

# miface / RS10

Universal-Interface CANopen / Seriell

## Benutzerhandbuch

## **Inhaltsverzeichnis**

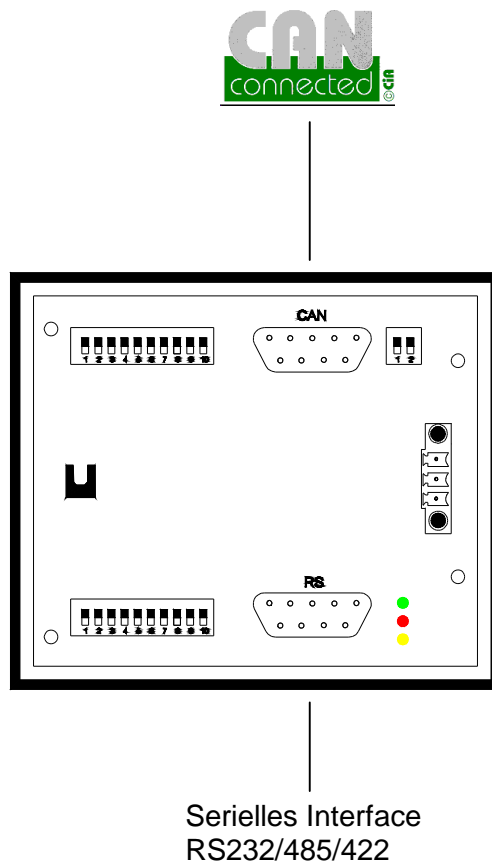
<b>1 ALLGEMEINES</b>	<b>3</b>
<b>2 ÜBERSICHT</b>	<b>4</b>
<b>3 TECHNISCHE INFORMATION</b>	<b>5</b>
<b>3.1 CANopen-Protokoll</b>	<b>6</b>
3.1.1 NMT-Telegramme	6
3.1.2 Nodeguard-Telegramm	7
3.1.3 Heartbeat-Telegramm	7
3.1.4 SDO-Telegramme	8
3.1.5 Receive-PDO-Telegramm	8
3.1.6 Transmit-PDO-Telegramm	8
<b>3.2 RS-Datentelegramm senden (CAN -&gt; serielle Schnittstelle)</b>	<b>9</b>
<b>3.3 RS-Datentelegramm empfangen (serielle Schnittstelle -&gt; CAN)</b>	<b>13</b>
<b>3.4 Anwendungsbeispiele</b>	<b>17</b>
<b>3.5 Hinweise</b>	<b>22</b>
<b>4 ANSCHLUßBELEGUNG / DIP-SCHALTER</b>	<b>23</b>
<b>5 GEHÄUSEABMESSUNGEN</b>	<b>28</b>
<b>6 ANHANG</b>	<b>29</b>
6.1 Lieferumfang	29
6.2 Optional erhältliche Komponenten	29
6.3 Gewährleistung und Haftung	30
6.4 Versionsübersicht	31

## 1 Allgemeines

Mit Hilfe des Universal-Interfaces können Datentelegramme zwischen dem CANopen-Bus und einer seriellen Schnittstelle (RS232/485/422) ausgetauscht werden. Dies ermöglicht Geräten, welche lediglich über eine serielle Schnittstelle verfügen, die Ankopplung an den CANopen-Bus.

Der mechanische Aufbau ist für Hutschienenmontage ausgelegt. Die DIP-Schalter für Adress- und Optionseinstellungen, sowie für den Busabschluss sind frontseitig zugänglich.

## 2 Übersicht



### 3 Technische Information

#### Gesamtspezifikation

<u>Schnittstelle 1:</u>	CANopen gemäß CIA-Norm DS301, V4.0
Bitrate:	10...1000 kBit/s (DIP-Schalter)
Knoten-ID:	1...127 (DIP-Schalter)
PDOs:	1 Receive-PDO, 1 Transmit-PDO
PDO-Linking:	Ja (COB-IDs der verwendeten PDOs sind per SDO verstellbar)
PDO-Mapping:	Fest
Node-Guarding:	Ja
Heartbeat:	Ja
 <u>Schnittstelle 2:</u>	 RS232, RS485 oder RS422
Baudrate:	1,2...115,2 kBaud (DIP-Schalter)
Datenbreite:	8 Bit
Parität:	keine, gerade oder ungerade (DIP-Schalter)
Stopbits:	1
Max. Telegrammlänge:	200 Bytes in Senderichtung 200 Bytes in Empfangsrichtung
Telegrammendeerkennung:	über „Empfangs-Timeout“ (DIP-Schalter)
 Betriebsspannung:	 24 VDC +/-20 %
Stromaufnahme:	150 mA bei 24 VDC Betriebsspannung
Gehäuse:	Hutschienengehäuse mit Metallabdeckung
Gehäusegröße:	siehe Kapitel 5
Befestigung:	Hutschiene auf 35 mm DIN-Tragschiene
Schutzart:	frontseitig IP00
Betriebstemperatur:	0...+50 °C
Lagertemperatur:	-25...+60 °C

### 3.1 CANopen-Protokoll

Die Ansteuerung des Interfaces über die CANopen-Schnittstelle erfolgt wie in CIA DS301, V4.0 (von CAN in Automation e.V., Erlangen) definiert.

Die im Gerät implementierten Protokolle sind im Folgenden kurz beschrieben.

#### 3.1.1 NMT-Telegramme

	COB-ID	B.1	B.2	B.3	B.4	B.5	B.6	B.7	B.8
Start Remote Node	000 <sub>h</sub>	1	0 / Knoten-ID	-	-	-	-	-	-
Stop Remote Node	000 <sub>h</sub>	2	0 / Knoten-ID	-	-	-	-	-	-
Enter Pre-Operational State	000 <sub>h</sub>	128	0 / Knoten-ID	-	-	-	-	-	-
Reset Node	000 <sub>h</sub>	129	0 / Knoten-ID	-	-	-	-	-	-
Reset Communication	000 <sub>h</sub>	130	0 / Knoten-ID	-	-	-	-	-	-

Jeder CANopen-Knoten befindet sich in einem der Betriebszustände „INITIALISATION“, „PRE-OPERATIONAL“, „OPERATIONAL“ oder „STOPPED“.

Nach dem Einschalten wird der Zustand „INITIALISATION“ durchlaufen, wobei die Einträge des Objektverzeichnisses auf ihre Standardwerte gesetzt werden.

Mit Hilfe der Kommandos „Reset Node“ und „Reset Communication“ können entweder alle oder nur die kommunikationsspezifischen Einträge (1000<sub>h</sub> -1FFF<sub>h</sub>) des Objektverzeichnisses jederzeit wieder auf die Standardwerte gesetzt werden.

Anschliessend befindet sich das Gerät im Zustand „PRE-OPERATIONAL“.

Durch Verwendung des Kommandos „Start Remote Node“ kann in den Zustand „OPERATIONAL“ gewechselt werden.

Durch Verwendung des Kommandos „Stop Remote Node“ kann in den Zustand „STOPPED“ gewechselt werden.

Durch Verwendung des Kommandos „Enter Pre-Operational State“ kann zurück in den Zustand „PRE-OPERATIONAL“ gewechselt werden.

### 3.1.2 Nodeguard-Telegramm

	COB-ID	B.1	B.2	B.3	B.4	B.5	B.6	B.7	B.8
Empfang	700 <sub>h</sub> + Knoten-ID RTR=1 DLC=1	-	-	-	-	-	-	-	-
Antwort	700 <sub>h</sub> + Knoten-ID RTR=0 DLC=1	128 x Togglebit (0 oder 1) + aktueller Betriebszustand: 4: STOPPED 5: OPERATIONAL 127: PRE-OPERATIONAL	-	-	-	-	-	-	-

„Nodeguard“ darf nur dann verwendet werden, wenn „Heartbeat“ inaktiv ist („Producer Heartbeat Time“ = Objekt 1017<sub>h</sub> = 0).

Bei aktiviertem „Life-Guarding“ („Guard Time“ = Objekt 100C<sub>h</sub> > 0 und „Life Time Factor“ = Objekt 100D<sub>h</sub> > 0) startet mit jeder Nodeguard-Anforderung ein Timer, welcher das Gerät nach Ablauf der „Life Time“ („Guard Time“ x „Life Time Factor“ ms) automatisch in den Zustand „PRE-OPERATIONAL“ bringt, falls nicht rechtzeitig die nächste Nodeguard-Anforderung eintrifft.

### 3.1.3 Heartbeat-Telegramm

	COB-ID	B.1	B.2	B.3	B.4	B.5	B.6	B.7	B.8
Antwort	700 <sub>h</sub> + Knoten-ID RTR=0 DLC=1	aktueller Betriebszustand: 0: BOOTUP 4: STOPPED 5: OPERATIONAL 127: PRE-OPERATIONAL	-	-	-	-	-	-	-

Nach dem Einschalten wird das „Boot-Up“-Telegramm (Betriebszustand = „BOOTUP“) gesendet.

Durch Einstellen einer „Producer Heartbeat Time“ = Objekt 1017<sub>h</sub> (Einheit: ms) beginnt das Gerät damit, zyklisch das Heartbeat-Telegramm zu senden.

### 3.1.4 SDO-Telegramme

	<b>COB-ID</b>	<b>B.1</b>	<b>B.2</b>	<b>B.3</b>	<b>B.4</b>	<b>B.5</b>	<b>B.6</b>	<b>B.7</b>	<b>B.8</b>
„Initiate Download Request“	600 <sub>h</sub> + Knoten-ID	22 <sub>h</sub> oder 23 <sub>h</sub> + *	Index - LOW	Index - HIGH	Subindex	D0 (LSB)	D1	D2	D3 (MSB)
„Initiate Download Response“	580 <sub>h</sub> + Knoten-ID	60 <sub>h</sub>	Index - LOW	Index - HIGH	Subindex	0	0	0	0
„Initiate Upload Request“	600 <sub>h</sub> + Knoten-ID	40 <sub>h</sub>	Index - LOW	Index - HIGH	Subindex	0	0	0	0
„Initiate Upload Response“	580 <sub>h</sub> + Knoten-ID	43 <sub>h</sub> + *	Index - LOW	Index - HIGH	Subindex	D0 (LSB)	D1	D2	D3 (MSB)
„Abort Domain Transfer“ (Empfang-SDO)	600 <sub>h</sub> + Knoten-ID	80 <sub>h</sub>	Index - LOW	Index - HIGH	Subindex	additional code	0	error code	error class
„Abort Domain Transfer“ (Sende-SDO)	580 <sub>h</sub> + Knoten-ID	80 <sub>h</sub>	Index - LOW	Index - HIGH	Subindex	additional code	0	error code	error class

\* = 4 x „Anzahl unbenutzter Datenbytes“

Mit Hilfe der SDO-Telegramme ist der Zugriff auf das Objektverzeichnis eines jeden CANopen-Knotens möglich.

### 3.1.5 Receive-PDO-Telegramm

	<b>COB-ID</b>	<b>B.1</b>	<b>B.2</b>	<b>B.3</b>	<b>B.4</b>	<b>B.5</b>	<b>B.6</b>	<b>B.7</b>	<b>B.8</b>
RPDO1	200 <sub>h</sub> + Knoten-ID	RS-Sende-daten_1	RS-Sende-daten_2	RS-Sende-daten_3	RS-Sende-daten_4	RS-Sende-daten_5	RS-Sende-daten_6	RS-Sende-daten_7	RS-Sende-daten_8

RPDO1 ist auf das Objekt 2000<sub>h</sub>, Subindex 1 bis 8 gemappt.

RPDO1 muß (mehrfach) zum Interface gesendet werden, um ein RS-Sendetelegramm zu erzeugen.

### 3.1.6 Transmit-PDO-Telegramm

	<b>COB-ID</b>	<b>B.1</b>	<b>B.2</b>	<b>B.3</b>	<b>B.4</b>	<b>B.5</b>	<b>B.6</b>	<b>B.7</b>	<b>B.8</b>
TPDO1	180 <sub>h</sub> + Knoten-ID	RS-Empfangs-daten_1	RS-Empf.-daten_2	RS-Empf.-daten_3	RS-Empf.-daten_4	RS-Empf.-daten_5	RS-Empf.-daten_6	RS-Empf.-daten_7	RS-Empf.-daten_8

TPDO1 ist auf das Objekt 2001<sub>h</sub>, Subindex 1 bis 8 gemappt.

TPDO1 wird (mehrfach) vom Interface gesendet, nachdem ein RS-Telegramm empfangen wurde.

### 3.2 RS-Datentelegramm senden (CAN -> serielle Schnittstelle)

#### Receive-PDO 1 (Objekt 2000<sub>h</sub>, Subindex 1 bis 8)

Byte	Funktion
1	Funktionsbyte: <ul style="list-style-type: none"> <li>• Bit 7:           Endeбит</li> <li>• Bit 6...5:       reserviert (=0)</li> <li>• Bit 4:           Togglebit</li> <li>• Bit 3:           reserviert (=0)</li> <li>• Bit 2...0:       Teil-Telegramm-Länge</li> </ul>
2	Teil-Telegramm-Byte 1
3	Teil-Telegramm-Byte 2
4	Teil-Telegramm-Byte 3
5	Teil-Telegramm-Byte 4
6	Teil-Telegramm-Byte 5
7	Teil-Telegramm-Byte 6
8	Teil-Telegramm-Byte 7

#### **Togglebit:**

Bei jeder Veränderung wird das aktuelle Teil-Telegramm an das Ende des Sendepuffers angehängt.

#### **Endeбит:**

- = 0:   Teil-Telegramme werden gesammelt
- = 1:   Die angesammelten Teil-Telegramme werden gesendet (inklusive dem gerade übergebenen Teil-Telegramm, falls sich auch das Togglebit geändert hat). Nach Ende des Sendevorgangs wird der Sendepuffer gelöscht (um wieder neue Teil-Telegramme aufnehmen zu können) und das Endeбит wird auf 0 gesetzt (um das Ende des Sendevorgangs per SDO erfragen zu können).

#### **Teil-Telegramm-Länge:**

Länge des übergebenen RS-Teil-Telegramms

#### **Teil-Telegramm-Bytes:**

Werden bei Veränderung des Togglebits an das Ende des sich bereits im RS-Sendepuffer befindlichen Telegramms angehängt.

## Vorgehensweise auf CANopen-Seite zur Aussendung eines RS-Telegramms:

### 1. Aufteilen des zu sendenden RS-Telegramms in Teil-Telegramme zu je max. 7 Bytes

### 2. Teil-Telegramme zum Interface übertragen

- Inhalt des zu sendenden PDOs vorbereiten:
  - Telegrammbytes des zu übergebenden Teil-Telegramms in die PDO-Bytes 2 bis max. 8 eintragen
  - PDO-Byte 1:
    - "Teil-Telegramm-Länge" (1 bis 7) eintragen
    - "Togglebit" verändern
    - "Endebit" = 1 setzen, falls kein weiteres Teil-Telegramm zu übertragen ist
- PDO absenden
- Übertragung des PDOs abwarten
- weitere 5 ms warten (um dem Interface Zeit zu geben, das PDO auszuwerten)
- wiederholen der letzten 4 Schritte, bis alle Teil-Telegramme übergeben wurden

### 3. Übertragung des RS-Telegramms abwarten:

- Die Übertragungszeit ist abhängig von der Telegrammlänge und der verwendeten RS-Baudrate:
    - =  $1,3^* \times \text{Telegrammlänge} \times (10 \times 1/\text{Baudrate})$  falls kein Paritätsbit verwendet wird
    - =  $1,3^* \times \text{Telegrammlänge} \times (11 \times 1/\text{Baudrate})$  falls ein Paritätsbit verwendet wird
- \*: Der Sicherheitsfaktor 1,3 muß berücksichtigt werden, da die einzelnen Telegrammbytes nicht völlig ohne Lücken gesendet werden können (wirkt sich insbesondere bei den hohen Baudraten - 38400...115200 Baud - aus).
- Das Ende der Telegrammaussendung kann auch durch Abfrage des „Endebits“ (per SDO) erkannt werden. Das Endebit geht auf 0 sobald der Sendepuffer wieder frei ist und das nächste Telegramm übergeben werden kann.

## Beispiel zur Aussendung eines RS-Telegramms

(RS-Telegramm "ABCDEFG12345" senden):

### 1. Aufteilen des zu sendenden RS-Telegramms in Teil-Telegramme zu je max. 7 Bytes

- 1. Teil-Telegramm (7 Bytes) = 41<sub>h</sub> 42<sub>h</sub> 43<sub>h</sub> 44<sub>h</sub> 45<sub>h</sub> 46<sub>h</sub> 47<sub>h</sub>
- 2. Teil-Telegramm (5 Bytes) = 31<sub>h</sub> 32<sub>h</sub> 33<sub>h</sub> 34<sub>h</sub> 35<sub>h</sub>

### 2. Teil-Telegramme zum Interface übertragen

- PDO absenden: 17<sub>h</sub> 41<sub>h</sub> 42<sub>h</sub> 43<sub>h</sub> 44<sub>h</sub> 45<sub>h</sub> 46<sub>h</sub> 47<sub>h</sub>  
(Voraussetzung: letztes Togglebit = 0)
- Übertragung des PDOs abwarten
- weitere 5 ms warten (um dem Interface Zeit zu geben, das PDO auszuwerten)
- PDO absenden: 85<sub>h</sub> 31<sub>h</sub> 32<sub>h</sub> 33<sub>h</sub> 34<sub>h</sub> 35<sub>h</sub> XX<sub>h</sub> XX<sub>h</sub>
- Übertragung des PDOs abwarten
- weitere 5 ms warten (um dem Interface Zeit zu geben, das PDO auszuwerten)

### 3. Übertragung des RS-Telegramms abwarten:

- Beispiel: 19200 Baud, 8 Datenbits, gerade Parität, 1 Stopbit  
=> Übertragungszeit =  $1,3 \times 12 \times (11 \times 1/19200)$  s = 9 ms  
=> mindestens weitere 9 ms warten (das Funktionsbyte wird in diesem Zeitraum nicht ausgewertet!)
- Abfrage des "Endebits" per SDO:
  - CAN-Telegramm senden (COB-ID = 600<sub>h</sub> + Knoten-ID):  
40<sub>h</sub> 00<sub>h</sub> 20<sub>h</sub> 01<sub>h</sub> 00<sub>h</sub> 00<sub>h</sub> 00<sub>h</sub> 00<sub>h</sub>
  - CAN-Telegramm empfangen (COB-ID = 580<sub>h</sub> + Knoten-ID):  
4F<sub>h</sub> 00<sub>h</sub> 20<sub>h</sub> 01<sub>h</sub> „Funktionsbyte“ 00<sub>h</sub> 00<sub>h</sub> 00<sub>h</sub>

Sobald Bit 7 (Endebit) des „Funktionsbyte“ mit 0 gelesen wird, darf das nächste RS-Telegramm zum Interface übertragen werden.

**Hinweise:**

- Der RS-Sendebetrieb funktioniert nur im CANopen-Zustand "OPERATIONAL"!
- Nach dem Übergang in den Zustand "OPERATIONAL":
  - Der Sendepuffer wird gelöscht (evtl. bisher angesammelte Teil-Telegramme werden gelöscht).
  - Ein evtl. noch offener RS-Sendevorgang wird vorher noch abgeschlossen.
  - Anschliessend wird Objekt 2000<sub>h</sub>, Subindex 1 (Funktionsbyte) gelöscht. => Das mit dem ersten Teil-Telegramm zu übergebende Togglebit muss 1 sein!
- Während eines laufenden RS-Sendevorgangs wird das Funktionsbyte nicht ausgewertet. Es sollte also erst nach Abschluß des RS-Sendevorgangs wieder verändert werden.
- Ist das übergebene RS-Telegramm länger als der verfügbare RS-Sendepuffer (200 Bytes), so wird ein entsprechend gekürztes RS-Telegramm ausgesendet!

### 3.3 RS-Datentelegramm empfangen (serielle Schnittstelle -> CAN)

#### Transmit-PDO 1 (Objekt 2001<sub>h</sub>, Subindex 1 bis 8)

Byte	Funktion
1	Funktionsbyte: <ul style="list-style-type: none"> <li>• Bit 7:       Endebeit</li> <li>• Bit 6...5:   reserviert (=0)</li> <li>• Bit 4:       Togglebit</li> <li>• Bit 3:       reserviert (=0)</li> <li>• Bit 2...0:   Teil-Telegramm-Länge</li> </ul>
2	Teil-Telegramm-Byte 1
3	Teil-Telegramm-Byte 2
4	Teil-Telegramm-Byte 3
5	Teil-Telegramm-Byte 4
6	Teil-Telegramm-Byte 5
7	Teil-Telegramm-Byte 6
8	Teil-Telegramm-Byte 7

#### **Togglebit:**

Ändert sich mit jedem übergebenen Teil-Telegramm

#### **Endebeit:**

= 0: Weitere Teil-Telegramme folgen

= 1: Letztes Teil-Telegramm

#### **Teil-Telegramm-Länge:**

Länge des aktuell übergebenen RS-Teil-Telegramms

#### **Teil-Telegramm-Bytes:**

Aktuell übergebenes RS-Teil-Telegramm

## **Vorgehensweise des Interfaces nach Empfang eines RS-Telegramms:**

### 1. Aufteilen des RS-Telegramms in Teil-Telegramme zu je max. 7 Bytes

### 2. Teil-Telegramme übertragen

- Inhalt des zu sendenden PDOs vorbereiten:
  - Telegrammbytes des zu übergebenden Teil-Telegramms in die PDO-Bytes 2 bis max. 8 eintragen
  - PDO-Byte 1:
    - "Teil-Telegramm-Länge" eintragen  
(= 7, außer letztes Teil-Telegramm = 1 ... 7)
    - "Togglebit" verändern
    - "Endebit" = 1 setzen, falls kein weiteres Teil-Telegramm zu übertragen ist
- eingestellte "Inhibit-Time" (Objekt 1800<sub>h</sub>, Subindex 3) noch zu Ende warten, falls vorher ein PDO gesendet wurde
- PDO absenden
- wiederholen der letzten 3 Schritte, bis alle Teil-Telegramme übergeben wurden

### 3. RS-Empfangspuffer wieder freigeben

## **Beispiel zum Empfang eines RS-Telegramms ("ABCDEFG12345"):**

### 1. Aufteilen des RS-Telegramms in Teil-Telegramme zu je max. 7 Bytes

- 1. Teil-Telegramm (7 Bytes) = 41<sub>h</sub> 42<sub>h</sub> 43<sub>h</sub> 44<sub>h</sub> 45<sub>h</sub> 46<sub>h</sub> 47<sub>h</sub>
- 2. Teil-Telegramm (5 Bytes) = 31<sub>h</sub> 32<sub>h</sub> 33<sub>h</sub> 34<sub>h</sub> 35<sub>h</sub>

### 2. Teil-Telegramme übertragen

- eingestellte "Inhibit-Time" noch zu Ende warten, falls vorher ein PDO gesendet wurde
- PDO absenden: 17<sub>h</sub> 41<sub>h</sub> 42<sub>h</sub> 43<sub>h</sub> 44<sub>h</sub> 45<sub>h</sub> 46<sub>h</sub> 47<sub>h</sub>  
(Voraussetzung: letztes Togglebit = 0)
- eingestellte "Inhibit-Time" abwarten
- PDO absenden: 85<sub>h</sub> 31<sub>h</sub> 32<sub>h</sub> 33<sub>h</sub> 34<sub>h</sub> 35<sub>h</sub> XX<sub>h</sub> XX<sub>h</sub>

### 3. RS-Empfangspuffer wieder freigeben

#### Hinweise:

- Der RS-Empfangsbetrieb funktioniert nur im CANopen-Zustand "OPERATIONAL"!
- Nach dem Übergang in den Zustand "OPERATIONAL":
  - Der Empfangspuffer wird gelöscht.
  - Evtl. sich bereits im Empfangspuffer befindliche Daten werden verworfen.
  - Objekt 2001<sub>h</sub> wird komplett gelöscht (alle Bytes = 0)  
=> Das mit dem ersten PDO übergebene Teil-Telegramm hat ein auf 1 gesetztes Togglebit.
- Das Ende der RS-Empfangstelegramme wird durch den "Empfangs-Timeout" erkannt (über DIP-Schalter einstellbar). Während des eingestellten Zeitraums dürfen keine Zeichen empfangen werden, um das Telegrammende zu erkennen.
- Die RS-Empfangstelegramme dürfen nicht länger als 200 Bytes sein.
- Während das RS-Empfangstelegramm über die CANopen-Schnittstelle versendet wird, darf kein weiteres RS-Telegramm ankommen (sonst wird es verworfen).

### 3.4 Anwendungsbeispiele

#### 1. Ausgabe des Online-Textes „Hallo Welt!“ auf der microSYST-Anzeige „MIGRA“ (ohne Antworttelegramm):

MIGRA-Adresse = 01<sub>h</sub>  
CANopen-Knoten-ID = 01<sub>h</sub>

Zu sendendes RS-Telegramm (siehe MIGRA-Handbuch):

02<sub>h</sub> 81<sub>h</sub> 80<sub>h</sub> 80<sub>h</sub> 48<sub>h</sub> 61<sub>h</sub> 6C<sub>h</sub> 6C<sub>h</sub> 6F<sub>h</sub> 20<sub>h</sub> 57<sub>h</sub> 65<sub>h</sub> 6C<sub>h</sub> 74<sub>h</sub> 21<sub>h</sub> 03<sub>h</sub>

- Knoten starten (falls noch nicht „OPERATIONAL“):  
NMT-Telegramm (COB-ID = 000<sub>h</sub>) senden: 01<sub>h</sub> 01<sub>h</sub>
- RPDO1-Telegramm (COB-ID = 201<sub>h</sub>) senden:  
17<sub>h</sub> 02<sub>h</sub> 81<sub>h</sub> 80<sub>h</sub> 80<sub>h</sub> 48<sub>h</sub> 61<sub>h</sub> 6C<sub>h</sub> (letztes Togglebit war 0)
- mind. 5 ms warten
- RPDO1-Telegramm (COB-ID = 201<sub>h</sub>) senden:  
07<sub>h</sub> 6C<sub>h</sub> 6F<sub>h</sub> 20<sub>h</sub> 57<sub>h</sub> 65<sub>h</sub> 6C<sub>h</sub> 74<sub>h</sub>
- mind. 5 ms warten
- RPDO1-Telegramm (COB-ID = 201<sub>h</sub>) senden:  
92<sub>h</sub> 21<sub>h</sub> 03<sub>h</sub> XX<sub>h</sub> XX<sub>h</sub> XX<sub>h</sub> XX<sub>h</sub> XX<sub>h</sub>
- mind. 5 ms warten
- Übertragung des RS-Telegramms abwarten: Bei einer RS-Baudrate von 19200 Baud, 8 Datenbits, gerader Parität und 1 Stopbit dauert der RS-Sendevorgang ca.  $1,3 \times 16 \times (11 \times 1/19200) \text{ s} = 12 \text{ ms}$ .
- Telegrammauswertung durch die MIGRA abwarten (siehe MIGRA-Handbuch)

## 2. Ausgabe des Online-Textes „Hallo Welt!“ auf der microSYST-Anzeige „MIGRA“ (mit Antworttelegramm):

MIGRA-Adresse = 01<sub>h</sub>

CANopen-Knoten-ID = 01<sub>h</sub>

Zu sendendes RS-Telegramm (siehe MIGRA-Handbuch):

02<sub>h</sub> 81<sub>h</sub> 80<sub>h</sub> 81<sub>h</sub> 48<sub>h</sub> 61<sub>h</sub> 6C<sub>h</sub> 6C<sub>h</sub> 6F<sub>h</sub> 20<sub>h</sub> 57<sub>h</sub> 65<sub>h</sub> 6C<sub>h</sub> 74<sub>h</sub> 21<sub>h</sub> 03<sub>h</sub>

- Knoten starten (falls noch nicht „OPERATIONAL“):  
NMT-Telegramm (COB-ID = 000<sub>h</sub>) senden: 01<sub>h</sub> 01<sub>h</sub>
- RPDO1-Telegramm (COB-ID = 201<sub>h</sub>) senden:  
17<sub>h</sub> 02<sub>h</sub> 81<sub>h</sub> 80<sub>h</sub> 81<sub>h</sub> 48<sub>h</sub> 61<sub>h</sub> 6C<sub>h</sub> (letztes Togglebit war 0)
- mind. 5 ms warten
- RPDO1-Telegramm (COB-ID = 201<sub>h</sub>) senden:  
07<sub>h</sub> 6C<sub>h</sub> 6F<sub>h</sub> 20<sub>h</sub> 57<sub>h</sub> 65<sub>h</sub> 6C<sub>h</sub> 74<sub>h</sub>
- mind. 5 ms warten
- RPDO1-Telegramm (COB-ID = 201<sub>h</sub>) senden:  
92<sub>h</sub> 21<sub>h</sub> 03<sub>h</sub> XX<sub>h</sub> XX<sub>h</sub> XX<sub>h</sub> XX<sub>h</sub> XX<sub>h</sub>

MIGRA-Antworttelegramm (siehe MIGRA-Handbuch):

02<sub>h</sub> 80<sub>h</sub> 81<sub>h</sub> 80<sub>h</sub> 30<sub>h</sub> 03<sub>h</sub>

- TPDO1-Telegramm (COB-ID = 181<sub>h</sub>) empfangen:  
96<sub>h</sub> 02<sub>h</sub> 80<sub>h</sub> 81<sub>h</sub> 80<sub>h</sub> 30<sub>h</sub> 03<sub>h</sub> 00<sub>h</sub> (letztes Togglebit war 0)

### 3. Ausgabe von „123456789012345“ auf der microSYST-Anzeige „MIGAN“:

MIGAN-Adresse = 01

MIGAN-Schnittstellenparameter = 9600 Baud, 7 Datenbits, keine Parität

CANopen-Knoten-ID = 01<sub>h</sub>

MIFACE-Schnittstellenparameter = 9600 Baud, 8 Datenbits, keine Parität

Da die MIGAN lediglich 7 Datenbits benötigt, muß das letzte übertragene Bit (Bit 7) eines jeden Bytes auf 1 gesetzt sein, um ein vorzeitiges „Stopbit“ zu simulieren. Die Übertragung entspricht dann dem Format: 7 Datenbits, keine Parität, 2 Stopbits.

MIGAN-Telegramm (siehe MIGAN-Handbuch):

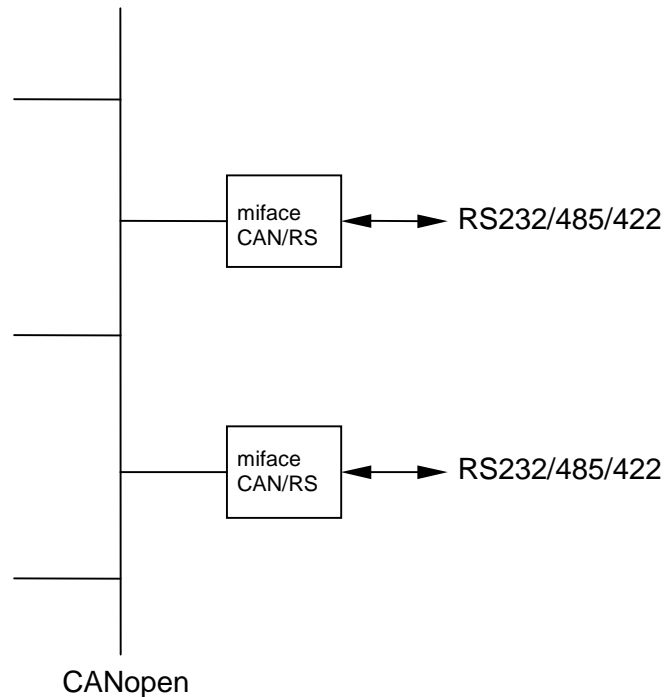
3C<sub>h</sub> 30<sub>h</sub> 31<sub>h</sub> 40<sub>h</sub> 40<sub>h</sub> 40<sub>h</sub> 31<sub>h</sub> 32<sub>h</sub> 33<sub>h</sub> 34<sub>h</sub> 35<sub>h</sub> 36<sub>h</sub> 37<sub>h</sub> 38<sub>h</sub> 39<sub>h</sub> 30<sub>h</sub> 31<sub>h</sub> 32<sub>h</sub> 33<sub>h</sub> →  
34<sub>h</sub> 35<sub>h</sub> 3E<sub>h</sub>

Zu sendendes RS-Telegramm (Bit 7 = 1):

BC<sub>h</sub> B0<sub>h</sub> B1<sub>h</sub> C0<sub>h</sub> C0<sub>h</sub> C0<sub>h</sub> B1<sub>h</sub> B2<sub>h</sub> B3<sub>h</sub> B4<sub>h</sub> B5<sub>h</sub> B6<sub>h</sub> B7<sub>h</sub> B8<sub>h</sub> B9<sub>h</sub> B0<sub>h</sub> B1<sub>h</sub> B2<sub>h</sub> →  
B3<sub>h</sub> B4<sub>h</sub> B5<sub>h</sub> BE<sub>h</sub>

- Knoten starten (falls noch nicht „OPERATIONAL“):  
NMT-Telegramm (COB-ID = 000<sub>h</sub>) senden: 01<sub>h</sub> 01<sub>h</sub>
- RPDO1-Telegramm (COB-ID = 201<sub>h</sub>) senden: 17<sub>h</sub> BC<sub>h</sub> B0<sub>h</sub> B1<sub>h</sub> C0<sub>h</sub> C0<sub>h</sub> C0<sub>h</sub> B1<sub>h</sub>  
(letztes Togglebit war 0)
- mind. 5 ms warten
- RPDO1-Telegramm (COB-ID = 201<sub>h</sub>) senden: 07<sub>h</sub> B2<sub>h</sub> B3<sub>h</sub> B4<sub>h</sub> B5<sub>h</sub> B6<sub>h</sub> B7<sub>h</sub> B8<sub>h</sub>
- mind. 5 ms warten
- RPDO1-Telegramm (COB-ID = 201<sub>h</sub>) senden: 17<sub>h</sub> B9<sub>h</sub> B0<sub>h</sub> B1<sub>h</sub> B2<sub>h</sub> B3<sub>h</sub> B4<sub>h</sub> B5<sub>h</sub>
- mind. 5 ms warten
- RPDO1-Telegramm (COB-ID = 201<sub>h</sub>) senden: 81<sub>h</sub> BE<sub>h</sub> XX<sub>h</sub> XX<sub>h</sub> XX<sub>h</sub> XX<sub>h</sub> XX<sub>h</sub> XX<sub>h</sub>
- mind. 5 ms warten
- Übertragung des RS-Telegramms abwarten: Bei einer RS-Baudrate von 9600 Baud, 8 Datenbits, keiner Parität und 1 Stopbit dauert der RS-Sendevorgang ca.  $1,3 \times 22 \times (10 \times 1/9600) \text{ s} = 30 \text{ ms}$ .

## 4. Verbindung zweier serieller Schnittstellen über den CANopen-Bus



Soll die CANopen-Kommunikation quasi „selbstständig“ ablaufen, so müssen die PDOs der beiden Interface miteinander verbunden werden (PDO-Linking).

### Beispiel:

#### Interface 1 :

- Knoten-ID = 1
- RPDO1 = 201<sub>h</sub> (Standardwert = 201<sub>h</sub>)
- TPDO1 = 181<sub>h</sub> (Standardwert = 181<sub>h</sub>)

#### Interface 2 :

- Knoten-ID = 2
- RPDO1 = 181<sub>h</sub> (Standardwert = 202<sub>h</sub>)
- TPDO1 = 201<sub>h</sub> (Standardwert = 182<sub>h</sub>)

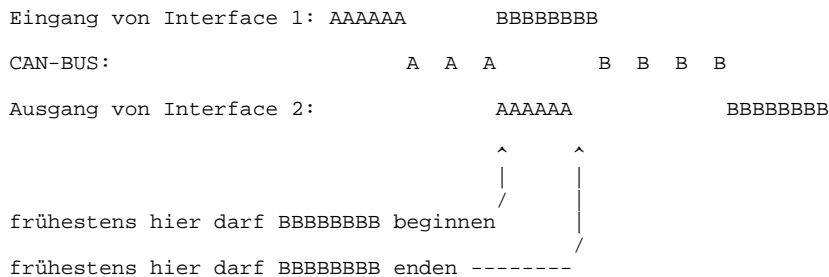
Nach dem Einschalten müssen also hier die COB-IDs der beiden PDOs von Interface 2 verstellt werden:

- SDO-Telegramm (COB-ID = 602<sub>h</sub>) senden: 22<sub>h</sub> 00<sub>h</sub> 14<sub>h</sub> 01<sub>h</sub> 81<sub>h</sub> 01<sub>h</sub> 00<sub>h</sub> 00<sub>h</sub>
- SDO-Antwort (COB-ID = 582<sub>h</sub>) abwarten: 60<sub>h</sub> 00<sub>h</sub> 14<sub>h</sub> 01<sub>h</sub> 00<sub>h</sub> 00<sub>h</sub> 00<sub>h</sub> 00<sub>h</sub>
- SDO-Telegramm (COB-ID = 602<sub>h</sub>) senden: 22<sub>h</sub> 00<sub>h</sub> 18<sub>h</sub> 01<sub>h</sub> 01<sub>h</sub> 02<sub>h</sub> 00<sub>h</sub> 00<sub>h</sub>
- SDO-Antwort (COB-ID = 582<sub>h</sub>) abwarten: 60<sub>h</sub> 00<sub>h</sub> 18<sub>h</sub> 01<sub>h</sub> 00<sub>h</sub> 00<sub>h</sub> 00<sub>h</sub> 00<sub>h</sub>
- Knoten 1 starten: NMT-Telegramm (COB-ID = 000<sub>h</sub>) senden: 01<sub>h</sub> 01<sub>h</sub>
- Knoten 2 starten: NMT-Telegramm (COB-ID = 000<sub>h</sub>) senden: 01<sub>h</sub> 02<sub>h</sub>

Bitte beachten Sie:

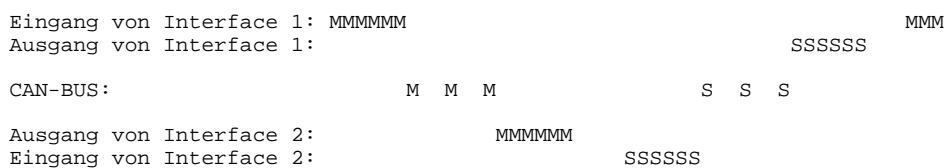
- Das jeweils nächste RS-Telegramm darf frühestens dann beginnen, wenn die CAN-Übertragung des vorherigen Telegramms beendet ist (erst dann ist der RS-Empfangspuffer wieder frei).
- Das jeweils nächste RS-Telegramm darf frühestens dann enden, wenn das vorherige Telegramm bereits vollständig vom zweiten Interface ausgegeben wurde (erst dann ist der RS-Sendepuffer im zweiten Interface wieder frei).

Veranschaulichung:



- Bei einem „Master-Slave-Verhältnis“ zwischen den beiden RS-Anschaltungen ist a) und b) immer erfüllt. „Master-Slave-Verhältnis“ bedeutet, daß auf ein Anforderungstelegramm des Masters hin der Slave ein Antworttelegramm erzeugt. Der Master wartet dieses Antworttelegramm ab, bevor er selber wieder etwas schickt.

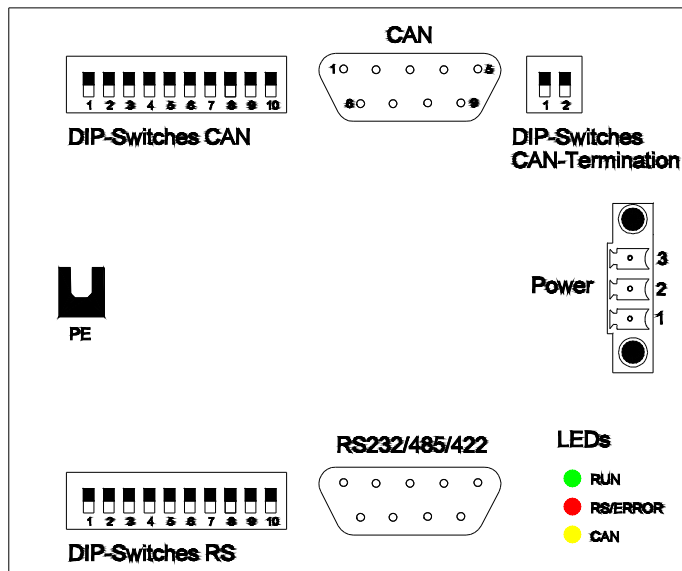
Veranschaulichung:



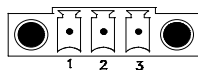
## 3.5 Hinweise

- Es können lediglich in sich abgeschlossene Telegramme übertragen werden (Abstand der Zeichen < „Empfangs-Timeout“).
- Die Telegramme müssen untereinander einen ausreichend großen Abstand besitzen:  
> („Empfangs-Timeout“ + Übertragungsdauer am CAN-Bus)
- Um eine möglichst schnelle Telegrammfolge zu erlauben, sollte der „Empfangs-Timeout“ (DIP-Schalter) und die „PDO-Inhibit-Time“ (Objekt 1800<sub>h</sub>, Subindex 3) so niedrig wie möglich eingestellt werden.
- Die „PDO-Inhibit-Time“ beträgt standardmäßig 10 ms.
- Verstellen der „PDO-Inhibit-Time“ (Knoten-ID = 1) auf (256 x HB + LB) x 0,1 ms:
  - SDO-Telegramm (COB-ID = 601<sub>h</sub>) senden:  
22<sub>h</sub> 00<sub>h</sub> 18<sub>h</sub> 03<sub>h</sub> LB HB 00<sub>h</sub> 00<sub>h</sub>
  - SDO-Antwort (COB-ID = 581<sub>h</sub>) abwarten:  
60<sub>h</sub> 00<sub>h</sub> 18<sub>h</sub> 03<sub>h</sub> 00<sub>h</sub> 00<sub>h</sub> 00<sub>h</sub> 00<sub>h</sub>
- Das Interface muß sich bereits beim Einschalten an einem aktiven CAN-Bus befinden.  
Durch Aussenden der Boot-Up-Message würde sonst ein CAN-Fehler entstehen.

## 4 Anschlußbelegung / DIP-Schalter



### 3pol. Klemmsteckverbinder

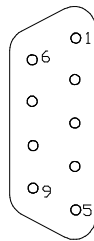


Pin	Power
1	+24 VDC
2	GND
3	PE

## 9pol. Sub-D Buchsenleiste/Stiftleiste (RS-Schnittstelle)

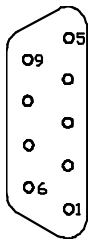
Es ist je nach RS-Schnittstelle (Bestelloption) folgende Belegung gültig:

### RS232 (Stiftleiste):



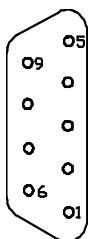
Pin	RS232
1	n.c.
2	RxD
3	TxD
4	n.c.
5	GND
6	n.c.
7	n.c.
8	n.c.
9	n.c.

### RS485 (Buchsenleiste):



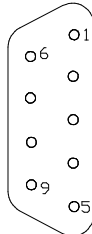
Pin	RS485
1	n.c.
2	n.c.
3	B-Ader
4	n.c.
5	GNDext
6	+5 Vext
7	n.c.
8	A-Ader
9	n.c.

### RS422 (Buchsenleiste):



Pin	RS422
1	n.c.
2	n.c.
3	TxD +
4	RTS
5	GNDext
6	+5 Vext
7	RxD -
8	TxD -
9	RxD +

### CAN (Stiftleiste):



Pin	CAN
1	n.c.
2	CAN_L
3	CAN_GND
4	n.c.
5	CAN_Shield
6	GND
7	CAN_H
8	n.c.
9	n.c.

### LEDs:

LED	Zustand
RUN (grün)	Normalbetrieb: Blinkend
RS/ERROR (rot)	RS-Komm.: Flackernd Fehler: Ein
CAN (gelb)	CAN-Telegramm empfangen oder gesendet => 100 ms EIN

## DIP-Schalter

Im Folgenden sind die DIP-Schalterstellungen mit 0 = OFF und 1 = ON angegeben.

Die DIP-Schalter werden nur einmalig während des Neuanlaufs (nach Reset) eingelesen.

## CAN-Knoten-ID:

DIP-Switches CAN							CAN-Knoten-ID
DIP							
7	6	5	4	3	2	1	
0	0	0	0	0	0	1	1 <sub>d</sub>
0	0	0	0	0	1	0	2 <sub>d</sub>
0	0	0	0	0	1	1	3 <sub>d</sub>
:							:
1	1	1	1	1	1	1	127 <sub>d</sub>

Hinweis: Es sind nur die Adressen 1...127 zulässig!

## CAN-Bitrate:

DIP-Switches CAN			CAN-Bitrate
DIP			
10	9	8	
0	0	0	1000 kBit/s
0	0	1	800 kBit/s
0	1	0	500 kBit/s
0	1	1	250 kBit/s
1	0	0	125 kBit/s
1	0	1	50 kBit/s
1	1	0	20 kBit/s
1	1	1	10 kBit/s

## CAN-Busabschluss:

DIP-Switches CAN-Termination		CAN-Busabschluss
DIP		
1	2	
0	0	nicht gesetzt
1	1	gesetzt

### RS-Baudrate / Standard-Timeout:

DIP-Switches RS			RS-Baudrate (Baud)	Standard-Timeout
DIP				
3	2	1		
0	0	0	1200	28 ms
0	0	1	2400	14 ms
0	1	0	4800	7 ms
0	1	1	9600	4 ms
1	0	0	19200	2 ms
1	0	1	38400	1 ms
1	1	0	57600	1 ms
1	1	1	115200	1 ms

### Parität / Empfangs-Timeout:

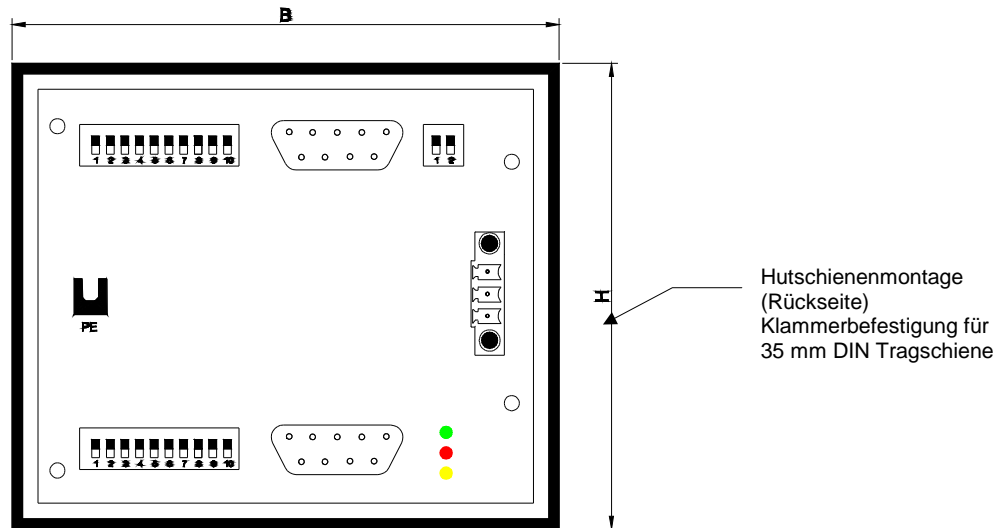
DIP-Switches RS			Parität	Empfangs-Timeout
DIP				
6	5	4		
0	0	0	keine	Standard-Timeout
0	0	1	gerade	Standard-Timeout
0	1	0	ungerade	Standard-Timeout
0	1	1	keine	30 ms
1	0	0	keine	3 x Standard-Timeout
1	0	1	gerade	3 x Standard-Timeout
1	1	0	ungerade	3 x Standard-Timeout
1	1	1	keine	60 ms

Der Empfangs-Timeout legt fest, nach welcher Zeit ein Empfangstelegramm als beendet angesehen und an die CANopen-Schnittstelle übergeben werden soll (Zeitdauer nach dem letzten empfangenen Byte). Der Abstand der einzelnen Telegrammbytes untereinander darf diese Zeit nicht überschreiten (sonst würde das Telegrammende vorzeitig erkannt).

### Busabschluss RS485/422:

DIP-Switches RS				Busabschluss
DIP				
7	8	9	10	
1	1			RS 485 gesetzt
0	0			RS 485 nicht gesetzt
1	1	1	1	RS 422 gesetzt
0	0	0	0	RS 422 nicht gesetzt

## 5 Gehäuseabmessungen



Maß	B	H	T
Größe	100	87	42

Alle Maße sind in mm angegeben.

## **6 Anhang**

### **6.1 Lieferumfang**

- Interface in der aktuellen SW- und HW-Version in Hutschienenmontage
- Bedienungsanleitung
- Diskette mit EDS-Datei
- Gegenstecker für Spannungsversorgung

### **6.2 Optional erhältliche Komponenten**

- Bedienungsanleitung (DIN A4)
- Diskette mit EDS-Datei
- Gegenstecker für Spannungsversorgung
- Gegenstecker für D-Sub-Buchsen/Stecker

### 6.3 Gewährleistung und Haftung

Für das gelieferte Interface wird innerhalb der gesetzlichen Gewährleistungsfrist die Haftung für bei Übergabe vorhandener Mängel übernommen.

Technisch bedingte Änderungen sowie Irrtum bleiben vorbehalten. Ein Anspruch auf Lieferung eines neuen Produkts besteht nicht. Der Erwerber hat Mängel innerhalb von 2 Wochen nach deren Wahrnehmung anzuzeigen. Bei Verletzung der Rügepflicht gilt der betreffende Mangel als genehmigt.

Generell sind auftretende Mängel und deren Symptome bestmöglich zu beschreiben, damit deren Reproduzierbarkeit - und damit auch Beseitigung - ermöglicht wird. Der Erwerber hat darüber hinaus kostenfrei alle erforderlichen und/oder sachdienlichen Informationen zu erteilen, gegebenenfalls Zugang und Zugriff auf und zu den fraglichen Geräten und Daten zu ermöglichen und sämtliche notwendigen Daten und Maschinenzeiten kostenlos zur Verfügung zu stellen.

Die Gewährleistung erstreckt sich nicht auf Mängel, die durch Nichteinhaltung der vorausgesetzten Einsatzbedingungen oder durch unsachgemäße Handhabung verursacht werden.

Sofern das Produkt für Testzwecke überlassen wurde und anschließend erworben wird, sind sich die Parteien einig, daß das Produkt im Rechtssinne als „gebraucht“ überlassen wurde und „wie getestet“ übernommen wurde. Gewährleistungsansprüche sind in diesem Fall ausgeschlossen.

Es gelten ergänzend die „Allgemeinen Lieferbedingungen“ für Erzeugnisse und Leistungen der Elektroindustrie.

#### 6.4 Versionsübersicht

Ver.	Datum	Bemerkungen
1.00	16.04.02	Kreuzer, Nickl: Dokument erstellt
1.10	16.12.02	Kreuzer: Neues Logo
1.20	05.05.03	Kreuzer: Layout