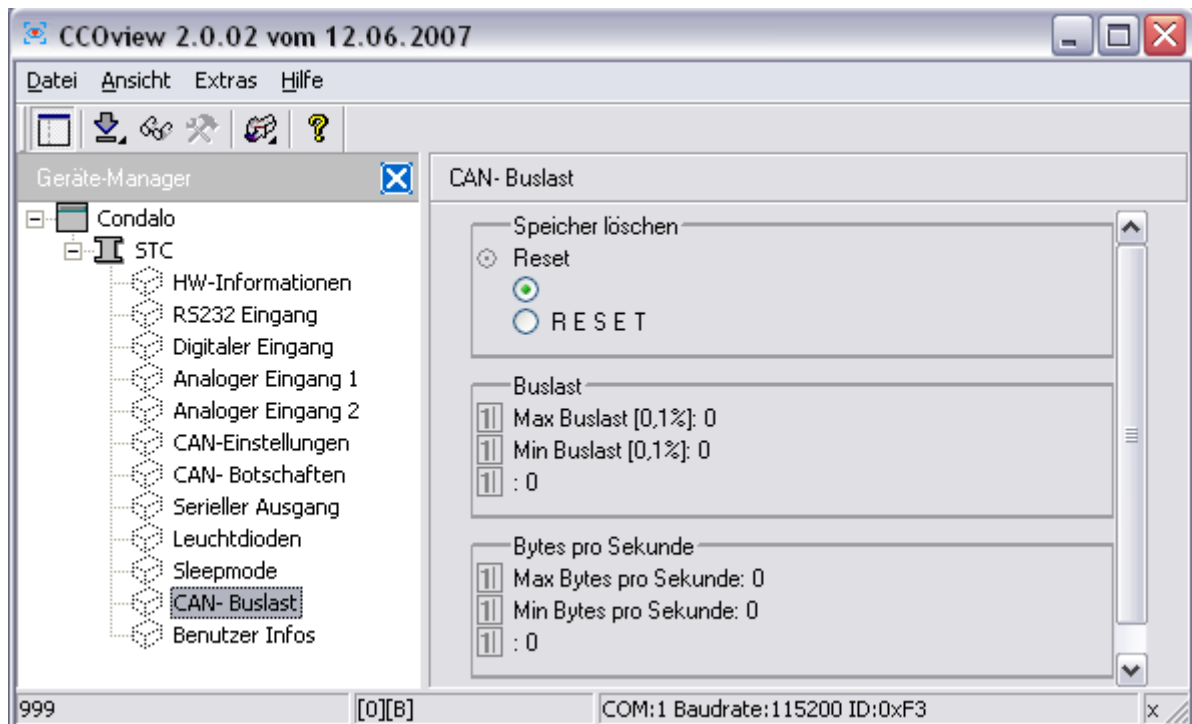
Auswahl des Sleepmodes.

Spannung, unterhalb der eine CAN-Ausgabe erfolgt (Sleepmode 1)

CAN-ID für die Spannungsausgabe (Sleepmode 1)

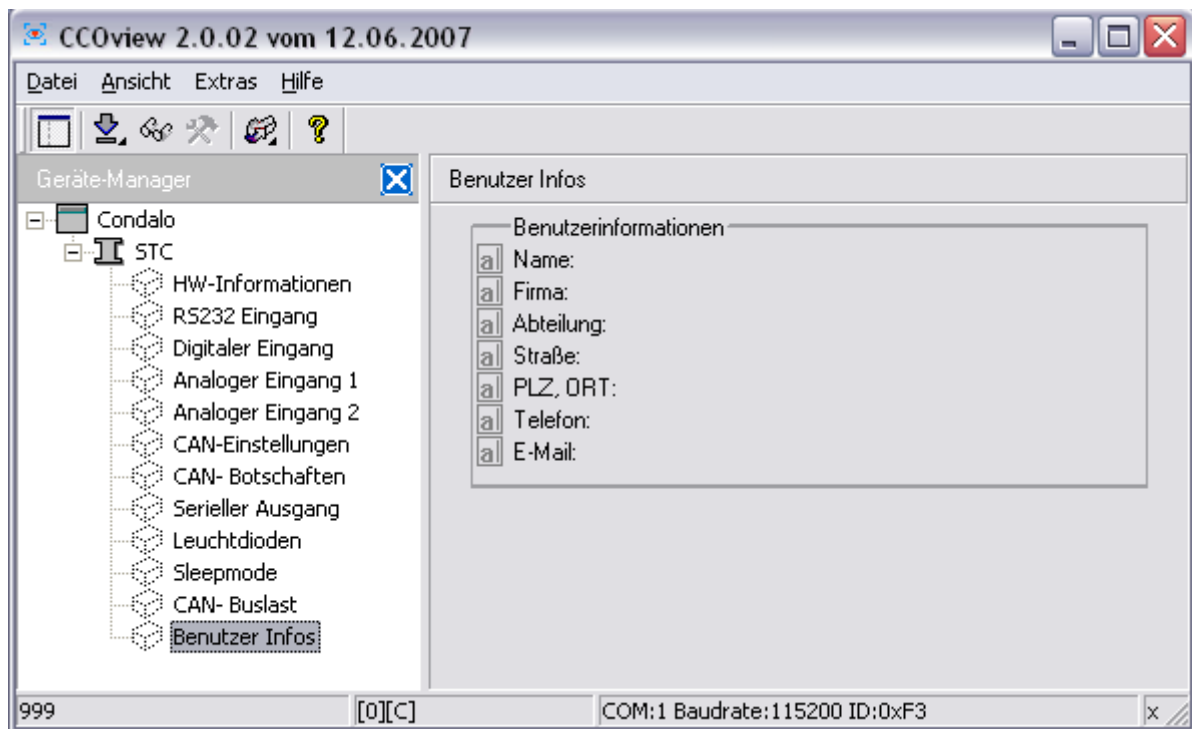
Zeit bis der LTC in den Sleepmode wechselt. (nachdem kein LIN-Signal und kein CAN-Signal mehr am LTC anliegt und das CCOview beendet ist.)

## CAN-Buslast:



Anzeige der gemessenen Werte

## Benutzer Infos:



Wurde eine Einstellung eines Zweiges verändert, so wird an das jeweilige Bauelement ein \* - Stern angehängt. Durch den Befehl „Speichern dauerhaft“ bzw. „Speichern temporär“ im Kontextmenü des Baumes kann die Änderung zum STC übertragen werden.

Die Änderungen können sowohl im EEPROM gespeichert als auch nur temporär(bis zum nächsten Reset) verändert werden.

### 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 vorgelegten symbolischen Namen, sowie auch freie Eingaben verwendet werden.

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

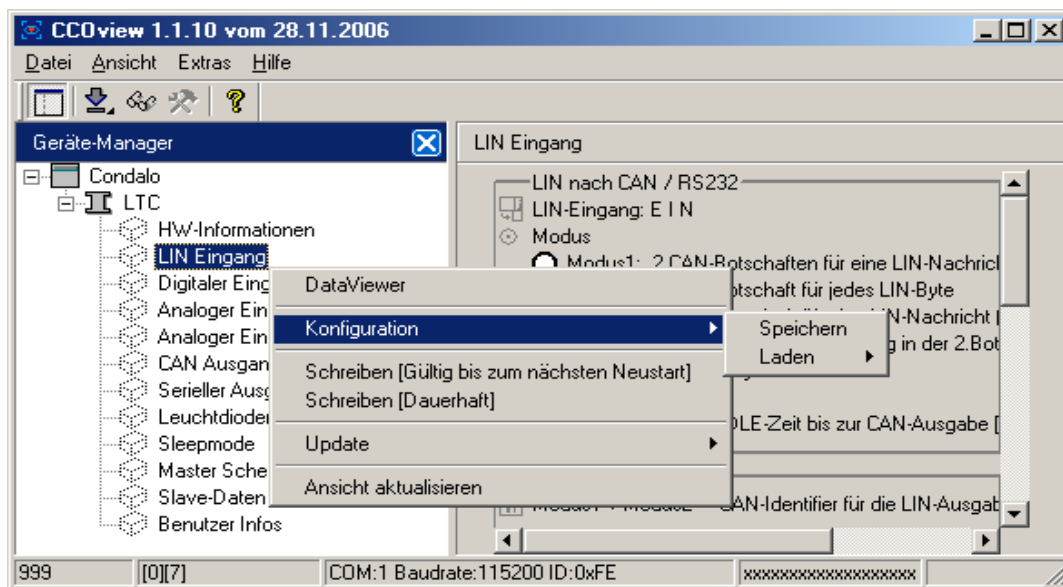


```
[Head]
Version=10
NumberOfEntries=3
[Item0]
Device=LTC
Name=Modus1 + Modus2 -- CAN-Identifer für die LIN-Ausgabe
ID=[NameA1;0x203][NameA2;100][NameA3;200][NameA4;0x123]
[Item1]
Device=LTC
Name=Modus3 -- CAN-Identifer Offset für die LIN-Ausgabe
ID=[NameB1;0x203][NameB2;100][NameB3;200][NameB4;0x123]

[Item2]
Device=BTC
Name=Identifer 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]                 |

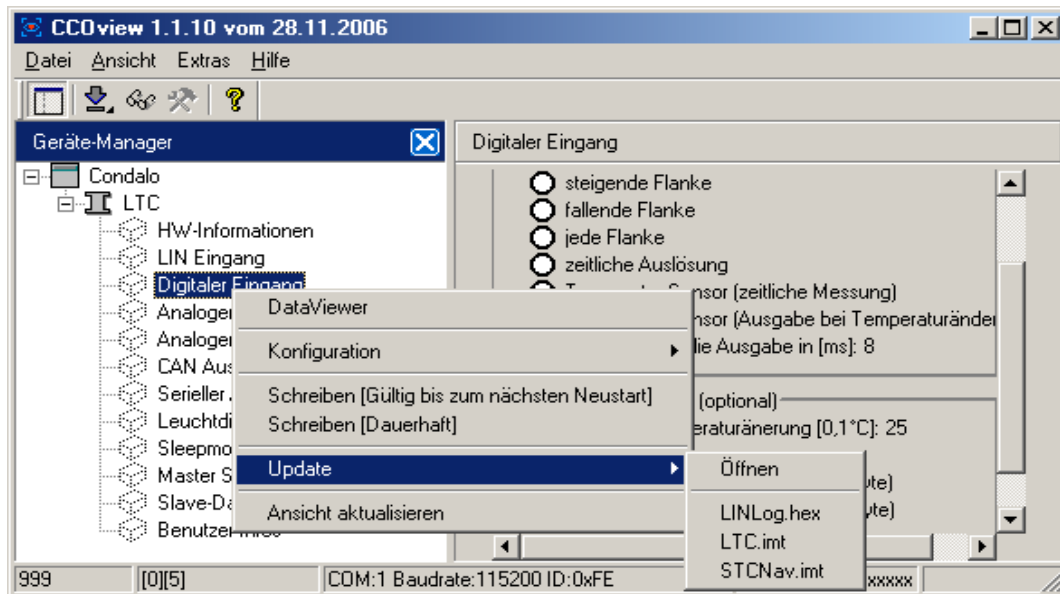
## 2.1.2. Konfiguration abspeichern



Die Verschiedenen Einstellungen des STC lassen sich in einem Konfigurationsfile abspeichern. Damit ist es möglich, verschiedenen Geräten die gleiche Einstellung zuzuweisen. Im Kontextmenü des Konfigurationsbaumes können die zuvor abgespeicherten Daten ausgewählt und auf den STC geladen werden.

## 2.2. Firmwareupdate:

Die Firmware („\*.hex“ bzw. „\*.imt“ Datei) muss in den selben Order kopiert werden, in dem sich auch die Programmdatei des Konfigurationstools befindet.



Im Kontextmenü des Konfigurationsbaumes kann mit dem Befehl "Update" eine neue Firmware auf den STC geladen werden. Die gewünschte Firmware wird im Untermenü dieses Befehls ausgewählt.

### 3. Technische Daten

#### Funktionen:

RS232 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

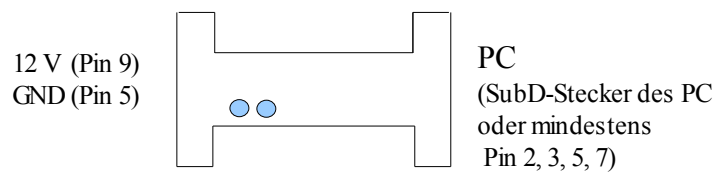
Material: Kunststoffgehäuse mit Metallüberzug

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

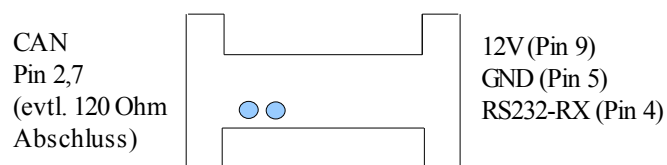
Digitaler Eingang: max. 50 V ohne zeitliche Beschränkung

## Anschlusschema:

Konfiguration:



LIN To CAN-Umsetzung:



## 4. Lieferumfang STC

- 1 STC (RS232 To CAN Konverter)
- 1 CD mit Software
- 1 Dokumentation

## 5. Impressum

### 5.1. Firmwareupdates und Programmneuheiten

finden Sie in unserem Downloadbereich unter

<http://www.condalo.de/pid130.html>

### 5.2. Telefonische Unterstützung

|               |                   |
|---------------|-------------------|
| Markus Kulzer | 08450 - 9264 - 43 |
| Zentrale      | 08450 - 9264 - 0  |

### 5.3. Fehlermeldungen und Verbesserungsvorschläge

[markus.kulzer@condalo.de](mailto:markus.kulzer@condalo.de) oder [info@condalo.de](mailto: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](mailto:info@condalo.de)  
Web: [www.condalo.de](http://www.condalo.de)