

**CD AUTOMATION SRL**

**CD VAR-IO CANOPEN BUSCOUPLER**



**Schnittstellenprotokoll  
CAN open  
9499 040 69918  
gültig ab 09/2005**

Alle Rechte vorbehalten. Ohne vorherige schriftliche Genehmigung ist der Nachdruck oder die auszugsweise fotomechanische oder anderweitige Wiedergabe dieses Dokumentes nicht gestattet.

Bei Änderungen erfolgt keine gesonderte Mitteilung.

Einschränkung der Gewährleistung:

Es wird keine Gewähr für die vollständige Richtigkeit des Inhalts übernommen, da sich trotz aller Sorgfalt Fehler nie ganz vermeiden lassen. Für Hinweise sind wir jederzeit dankbar.

---

|   |          |
|---|----------|
| <b>CDvario</b> .....  | <b>1</b> |
| 1. Einleitung.....  | 4        |
| 1.1 Allgemeines: .....  | 4        |
| 1.2 Technische Daten .....  | 4        |
| 1.3 Predefined Master/Slave Connection Set .....                            | 5        |
| 1.4 Inbetriebnahme des CD vario am CAN Bus: .....                           | 6        |
| 2. Service-Daten-Objekte (SDO) .....  | 7        |
| 3. Prozeß-Daten-Objekte (PDO) .....   | 9        |
| 3.1 Anwendung .....   | 9        |
| 3.2 Einstellung der Kommunikationsparameter für Empfangs-PDOs.....          | 9        |
| 3.2.1. Einstellung für Empfangs-PDO über CAN-Bus .....                      | 9        |
| 3.2.2. Einstellung für Empfangs-PDO über Engineering-Tool BlueControl®..... | 10       |
| 3.3 Einstellen der Kommunikationsparameter für Sende-PDOs .....             | 10       |
| 3.3.1. Einstellung für Sende-PDO über CAN-Bus .....                         | 11       |
| 3.3.2. Einstellung für Sende-PDO über BlueControl®.....                     | 12       |
| 3.4 Einstellung der Mappingparameter der PDOs.....                          | 13       |
| 3.4.1. Allgemeines zur Einstellung der Mappingparameter .....               | 13       |
| 3.4.2. Standardverfahren.....   | 13       |
| 3.4.3. Multiplexverfahren bei kanalweiser Übertragung.....                  | 15       |
| 3.4.4. Multiplexverfahren mit indizierter Übertragung .....                 | 16       |
| 3.5 Lesen der Mappingparameter über CAN-Bus .....                           | 18       |
| 4. NMT-Dienste .....  | 19       |
| 5. SYNC .....   | 20       |
| 6. LSS-Dienste .....  | 21       |
| 6.1 Umschalteprotokolle (Switch Mode Protocols).....                        | 21       |
| 6.1.1. Globale Umschaltung in den Operation/Konfiguration-Mode .....        | 21       |
| 6.1.2. Selektives Umschaltung in den Operation/Konfiguration-Mode .....     | 21       |
| 6.2 Konfigurationsprotokolle (Configuration Protocols) .....                | 22       |
| 6.2.1. Konfiguration der Knoten-ID.....                                     | 22       |
| 6.2.2. Konfiguration der Baudrate .....                                     | 22       |
| 6.3 Anfrageprotokolle (Inquiry Protocols).....                              | 23       |
| 6.3.1. Anfrage der Vendor ID .....  | 23       |
| 6.3.2. Anfrage des Product Code .....                                       | 23       |
| 6.3.3. Anfrage der Revision Number .....                                    | 23       |
| 6.3.4. Anfrage der Serial Number.....                                       | 24       |
| 6.3.5. Anfrage der Knoten ID.....   | 24       |
| 6.4 Identifikationsprotokolle (Identification Protocols) .....              | 24       |
| 6.4.1. Identifizieren von nicht konfigurierten Teilnehmern .....            | 24       |
| 7. Speichern und Laden von Parametern (Store/Restore) .....                 | 25       |
| 7.1 Nichtflüchtiges Speichern von Parametern (Index 1010) .....             | 25       |
| 7.2 Laden von gespeicherten Parametern (Index 1011) .....                   | 25       |
| 8. Heartbeat.....   | 25       |
| 8.1 Konfiguration als Heartbeat Producer (Index 1017): .....                | 25       |
| 8.2 Konfiguration als Heartbeat Consumer (Index 1016): .....                | 25       |
| 9. Konfigurierung und Parametrisierung über den CAN-Bus .....               | 26       |
| 10. CAN Physical Layer .....  | 28       |
| 10.1 ISO 11898-2 Knoten .....   | 28       |
| 10.2 Baudraten und Buslängen.....   | 29       |
| 10.3 Praktische Buslängen .....   | 29       |
| 10.4 Kabel-Parameter.....   | 30       |

## 1. Einleitung

### 1.1 Allgemeines:

Der CD vario besitzt eine CAN-Schnittstelle, über die dem Regler alle zum Betrieb notwendigen Daten (Parameter- und Konfigurationsdaten) gesendet werden können. Ebenso kann ein entsprechender Master (PC oder Steuerung) alle Prozeßdaten abrufen.

Gemäß "CANopen" sind die Zugriffe in SDO (Service Data Objects) und PDO (Process Data Objects) gegliedert. Dabei sind SDOs zur Konfiguration und Parametrisierung von Busteilnehmern vorgesehen und PDOs für die üblichen Betriebswerte.

Die im CD vario implementierten Kommunikationsdienste basieren auf dem "CANopen Application Layer and Communication Profile" (CIA Draft Standard 301).

### 1.2 Technische Daten

- ÷ 2 SDO-Kanäle
- ÷ 4 Sende-PDOs (TPDO), synchron/asynchron
- ÷ 4 Empfangs-PDOs (RPDO), asynchron
- ÷ Mapping der PDOS
  - ÷ Standardverfahren
  - ÷ Multiplexverfahren bei kanalweiser Übertragung
  - ÷ Multiplexverfahren mit indizierter Übertragung
- ÷ Datenformat: Float, Fixpoint1 und Integer
- ÷ Baudraten von 10kBaud bis 1MBaud
- ÷ NMT-Dienste
- ÷ LSS-Dienste
- ÷ Heartbeat
- ÷ Store/Restore
- ÷ Predefined Master/Slave Connection Set

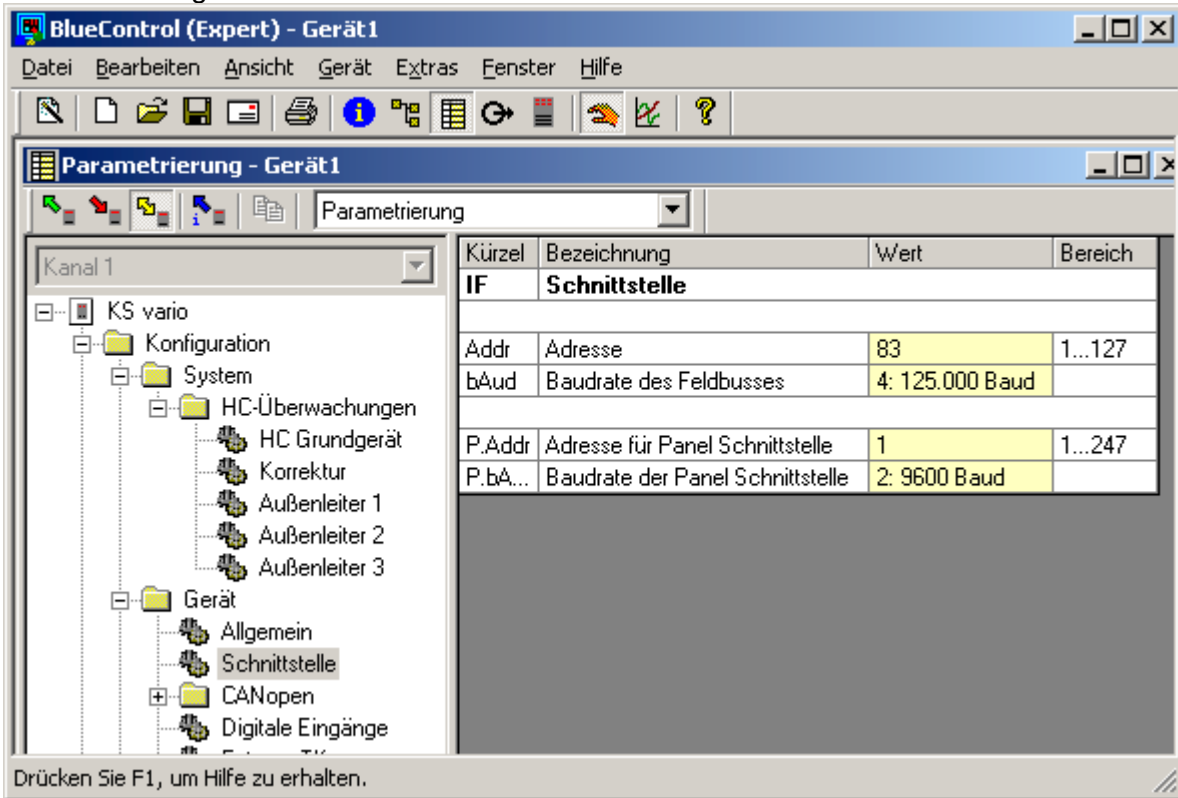
### 1.3 Predefined Master/Slave Connection Set

Die COB-IDs werden im CD vario gemäß Predefined Master/Slave Connection Set vergeben. Eine Ausnahmen bilden die PDOs, bei ihnen kann durch gesonderte Einstellung hiervon abgewichen werden.

| Object                            | Functions-Code | COB_ID Range  | Index in the object listing |
|-----------------------------------|----------------|---------------|-----------------------------|
| <b>Broadcast-Nachrichten</b>      |                |               |                             |
| NMT                               | 0000           | 0             | -                           |
| Sync                              | 0001           | 0x80          | 0x1005h                     |
| <b>Punkt-zu-Punkt_Nachrichten</b> |                |               |                             |
| TPDO1                             | 0011           | 0x181-0x0x1FF | 0x1800                      |
| RPDO1                             | 0100           | 0x201-0x27F   | 0x1400                      |
| TPDO2                             | 0101           | 0x281-0x2FF   | 0x1801                      |
| RPDO2                             | 0110           | 0x301-0x37F   | 0x1401                      |
| TPDO3                             | 0111           | 0x381-0x3FF   | 0x1802                      |
| RPDO3                             | 1000           | 0x401-0x47F   | 0x1402                      |
| TPDO4                             | 1001           | 0x481-0x4FF   | 0x1803                      |
| RPDO4                             | 1010           | 0x501-0x57F   | 0x1403                      |
| Default-TSDO                      | 1011           | 0x581-0x5FF   | 0x1200                      |
| Default-RSDO                      | 1100           | 0x601-0x67F   | 0x1200                      |
| LSS                               | 1111           | 0x7E4/0x7E5   |                             |

### 1.4 Inbetriebnahme des CDvario am CAN Bus:

Das Gerät hat im Auslieferungszustand die Knotenadresse auf 255 und die Baudrate auf 20 kBaud. Mit der Knotenadresse 255 startet das Gerät im LSS-Mode. Wird dies nicht gewünscht, so ist eine Knotenadresse im Bereich von 1 – 127 einzustellen. Diese Einstellungen können mit Hilfe des BlueControl® über die BlueControl®-Schnittstelle oder den LSS-Diensten über den CAN-Bus erfolgen



Nach Aufstart des CDvario (mit gültiger Knotenadresse) befindet sich das Gerät im "Pre Operational Mode". Man kann über die CAN-Schnittstelle mittels SDOs kommunizieren. Nach der Bus- und Gerätekonfiguration muß der Master den CDvario in den "Operational Mode" setzen (NMT Start) um auch PDOs nutzen zu können. Wie PDOs genutzt werden, muß vorab konfiguriert werden. Man unterscheidet "asynchrone PDOs" und "synchrone PDOs". Asynchrone PDOs werden bei geänderten Prozeßwerten gesendet. Synchrone PDOs werden auf Anforderung durch den Master gesendet. Dazu sendet der Master einen SYNC Frame.

## 2. Service-Daten-Objekte (SDO)

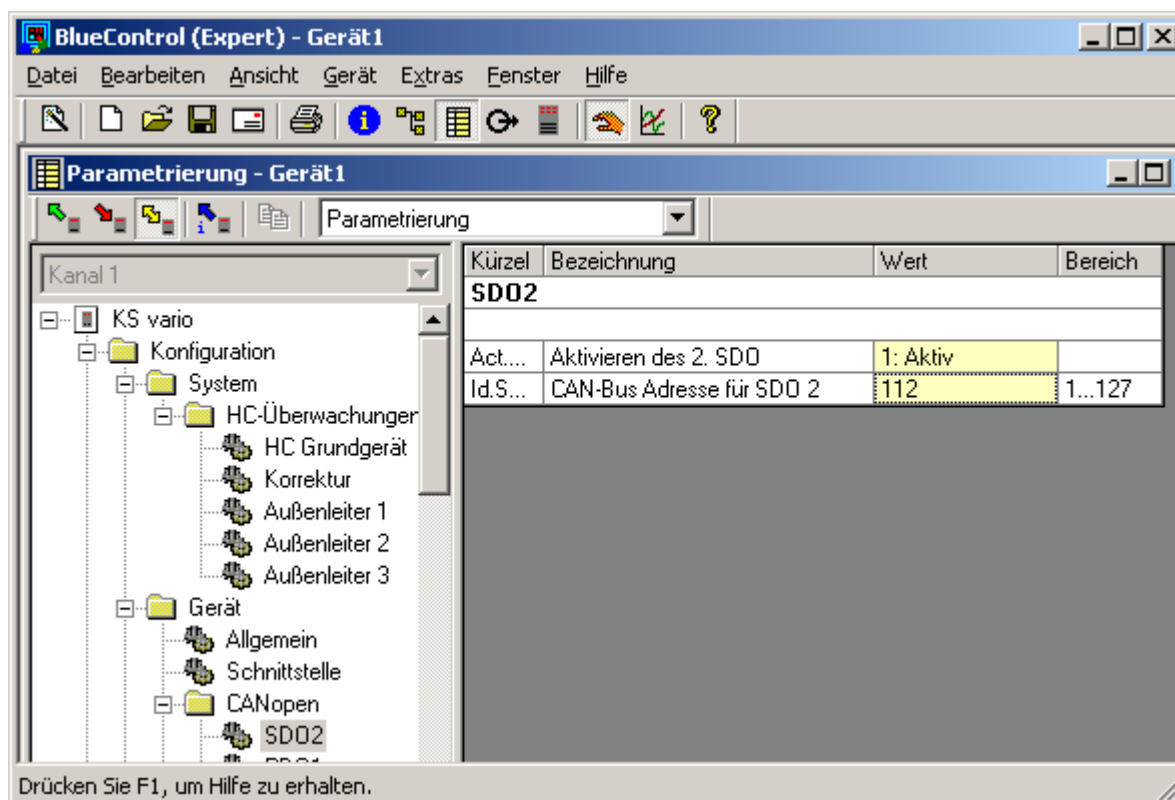
Ein SDO ist logisch festgelegt auf genau zwei Partner - 1 Master und 1 Slave. Da der CDvario 2 SDO-Kanäle besitzt, ist es möglich, von zwei unterschiedlichen SDO-Mastern gleichzeitig auf den CDvario zuzugreifen. Die Datenübertragung per SDO ist ein bestätigter Dienst.

Mit Hilfe der SDOs können Konfiguration und Parametrisierung des Reglers über Einzelzugriffe erfolgen. Aber auch alle Bediendaten, die üblicherweise über PDOs übertragen werden, können über SDOs angesprochen werden.

Die verwendete COB-ID richtet sich nach der eingestellten Geräteadresse:

Für den Sende-SDO liegt dieser Wert im Bereich von 1409 – 1535 (0x581 – 0x5FF).

Für den Empfangs-SDO im Bereich von 1537 – 1663 (0x601 – 0x67F)



Der 2. SDO-Kanal kann mit Hilfe des BlueControl® eingestellt werden. Im gezeigten Beispiel wird der 2. SDO-Kanal aktiviert und es wird die CAN-Bus-Adresse 112 benutzt, so daß die COB-Ids 1520 (0x5f0) bzw. 1648 (0x670) verwendet werden. Die Adressen der beiden SDO-Kanäle müssen sich unterscheiden. Die COB-ID des 1. SDO-Kanals wird durch die eingestellte Knotenadresse definiert („Predefined Connection Set“).

Eine Konfiguration über CAN-Bus ist ebenfalls möglich. Sie erfolgt über die Objekte mit dem Index/Subindex 0x1201/1 und 0x1201/2 .

Gerätespezifische Befehle, wie Betriebsartenwechsel (Pre-Operation Mode, Operation Mode usw.), können nur über den 1. SDO-Kanal erfolgen.

Mittels SDOs können maximal 4 Byte Nutzdaten übertragen werden. Die restlichen 4 Byte werden wie folgt genutzt:

- 1 Byte für den Command: Art der Übertragung
- 2 Byte für den Index: Objektkennung (z.B. 0x2476 für Sollwert)
- 1 Byte für den Subindex: Kanalnummer

Im folgenden wird jeweils ein Beispiel eines SDO-Datenframes für Lesen und für Schreiben gegeben.

**Beispiel 1. Ein Master sendet einen Sollwert an einen CDvario mit Knotenadresse 4:**

Der Aufbau der Nachricht ist wie folgt:

- COB\_ID = SDO an Knoten 4 = 0x600 + 4 = 0x604
- CMD = Schreibzugriff = 0x2B = 0x2B
- INDEX = Sollwert = 0x2476 = 0x2476
- SUBINDEX = z.B. Regler 1 = 01 = 0x01
- WERT = 30,0°C ·) = 300 ·) = 0x12C

·) Fixpoint1- Darstellung, d.h. 1 feste Nachkommastelle.

|                      | COB-ID | LEN  | 8 data bytes |       |      |      |       |      |      |      |
|----------------------|--------|------|--------------|-------|------|------|-------|------|------|------|
|                      |        |      | Cmd          | Index |      | Sub. | Daten |      |      |      |
| Anforderung (Master) | 0x604  | 0x08 | 0x2B         | 0x76  | 0x24 | 0x01 | 0x2c  | 0x01 | 0x00 | 0x00 |
| Antwort (CDvario)    | 0x584  | 0x08 | 0x60         | 0x76  | 0x24 | 0x01 | 0x00  | 0x00 | 0x00 | 0x00 |

**Beispiel 2. Ein Master liest einen Istwert von einem CDvario mit Knotenadresse 2:**

Der Aufbau der Nachricht ist wie folgt:

- COB\_ID = SDO an Knoten 2 = 0x600 + 2 = 0x602
- CMD = Lesezugriff = 0x40 = 0x40
- INDEX = Sollwert = 0x2441 = 0x2441
- SUBINDEX = z.B. Regler 3 = 03 = 0x03

|                      | COB-ID | LEN  | 8 data bytes |       |      |      |       |      |      |      |
|----------------------|--------|------|--------------|-------|------|------|-------|------|------|------|
|                      |        |      | Cmd          | Index |      | Sub. | Daten |      |      |      |
| Anforderung (Master) | 0x602  | 0x08 | 0x40         | 0x41  | 0x24 | 0x03 | 0x00  | 0x00 | 0x00 | 0x00 |
| Antwort (CDvario)    | 0x582  | 0x08 | 0x4B         | 0x41  | 0x24 | 0x03 | 0xFA  | 0x00 | 0x00 | 0x00 |

WERT = 0x00FA = 250 = 25,0°C



### 3. Prozeß-Daten-Objekte (PDO)

Im CDvario werden 4 Empfangs-PDOs (RPDO) und 4 Sende-PDOs (TPDO) zur Verfügung gestellt. PDOs sind unbestätigte Dienste und haben maximal 8 Byte Nutzdaten. Alle eintreffenden Nachrichten werden, in der Reihenfolge wie sie eingetroffen sind, abgearbeitet. Eine vorrangige Bearbeitung spezieller Nachrichten ist nicht möglich. Das Senden und Empfangen von PDOs ist nur im „Operational Mode“ möglich.

#### 3.1 Anwendung

Der Datentransfer über PDOs ist zum Beispiel dann sinnvoll, wenn Istwerte und Statuswerte des Reglers ständig im Master aktuell gehalten werden sollen. In diesem Fall wird man asynchrone PDOs verwenden, die bei Wertänderungen auf dem Regler die jeweiligen geänderten Daten automatisch zum Master senden. Ein anderes Beispiel wäre die zyklische Anforderung von Istwerten durch eine Bedieneinheit. In diesem Fall kann der CDvario PDOs synchronisiert zu einem Messtakt (SYNC-EVENT) senden.

#### 3.2 Einstellung der Kommunikationsparameter für Empfangs-PDOs

Für den Empfangs-PDO (RPDO) ist nur die COB-ID einzustellen. Prinzipiell kann für den PDO eine COB-ID aus dem Bereich 385 – 1407 (0x181 – 0x57F) benutzt werden.

Im „Predefined Connection Set“ wird die eingestellte Geräteadresse zur Definition der COB-ID herangezogen. Bei einer Geräteadresse 5 gelten hier folgende COB-IDs

- ÷ RPDO1 517 (0x205)
- ÷ RPDO2 773 (0x305)
- ÷ RPDO3 1029 (0x405)
- ÷ RPDO4 1285 (0x505)

Der CDvario kann PDOs nur asynchron empfangen.

##### 3.2.1. Einstellung für Empfangs-PDO über CAN-Bus

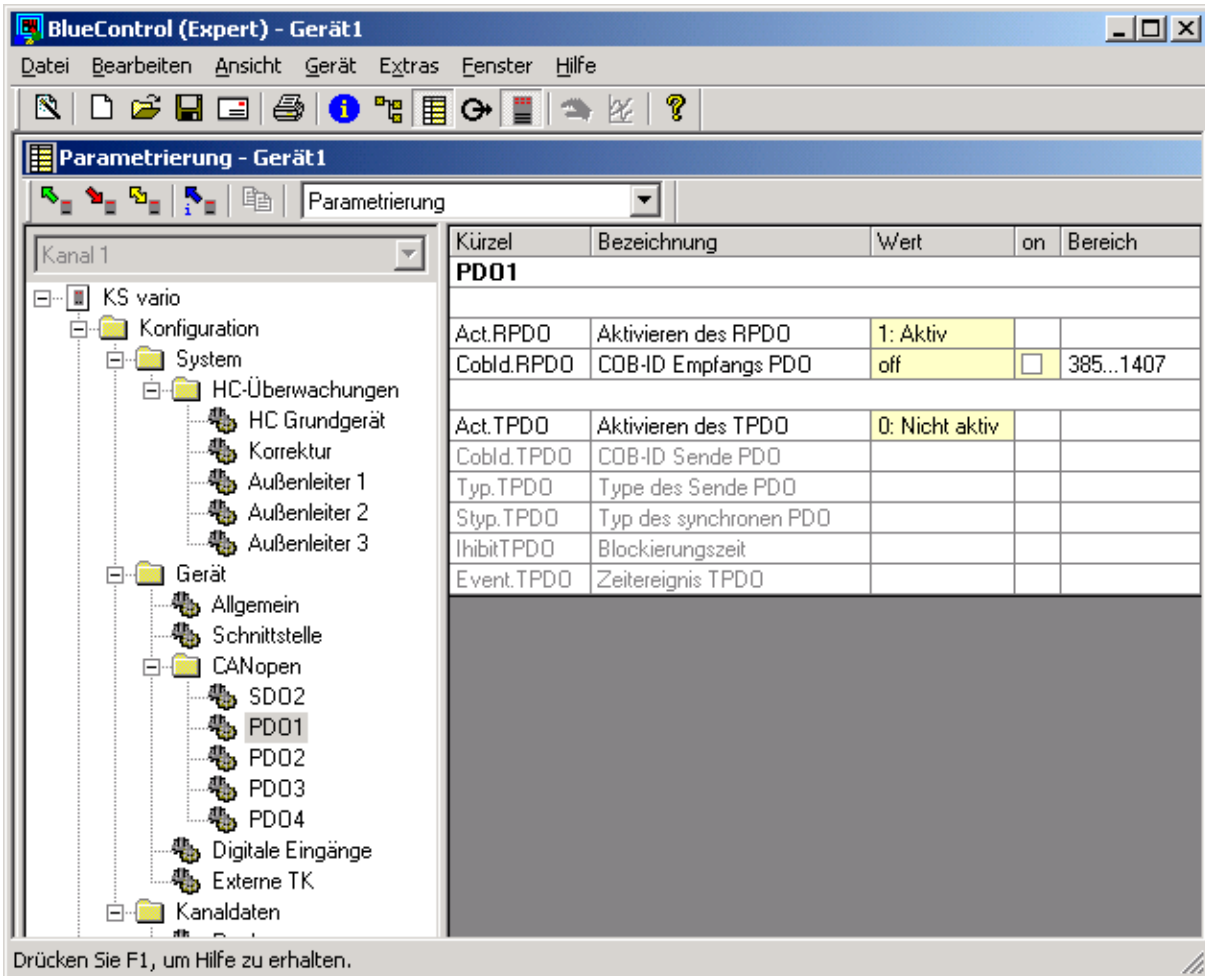
Die Einstellung über CAN-Bus erfolgt über folgende Objekte:

- ÷ RPDO1 über Index 0x1400 und Subindex 1.
  - ÷ RPDO2 über Index 0x1401 und Subindex 1.
  - ÷ RPDO3 über Index 0x1402 und Subindex 1.
  - ÷ RPDO4 über Index 0x1403 und Subindex 1.
- Ist das Höchstwertigste Bit der 32 Bit Date gesetzt, so ist der RPDO nicht aktiv.

### 3.2.2. Einstellung für Empfangs-PDO über das Engineering-Tool BlueControl®

Im gezeigten Beispiel wird der RPDO 1 aktiviert.

Da die COB-ID abgeschaltet ist, benutzt das Gerät das „Predefined Connection Set“, das bei einer eingestellten Geräteadresse 5 in diesem Fall 517 (0x205) ist.



Soll eine eigene COB-ID ausgewählt werden, so ist die Eingabe durch einen Haken in der Spalte on zu aktivieren. Danach kann eine Zahl im Bereich von 385 – 1407 in der Spalte Wert eingetragen werden.

### 3.3 Einstellen der Kommunikationsparameter für Sende-PDOs

Für den Sende-PDO (TPDO) sind mehrere Daten einzustellen. Dieses sind:

÷ COB-ID.

Prinzipiell kann für den PDO eine COB-ID aus dem Bereich 385 – 1407 (0x181 – 0x57F). Im „Predefined Connection Set“ wird die eingestellte Geräteadresse zur Definition der COB-ID herangezogen. Bei einer Geräteadresse 5 gelten hier folgende COB-IDs

- TPDO1      389    (0x185)
- TPDO2      645    (0x285)
- TPDO3      901    (0x385)
- TPDO4      1157   (0x485)

- ÷ Betriebsart Synchron oder Asynchron
  - 1 – 240 Synchron. Das Gerät erzeugt einen TPDO nach dem Empfang des 1. bzw. 240 SYNC-Signals
  - 255 Asynchron. Das Gerät erzeugt einen TPDO wenn sich ein Wert im zu übertragenden TPDO geändert hat.
- ÷ Inhibit Time. Gibt die Zeit an nach der frühestens ein TPDO wieder gesendet werden darf. Um zu verhindern, dass bei schnellen Änderungen der Werte im TPDO sehr viele Nachrichten generiert werden, kann durch die Einstellung dieses Wertes eine Wartezeit definiert werden. Die Einstellung erfolgt in Schritten von 0,1 ms. Wird der Wert auf 0 gesetzt, so ist keine Wartezeit aktiv.
- ÷ Event Timer. Gibt die Zeit an nach der spätestens ein TPDO gesendet wird, auch wenn sich kein Wert geändert hat. Die Einstellung erfolgt in Schritten von 1 ms. Wird der Wert auf 0 gesetzt, so ist kein automatisches Senden aktiv. Werden gemultiplexte Daten übertragen (kanalweise oder per Index), so ist der Event Timer nicht aktiv.

### 3.3.1. Einstellung für Sende-PDO über CAN-Bus

Die Einstellung über CAN-Bus erfolgt über folgende Objekte:

- ÷ COB-ID.
  - TPDO1 über Index 0x1800 und Subindex 1.
  - TPDO2 über Index 0x1801 und Subindex 1.
  - TPDO3 über Index 0x1802 und Subindex 1.
  - TPDO4 über Index 0x1803 und Subindex 1.Ist das Höchstwertigste Bit der 32 Bit Date gesetzt, so ist der TPDO nicht aktiv.
- ÷ Betriebsart Synchron oder Asynchron
  - TPDO1 über Index 0x1800 und Subindex 2.
  - TPDO2 über Index 0x1801 und Subindex 2.
  - TPDO3 über Index 0x1802 und Subindex 2.
  - TPDO4 über Index 0x1803 und Subindex 2.
- ÷ Inhibit Time
  - TPDO1 über Index 0x1800 und Subindex 3.
  - TPDO2 über Index 0x1801 und Subindex 3.
  - TPDO3 über Index 0x1802 und Subindex 3.
  - TPDO4 über Index 0x1803 und Subindex 3.
- ÷ Event Timer
  - TPDO1 über Index 0x1800 und Subindex 5.
  - TPDO2 über Index 0x1801 und Subindex 5.
  - TPDO3 über Index 0x1802 und Subindex 5.
  - TPDO4 über Index 0x1803 und Subindex 5.

Der Subindex 4 wird nicht verwendet.

### 3.3.2. Einstellung für Sende-PDO über BlueControl®

Im gezeigten Beispiel wird der TPDO 2 aktiviert.

Als COB-ID wurde die feste Adresse 385 (0x181) zugewiesen.

Es handelt sich um einen asynchronen PDO. Die Inhibit Time wurde auf 1 Sekunde gestellt, so dass frühestens nach dieser Zeit ein erneuter TPDO generiert wird.

Der Event Timer steht auf 25 Sekunden. Es wird somit spätestens nach 25 Sekunden ein TPDO erzeugt, auch wenn sich keine Daten ändern.

| Kürzel       | Bezeichnung            | Wert                | on                                  | Bereich    |
|--------------|------------------------|---------------------|-------------------------------------|------------|
| <b>PDO2</b>  |                        |                     |                                     |            |
| Act.RPDO     | Aktivieren des RPDO    | 0: Nicht aktiv      |                                     |            |
| CobId.RPDO   | COB-ID Empfangs PDO    |                     |                                     |            |
| Act.TPDO     | Aktivieren des TPDO    | 1: Aktiv            |                                     |            |
| CobId.TPDO   | COB-ID Sende PDO       | 385                 | <input checked="" type="checkbox"/> | 385...1407 |
| Typ.TPDO     | Type des Sende PDO     | 1: Asynchroner TPDO |                                     |            |
| Styp.TPDO    | Typ des synchronen PDO |                     |                                     |            |
| Inhibit.TPDO | Blockierungszeit       | 10000               |                                     | 0...30000  |
| Event.TPDO   | Zeitereignis TPDO      | 25000               |                                     | 0...30000  |

### 3.4 Einstellung der Mappingparameter der PDOs

#### 3.4.1. Allgemeines zur Einstellung der Mappingparameter

Die Mappingparameter definieren welche Daten in den PDOs kommuniziert werden. Diese Parameter können nur über das BlueControl® eingestellt werden. Ein Lesen über den CAN-Bus ist möglich.

Es können nur Parameter und Signale verwendet werden. Bei den Daten der RPDOs muß es sich um beschreibbare Daten handeln. Das BlueControl® stellt durch die Auswahlmöglichkeiten sicher, dass diese Bedingungen eingehalten werden.

Es können Daten sowohl im Integer- als auch im Floatformat übertragen werden. Daten, die im Integerformat kommuniziert werden und im Gerät als Floatwert behandelt werden, werden als Fixpoint1-Date (Wert ist mit dem Faktor 10 multipliziert) übertragen.

Die Anzahl der in einem PDO übertragbaren Daten hängt sowohl von dem Datentyp (Integer- oder Floatformat) als auch von der Art der Übertragung (Standard- oder Multiplexverfahren) ab. Werden zu viele Daten ausgewählt, so sind deren Einträge in rot gekennzeichnet. Diese eingestellten Daten werden vom BlueControl® nicht zum Gerät übertragen.

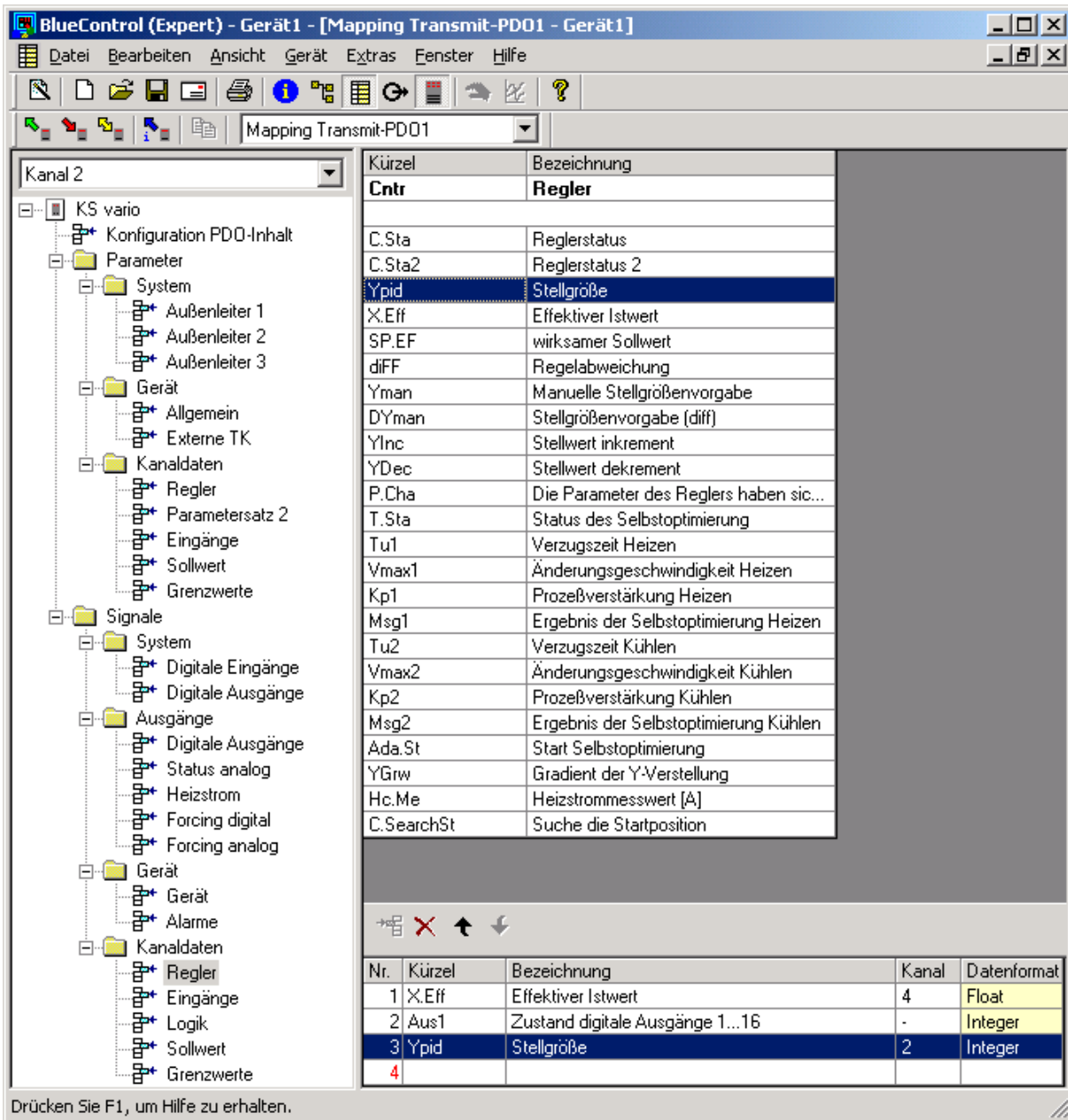
Es gibt verschiedene Möglichkeiten, die Daten im PDO zu übermitteln.

#### 3.4.2. Standardverfahren

In diesem Fall werden dem PDO feste SDO-Parameter zugewiesen. Die Festlegung erfolgt durch den SDO-Parameter und, wenn es sich um Daten aus dem Kanalbereich handelt, zusätzlich durch den Subindex. Die Länge der Daten hängt davon ab, ob diese aus dem Integer-Bereich 0x2000 (2 Byte) oder aus dem Float-Bereich 0x3000 (4 Byte) gewählt wurden. Eine Mischung zwischen 2 Byte und 4 Byte Daten ist möglich. In die 8 Byte, die ein PDO zur Verfügung stellt, können somit maximal 4 Daten übermittelt werden. Es müssen nicht alle Daten des PDO belegt werden.

|                 |  |
|-----------------|--|
| Integer 1 Low   | Date 1: im Regler als Integer definiert und wird als Integergröße ausgelesen |
| Integer 1 High  |  |
| Fixpoint 2 Low  | Date 2: im Regler als Float definiert und wird als Integergröße ausgelesen   |
| Fixpoint 2 High |  |
| Float 3 Byte 1  | Date 3: im Regler als Float definiert und wird als Floatgröße ausgelesen     |
| Float 3 Byte 2  |  |
| Float 3 Byte 3  |  |
| Float 3 Byte 4  |  |

Die Festlegung welche Daten zu verwenden sind erfolgt im BlueControl® im Modus „Mapping Transmit-PDO1 – 4“ bzw. „Mapping Receive PDO1 – 4“.



Im oben angegeben Beispiel werden folgende Daten dem TPDO1 zugewiesen:

- ÷ Effektiver Istwert des Regelkreises 4
- ÷ Zustand der digitalen Ausgänge 1 ... 16
- ÷ Stellgröße des Regelkreises 2

Da der Istwert im Floatformat übertragen wird, kann der 4. Wert nicht mehr belegt werden.

Zusätzlich gibt es die Möglichkeit die PDO-Daten im Multiplexverfahren zu übertragen.

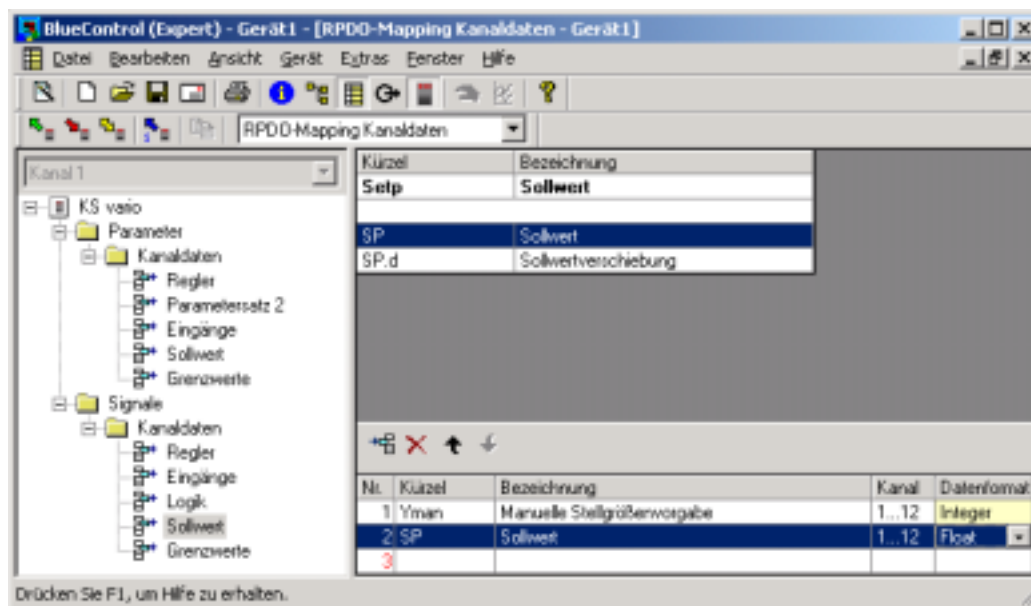
Hier stehen 2 Alternativen zur Verfügung:

### 3.4.3. Multiplexverfahren bei kanalweiser Übertragung

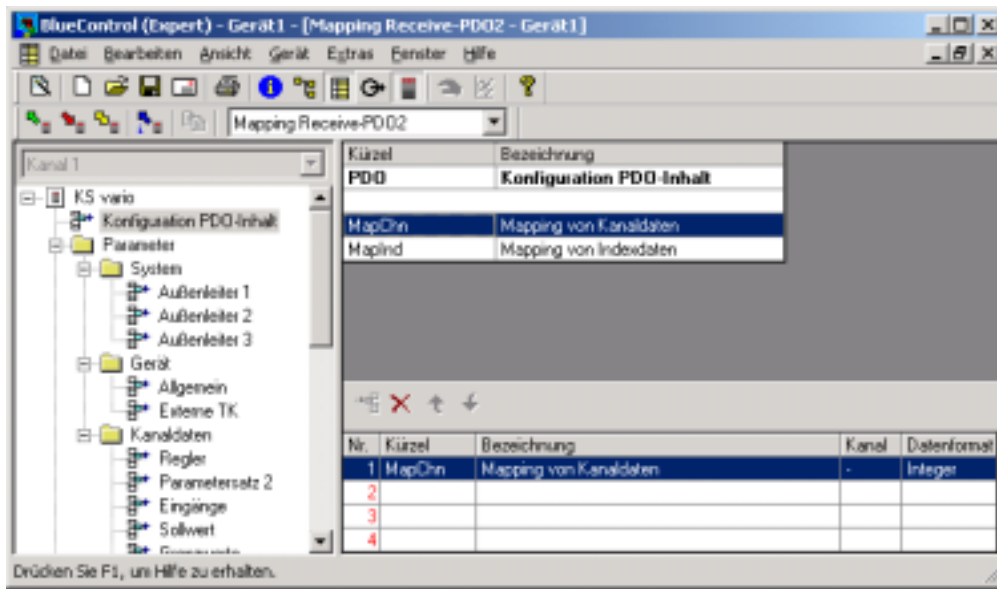
Es besteht die Möglichkeit die Daten gemultiplext und zwar kanalweise zu übertragen. Es werden maximal 3 Daten definiert, die für jeden im Gerät definierten Regelkreise übertragen werden. In den ersten 2 Byte des PDOs wird signalisiert für welchen Kanal (Regelkreis) die nachfolgenden Daten gelten. Es können sowohl Werte im Integer- als auch im Floatformat übertragen werden.

|                    |  |
|--------------------|--|
| Kanalnummer        | 1 – 30, Wird der Wert 0 übertragen, so sind die Daten nicht zu verwenden   |
| muß immer „0“ sein |  |
| Fixpoint 1 Low     | Date 1: im Regler als Float definiert und wird als Integergröße ausgelesen |
| Fixpoint 1 High    |  |
| Float 2 Byte 1     | Date 2: im Regler als Float definiert und wird als Floatgröße ausgelesen   |
| Float 2 Byte 2     |  |
| Float 2 Byte 3     |  |
| Float 2 Byte 4     |  |

Hierfür muß die Definition in 2 Schritten erfolgen. Zuerst wird festgelegt welche Daten zu übertragen sind, danach mit welchem PDO die Übertragung erfolgen soll. Nachfolgend ist gezeigt welche Einstellungen erfolgen müssen, damit diese Art der Übertragung im RPDO2 durchgeführt wird.



Die zu übertragenden Daten werden festgelegt. Als 1. Date wird die manuelle Stellgrößenvorgabe zum CDVario gesendet. In der 2. Date wird der Sollwert übermittelt. Da der Sollwert im Float-Format übertragen wird, steht die 3. Date nicht zur Verfügung. Aus der Spalte Kanal kann entnommen werden, dass Einstellungen für die Kanäle 1 – 12 erfolgen können. Diese Regelkreise sind im BlueControl® definiert. Werden Daten für die anderen Regelkreise übertragen, so werden die Werte nicht übernommen.



Danach erfolgt die Definition, welcher PDO die kanalweise, gemultiplexten Daten verwenden soll. Im oben gezeigten Beispiel wird der RPDO2 für die Übertragung der kanalweise, gemultiplexten Daten verwendet. Durch die Verwendung dieser Übertragungsart sind alle Daten des PDO belegt.

### 3.4.4. Multiplexverfahren mit indizierter Übertragung

Es können bis zu 120 Daten definiert werden, die durch einen Index adressiert übertragen werden. Es werden nur Daten im Integerformat übertragen. Der Aufbau eines solchen PDOs sieht folgendermaßen aus:

|                    |   |
|--------------------|---|
| Index              | 1 - 118, is transferred the value 0, then the data are not to be used                 |
| "0" must always be |   |
| Fixpoint 1 Low     | DATE according to the index number.   |
| Fixpoint 1 High    | in the automatic controller as if float defined and becomes as integer value selected |
| Integer 2 Low      | DATE with index number + 1.   |
| Integer 2 High     | in the automatic controller as Integer defined and becomes as integer value selected  |
| Integer 3 Low      | DATE with index number + 2.   |
| Integer 3 High     | in the automatic controller as Integer defined and becomes as integer value selected  |

Hierfür muß die Definition in 2 Schritten erfolgen. Zuerst wird definiert welche Daten zu übertragen sind, danach mit welchem PDO die Übertragung erfolgen soll. Nachfolgend ist gezeigt welche Einstellungen erfolgen müssen, damit diese Art der Übertragung im TPDO3 durchgeführt wird.



The screenshot shows the BlueControl (Expert) software interface. The title bar reads "BlueControl (Expert) - Gerät1 - [TPDO-Mapping Indexdaten - Gerät1]". The menu bar includes "Datei", "Bearbeiten", "Ansicht", "Gerät", "Extras", "Fenster", and "Hilfe". The toolbar contains various icons for file operations and navigation. The main window is titled "TPDO-Mapping Indexdaten".

On the left, a tree view shows the device structure under "Kanal 7":

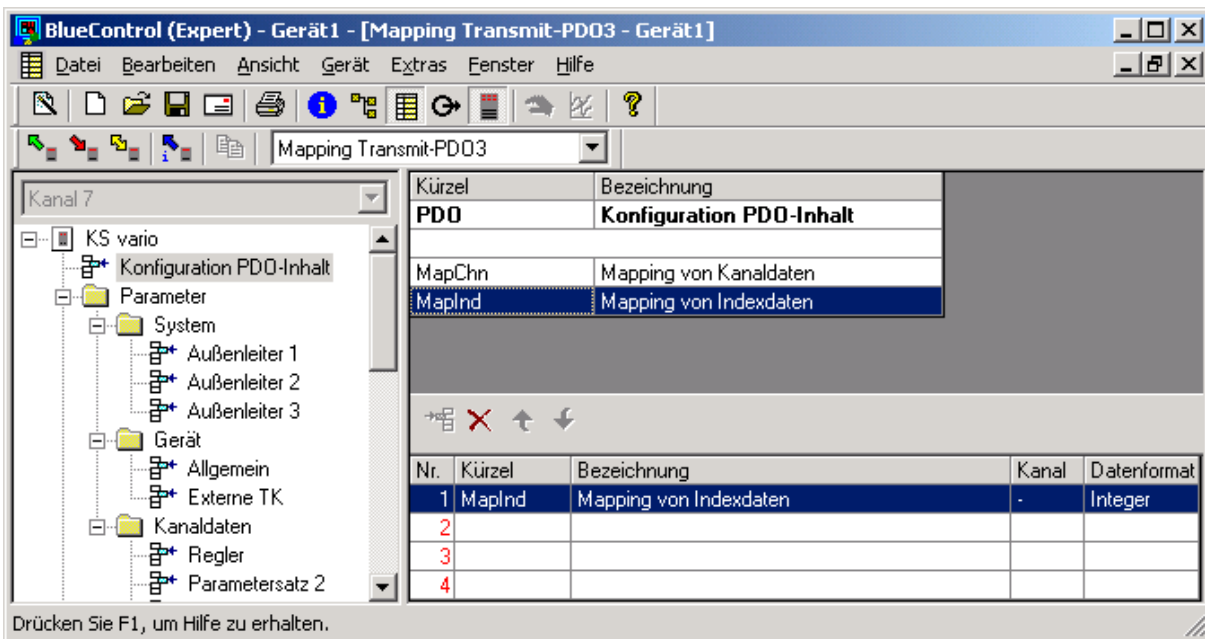
- KS vario
  - Parameter
    - System
      - Außenleiter 1
      - Außenleiter 2
      - Außenleiter 3
    - Gerät
      - Allgemein
      - Externe TK
    - Kanaldaten
      - Regler
      - Parametersatz 2
      - Eingänge
      - Sollwert
      - Grenzwerte
  - Signale
    - System
      - Digitale Eingänge
      - Digitale Ausgänge
    - Ausgänge
      - Digitale Ausgänge
      - Status analog
      - Heizstrom
      - Forcing digital
      - Forcing analog
    - Gerät
      - Gerät
      - Alarmer
    - Kanaldaten
      - Regler
      - Eingänge
      - Logik
      - Sollwert
      - Grenzwerte

Für die Datenübertragung stehen maximal 120 Daten zur Verfügung. Alle Daten werden im Integerformat übertragen. Daten die im Gerät als Float-Wert bearbeitet werden, werden als Fixpoint 1 (Date ist mit dem Faktor 10 multipliziert) übertragen. Im obigen Beispiel wurden folgende Daten für die Übertragung definiert:

Date 1 – 7 Istwert des Regelkreise 1 – 7

Date 8 – 14 Sollwert der Regelkreise 1 – 7

Date 15 Zustand der digitalen Ausgänge 1 – 16



Danach erfolgt die Definition, welcher PDO durch den Index gemultiplexten Daten verwenden soll. Im oben gezeigten Beispiel wird der TPDO3 für die Übertragung der durch den Index gemultiplexten Daten verwendet. Durch die Verwendung dieser Übertragungsart, sind alle Daten des PDO belegt.

Die Definition der Daten für kanalweisen Zugriff oder per Index werden, unabhängig vom Checkfeld "Schnittstellenparameter übertragen" immer vom BlueControl® übertragen.

### 3.5 Lesen der Mappingparameter über CAN-Bus

Das Lesen der Mappingparameter ist über CAN-Bus möglich. Es werden folgende Indizes benutzt:

- ÷ RPDO1 über Index 1600 und Subindex 1 - 3.
- ÷ RPDO2 über Index 1601 und Subindex 1 - 3.
- ÷ RPDO3 über Index 1602 und Subindex 1 - 3.
- ÷ RPDO4 über Index 1603 und Subindex 1 - 3.
- ÷ TPDO1 über Index 1A00 und Subindex 1 - 3.
- ÷ TPDO2 über Index 1A01 und Subindex 1 - 3.
- ÷ TPDO3 über Index 1A02 und Subindex 1 - 3.
- ÷ TPDO4 über Index 1A03 und Subindex 1 - 3.

Für das Mapping von Indexdaten wird der Wert 0x2FFF, für das Mapping von Kanaldaten der Wert 0x2FFE übertragen.

## 4. NMT-Dienste

Es werden folgende NMT-Diensten (COB-ID 0x00) unterstützt:

- ÷ Wechsel in den Operational Mode (Befehlscode 0x01)
- ÷ Wechsel in den Stopp Mode (Befehlscode 0x02)
- ÷ Wechsel in den Pre-Operational Mode (Befehlscode 0x80)
- ÷ Reset Kommunikationsschnittstelle (Befehlscode 0x82)
- ÷ Reset Knoten (Befehlscode 0x81)

Nach dem Aufstart des CAN-Netzwerkes ist der CDvario im Pre-OP-Mode. Das heißt, daß er nur über SDOs angesprochen werden kann. Wird der CDvario in den OP-Mode gesetzt, kann auch über die freigegebenen PDOs kommuniziert werden.

**CDvario in Operational Mode setzen:**

| COB-ID | LEN | 2 data bytes |          |
|--------|-----|--------------|----------|
|        |     | Command      | Node     |
| 0x00   | 2   | 0x01         | <Nummer> |

**CDvario in Stopp Mode setzen:**

| COB-ID | LEN | 2 data bytes |          |
|--------|-----|--------------|----------|
|        |     | Command      | Node     |
| 0x00   | 2   | 0x02         | <Nummer> |

**CDvario in Pre-Operational Mode setzen:**

| COB-ID | LEN | 2 data bytes |          |
|--------|-----|--------------|----------|
|        |     | Command      | Node     |
| 0x00   | 2   | 0x80         | <Nummer> |

**Reset Node:**

| COB-ID | LEN | 2 data bytes |          |
|--------|-----|--------------|----------|
|        |     | Command      | Node     |
| 0x00   | 2   | 0x81         | <Nummer> |

**Achtung:** Reset-Node führt zu einem Hardware-Reset des CDvario, d.h. er ist erst nach einigen Sekunden wieder über den CAN-Bus erreichbar. Alle Command-Parameter werden auf die Default-Werte zurückgesetzt.

**Reset Communication:**

| COB-ID | LEN | 2 data bytes |          |
|--------|-----|--------------|----------|
|        |     | Command      | Node     |
| 0x00   | 2   | 0x82         | <Nummer> |

Dieser Befehl wird bei einer Node-ID 0 von an alle Knoten ausgeführt, ist eine andere Node-ID angegeben führt nur der adressierte Teilnehmer den Befehl aus.

Nach dem Befehl „Reset Kommunikationsschnittstelle“ und „Reset Knoten“ befindet sich das Gerät im Pre-Operation Mode. Diesen Zustand nimmt das Gerät ebenfalls nach dem Einschalten der Spannung ein. Um eine Kommunikation über die PDOs durchführen zu können ist ein Umschalten in den Operational Mode notwendig.

**5. SYNC**

Wie bereits beschrieben werden synchrone PDO-Daten mittels SYNC-Nachrichten vom CDvario ausgewertet oder gesendet. Die entsprechenden CDvario müssen im Operation-Mode sein und synchrone PDOs müssen konfiguriert sein.

Eine SYNC-Nachricht ist ein sogenannter "Broadcast" (Nachricht an alle Teilnehmer im Netz) und wird wie folgt vom Master gesendet:

| COB-ID | LEN | 2 data bytes |      |
|--------|-----|--------------|------|
|        |     | Command      | Node |
| 0x80   | 2   |              |      |

## 6. LSS-Dienste

Es stehen die LSS-Dienste gemäß CiA/DS305 Version 1.1.1 zur Verfügung. Einschränkungen, bzw. spezielle Erweiterungen sind den nachfolgenden Service-Beschreibungen zu entnehmen. Die zugehörigen COB-ID's sind 2021 (Master => Slave), bzw. 2020 (Slave => Master), wobei der CDvario jeweils Slave ist. Bei einer Knotenadresse von 255 (Auslieferungszustand) startet das Gerät im LSS-Mode.

### 6.1 Umschalteprotokolle (Switch Mode Protocols)

#### 6.1.1. Globale Umschaltung in den Operation/Konfiguration-Mode

M => S

|         |      |   |   |   |   |   |   |
|---------|------|---|---|---|---|---|---|
| 0       | 1    | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 04 | mode | r | r | r | r | r | r |

cs            LSS command specifier  
mode         0: switches to operation mode  
              1: switches to configuration mode  
r             reserved

#### 6.1.2. Selektives Umschaltung in den Operation/Konfiguration-Mode

LSS-Zustand wird getoggelt. Es wird erkannt ob ein passender Teilnehmer am Bus ist.

M => S

|         |           |   |   |   |   |   |   |
|---------|-----------|---|---|---|---|---|---|
| 0       | 1         | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 64 | vendor-id |   |   |   | r | r | r |

M => S

|         |            |   |   |   |   |   |   |
|---------|------------|---|---|---|---|---|---|
| 0       | 1          | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 65 | product-id |   |   |   | r | r | r |

M => S

|         |                 |   |   |   |   |   |   |
|---------|-----------------|---|---|---|---|---|---|
| 0       | 1               | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 66 | revision number |   |   |   | r | r | r |

M => S

|         |               |   |   |   |   |   |   |
|---------|---------------|---|---|---|---|---|---|
| 0       | 1             | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 67 | serial number |   |   |   | r | r | r |

S => M

|         |   |   |   |   |   |   |   |
|---------|---|---|---|---|---|---|---|
| 0       | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 68 | r | r | r | r | r | r | r |

## 6.2 Konfigurationsprotokolle (Configuration Protocols)

### 6.2.1. Konfiguration der Knoten-ID

Dem Gerät im Konfigurationsmode wird eine neue Knotenadresse gegeben. Nach Umschalten in den Operation Mode führt das Gerät einen PowerOnReset aus

M => S

|         |     |   |   |   |   |   |   |
|---------|-----|---|---|---|---|---|---|
| 0       | 1   | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 17 | NID | r | r | r | r | r | r |

cs            LSS command specifier  
 NID          the knot ID which can be adjusted again  
 r            reserved

S => M

|         |            |             |   |   |   |   |   |
|---------|------------|-------------|---|---|---|---|---|
| 0       | 1          | 2           | 3 | 4 | 5 | 6 | 7 |
| cs = 17 | error code | spec. error | r | r | r | r | r |

cs            LSS command specifier  
 error code    0: no error  
               1: not ID incorrectly  
 spec. error  
 r            reserved

### 6.2.2. Konfiguration der Baudrate

Dem Gerät im Konfigurationsmode wird die Baudrate neu eingestellt.

M => S

|         |                |             |   |   |   |   |   |
|---------|----------------|-------------|---|---|---|---|---|
| 0       | 1              | 2           | 3 | 4 | 5 | 6 | 7 |
| cs = 19 | table selector | table index | r | r | r | r | r |

cs            LSS command specifier  
 NID          die neu einzustellende Knoten-ID  
 r            reserved

S => M

|         |            |             |   |   |   |   |   |
|---------|------------|-------------|---|---|---|---|---|
| 0       | 1          | 2           | 3 | 4 | 5 | 6 | 7 |
| cs = 19 | error code | spec. error | r | r | r | r | r |

cs            LSS command specifier  
 error code    0: kein Fehler  
               1: Knoten-ID fehlerhaft  
 spec. error  
 r            reserved

## 6.3 Anfrageprotokolle (Inquiry Protocols)

### 6.3.1. Anfrage der Vendor ID

M => S

|         |   |   |   |   |   |   |   |
|---------|---|---|---|---|---|---|---|
| 0       | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 90 | r | r | r | r | r | r | r |

S => M

|         |           |   |   |   |   |   |   |
|---------|-----------|---|---|---|---|---|---|
| 0       | 1         | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 90 | vendor id |   |   |   | r | r | r |

cs            LSS command specifier  
r            reserved

### 6.3.2. Anfrage des Product Code

M => S

|         |   |   |   |   |   |   |   |
|---------|---|---|---|---|---|---|---|
| 0       | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 91 | r | r | r | r | r | r | r |

S => M

|         |              |   |   |   |   |   |   |
|---------|--------------|---|---|---|---|---|---|
| 0       | 1            | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 91 | product code |   |   |   | r | r | r |

cs            LSS command specifier  
r            reserved

### 6.3.3. Anfrage der Revision Number

M => S

|         |   |   |   |   |   |   |   |
|---------|---|---|---|---|---|---|---|
| 0       | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 92 | r | r | r | r | r | r | r |

S => M

|         |                 |   |   |   |   |   |   |
|---------|-----------------|---|---|---|---|---|---|
| 0       | 1               | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 92 | revision number |   |   |   | r | r | r |

cs            LSS command specifier  
r            reserved

### 6.3.4. Anfrage der Serial Number

M => S

|         |   |   |   |   |   |   |   |
|---------|---|---|---|---|---|---|---|
| 0       | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 93 | r | r | r | r | r | r | r |

S => M

|         |               |   |   |   |   |   |   |
|---------|---------------|---|---|---|---|---|---|
| 0       | 1             | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 93 | serial number |   |   |   | r | r | r |

cs            LSS command specifier  
 r            reserved

### 6.3.5. Anfrage der Knoten ID

M => S

|         |   |   |   |   |   |   |   |
|---------|---|---|---|---|---|---|---|
| 0       | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 94 | r | r | r | r | r | r | r |

S => M

|         |     |   |   |   |   |   |   |
|---------|-----|---|---|---|---|---|---|
| 0       | 1   | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 94 | NID | r | r | r | r | r | r |

cs            LSS command specifier  
 NID         Node-ID  
 r            reserved

## 6.4 Identifikationsprotokolle (Identification Protocols)

### 6.4.1. Identifizieren von nicht konfigurierten Teilnehmern

M => S

|         |   |   |   |   |   |   |   |
|---------|---|---|---|---|---|---|---|
| 0       | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 76 | r | r | r | r | r | r | r |

cs            LSS command specifier  
 r            reserved

S => M

|         |   |   |   |   |   |   |   |
|---------|---|---|---|---|---|---|---|
| 0       | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| cs = 80 | r | r | r | r | r | r | r |

cs            LSS command specifier  
 r            reserved



## 7. Speichern und Laden von Parametern (Store/Restore)

### 7.1 Nichtflüchtiges Speichern von Parametern (Index 0x1010)

Unterstützt werden folgende Varianten:

- ÷ Alle Parameter (Subindex 1)
- ÷ Kommunikationsparameter (Subindex 2)

Das Verhalten unterscheidet sich nicht, da nur die Kommunikationsparameter der PDOs und des 2. SDO-Kanals unverlierbar im EEPROM abgespeichert werden. Nach einem Reset-Node oder dem Einschalten der Spannung startet das Gerät mit diesen eingestellten Daten.

### 7.2 Laden von gespeicherten Parametern (Index 0x1011)

Unterstützt werden folgende Varianten:

- ÷ Alle Parameter (Subindex 1)
- ÷ Kommunikationsparameter (Subindex 2)

Das Verhalten unterscheidet sich nicht, da nur die Kommunikationsparameter der PDOs und des 2. SDO-Kanals neu geladen werden. Wurden Einstellungen an den Kommunikationsparametern der PDOs oder des 2. SDO vorgenommen und sollen diese wieder rückgängig gemacht werden, so werden durch diesen Befehl die unverlierbar im EEPROM abgespeicherten Parameter wieder geladen. Mit dem Befehl „Reset Kommunikationsschnittstelle“ wird die CAN-Bus-Schnittstelle mit dieser abgespeicherten Konfiguration initialisiert.

## 8. Heartbeat

### 8.1 Konfiguration als Heartbeat Producer (Index 0x1017):

Hierfür wird die Zeit in 1 ms Schritten eingegeben in der ein Heartbeat Signal erzeugt werden soll. Wird der Wert 0 eingetragen so erfolgt keine Signalerzeugung.

### 8.2 Konfiguration als Heartbeat Consumer (Index 0x1016):

Es können 2 Überwachungen eingestellt werden (Subindex 1 bzw. Subindex 2):

Es ist die zu überwachende Zeit in 1 ms Schritten einzugeben. Die eingestellte Zeit muß deutlich länger (min. 100 ms länger) als die eingestellte Zeit am Producer sein.

In dem Wert für die zu überwachende Zeit wird im oberen Teil die Knoten-ID des Producers eingetragen.

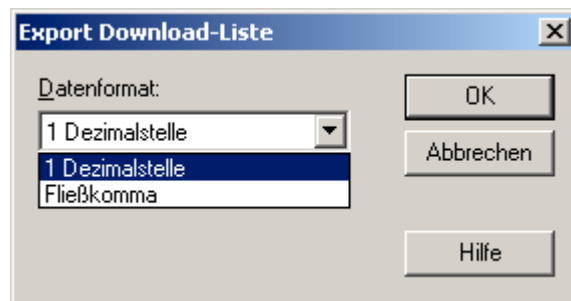
Wird der Wert 0 eingetragen, so erfolgt keine Überwachung.

## 9. Konfigurierung und Parametrisierung über den CAN-Bus

Am einfachsten lässt sich der CDvario mit Hilfe des Engineering-Tool (BlueControl®) über die BlueControl® -Schnittstelle konfigurieren. Eine Konfigurierung direkt über den CAN-Bus ist jedoch auch möglich. Hierzu bietet das BlueControl® den Export einer kompletten Liste der Parameter und Konfigurationsdaten an. Diese Download-Liste enthält den identischen Datensatz, den auch das BlueControl® an den Regler sendet.

Im BlueControl® unter <Datei>,<Export>,<Downloadliste> kann so eine Download-Liste (\*.csv) gespeichert werden.

Es kann ausgewählt werden ob die reglerinternen Float-Daten als Float-Werte oder als Fixpoint1-Werte in der Download-Liste abgelegt werden sollen.



Einen Ausschnitt einer Download-Liste im Float-Format zeigt folgender Ausdruck:

```
Name;Index;Subindex;Wert
Config;8193;0;1
ConF_othr_UseStatErr;8203;0;0
ConF_othr_Unit;8204;0;1
ConF_othr_PowerOnContrOff;8205;0;0
ConF_othr_IntMasterMod1;8207;0;0
ConF_othr_IntMasterMod2;8208;0;0
ConF_othr_IntMasterMod3;8209;0;0
ConF_othr_IntMasterMod4;8210;0;0
...
PArA_sEtP_tSt.2;13424;2;10.0
PArA_sEtP_Gefuehrt.2;9329;2;0
PArA_Cntr_Pb1.2;13462;2;14.095582
PArA_Cntr_Pb2.2;13463;2;14.095582
PArA_Cntr_ti1.2;13464;2;1.9706573
PArA_Cntr_ti2.2;13465;2;1.9706573
PArA_Cntr_td1.2;13466;2;1.9706573
PArA_Cntr_td2.2;13467;2;1.9706573
PArA_Cntr_t1.2;13468;2;0.4000001
PArA_Cntr_t2.2;13469;2;0.4000001
...
PArA_Lim3_L.30;13597;30;-10.0
```

```
PArA_Lim3_H.30;13598;30;10.0  
PArA_Lim3_HYS.30;13599;30;1.0  
PArA_Out1_HcLim.30;13628;30;-32000.0  
PArA_Out2_HcLim.30;13658;30;-32000.0  
Config;8193;0;0
```

In jeder Zeile wird eine Date beschrieben. Der Name, der Index, der Subindex und der Wert sind durch Semikolon getrennt.

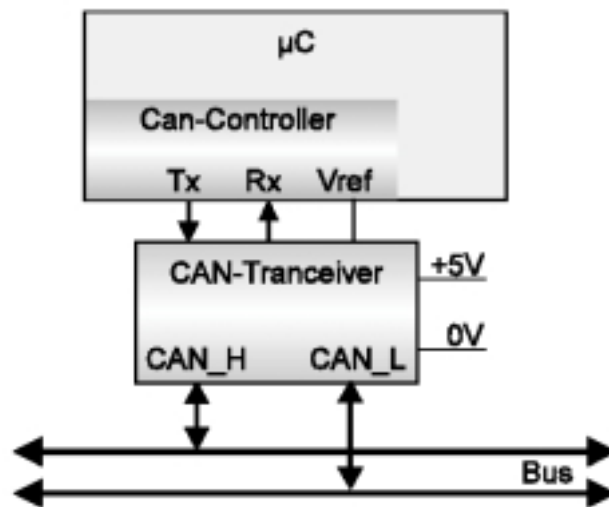
Zu Beginn der Übertragung wird der CDvario in den Konfigurations-Mode geschaltet, danach werden die Konfigurationsdaten und Parameter übertragen. Am Ende wird der Konfigurations-Mode wieder ausgeschaltet und der Regler initialisiert sich. Die Initialisierungsphase dauert ca. 5 Sekunden, dann arbeitet der Regler mit den neue Daten.

## 10. CAN Physical Layer

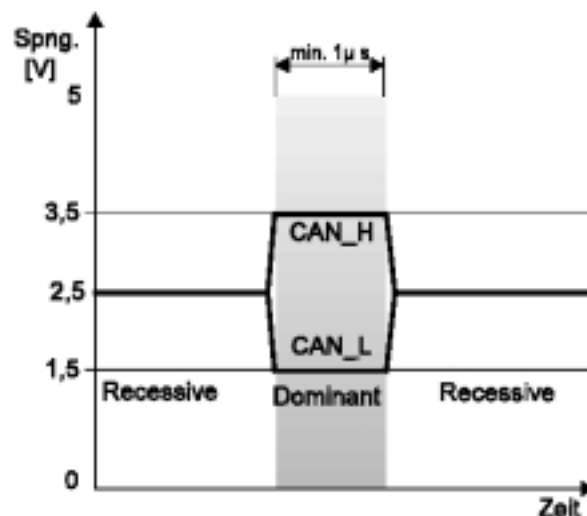
Es gibt eine Reihe von genormten Standards bezüglich des CAN Physical Layers. Der wichtigste für allgemeine Anwendungen ist der "CAN High-Speed Standard ISO 11898-2". Die nachfolgenden Empfehlungen basieren primär auf diesem Standard und gelten unabhängig vom verwendeten CAN-Protokoll (CANopen / DeviceNet).

### 10.1 ISO 11898-2 Knoten

Ein ISO 11898-2 konformer Knoten besteht aus einem  $\mu\text{C}$  mit CAN-Controller (evtl. auch integriert), der über Rx- und Tx-Line mit einem CAN-Transceiver verbunden ist. Der Transceiver wiederum ist mit den differentiellen CAN-H und CAN-L Leitungen am CAN-Bus angeschlossen. Dieser (Transceiver-) Anschluß ist beim CDvario galvanisch getrennt ausgeführt.



Die nominellen CAN-Buspegel werden beim CAN-Bus mit "Recessive" (nominelle Spannung von 2,5 V für CAN-H und CAN-L) und "Dominant" (nominell 3,5V für CAN-H und 1,5V für CAN-L) bezeichnet.



## 10.2 Baudraten und Buslängen

Die maximale, nutzbare Buslänge in einem CAN-Netzwerk wird durch eine Vielzahl von Einflüssen bestimmt, vor allem durch die folgenden physikalischen Effekte:

- ÷ Verzögerungszeiten der angeschlossenen Bus-Knoten (mit/ohne Opto-Koppler) und Verzögerungszeit des Bus-Kabels (propagation delays)
- ÷ unterschiedliche Abtastzeitpunkte innerhalb einer CAN-Bit-Zelle, bedingt durch Oszillatortoleranzen der einzelnen Bus-Knoten
- ÷ Signal-Amplituden Dämpfung, bedingt durch den ohmschen Widerstand des Bus-Kabels und den Eingangs-Widerständen der Bus-Knoten

Die im folgenden aufgeführten praktischen Buslängen können bei Verwendung von ISO 11898-2 konformen Transceivern mit Standard Buskabeln erreicht werden. Bei den hohen Baudraten (1 MBd / 800 kBd) kann es allerdings durch die Anzahl/Geschwindigkeit der evtl. vorhandenen Opto-Koppler (galv. Trennung) zu erheblich kürzeren Buslängen kommen !

## 10.3 Praktische Buslängen

| Baudrate<br>in kBd | Buslänge<br>in m | Nominelle Bit-Time<br>in $\mu$ s |
|--------------------|------------------|----------------------------------|
| 1000               | 30               | 1                                |
| 800                | 50               | 1,25                             |
| 500                | 100              | 2                                |
| 250                | 200              | 4                                |
| 125                | 500              | 8                                |
| 50                 | 1000 *)          | 20                               |
| 20                 | 2500 *)          | 50                               |
| 10                 | 5000 *)          | 100                              |

\*) Bei großen Kabellängen ist der Einsatz von galv. Trennungen und Repeatern zwingend notwendig

Weitere Hinweise zu den Buslängen können auch den Standards CiA „DS-102“ (CANopen entnommen werden.

## 10.4 Kabel-Parameter

ISO 11898-2 definiert einige DC- bzw. AC-Parameter für die in CAN-Bus Netzwerken einsetzbaren Kabel (typischerweise kommen paarweise verdrehte Kabel mit definierten elektrischen Eigenschaften zum Einsatz). Die wichtigen AC Parameter sind 120 Ohm Kabelimpedanz und eine nominelle "propagation delay" von 5 ns/m ! Empfehlungen für die zu verwendenden Buskabel und Abschlußwiderstände können der folgenden Tabelle entnommen werden:

| Bus length     | Bus cable (Z: 120 Ohm, tp: 5ns/m) |   | Terminal resistance | Max. Bit-Rate       |
|----------------|-----------------------------------|---|---------------------|---------------------|
|                | Spec. Resistance                  | Cable diameter  |                     |                     |
| 0 .. 40 m      | 70 mOhm/m                         | 0,25mm <sup>2</sup> , 0,34mm <sup>2</sup><br>AWG 23, AWG 22 | 124 Ohm, 1%         | 1 MBd<br>@ 40m      |
| 40 m .. 300 m  | <60 mOhm/m                        | 0,34mm <sup>2</sup> , 0,6mm <sup>2</sup><br>AWG 22, AWG 20  | 127 Ohm, 1% *)      | > 500 kBd<br>@ 100m |
| 300 m .. 600 m | <40 mOhm/m                        | 0,5mm <sup>2</sup> , 0,6mm <sup>2</sup><br>AWG 20           | 127 Ohm, 1% *)      | > 100 kBd<br>@ 500m |
| 600 m .. 1 km  | <26 mOhm/m                        | 0,75mm <sup>2</sup> , 0,8mm <sup>2</sup><br>AWG 18          | 127 Ohm, 1% *)      | > 50 kBd<br>@ 1 km  |

\*) Bei großen Kabellängen ist ein höherer Wert für den Abschlußwiderstand (150 .. 300 Ohm) hilfreich, zur Reduzierung der Dämpfung.

Weitere Empfehlungen für CAN Netzwerke (speziell auch mit großer Ausdehnung):

- ÷ galv. Trennungen sind notwendig bei großen Längen (z.B. bei 400m Buskabel)
- ÷ separate Ground-Leitung ist sinnvoll
- ÷ der Spannungseinbruch (Potentialdifferenz) zwischen den Ground-Potentialen der Transceiver sollte gering sein (kleiner 2V). Einspeisung des Netzteils evtl. in der Mitte des Kabels
- ÷ der Gesamt-Eingangswiderstand der Bus-Knoten sollte > 500 Ohm sein
- ÷ evtl. notwendige Stichleitungen sollten so kurz wie möglich sein, um Reflektionen zu vermeiden/verringern.

Weitergehende Informationen sind dem CiA (CANopen), den diversen Chip-Herstellern und im Internet zu bekommen.





Subject to alterations without notice.

Bei Änderungen erfolgt keine Mitteilung Postfach 310 229, D - 34058 Kassel

Sous réserve de modifications sans avis préalable

9499 040 69918 (08/2005)