

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## Benutzerhandbuch



# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## Inhaltsverzeichnis

<b>1</b>	<b>ALLGEMEINES</b>	<b>4</b>
<b>2</b>	<b>ÜBERSICHT</b>	<b>5</b>
<b>3</b>	<b>TECHNISCHE INFORMATION</b>	<b>6</b>
3.1	Gesamtspezifikation	6
3.2	Gerätekonfiguration	7
3.3	Besondere Hinweise	8
3.4	Anzeigeelemente	9
3.5	System- bzw. Gerätestart	10
3.6	Telegramm-Aufbau	11
3.6.1	Header	12
3.6.2	Data-Unit	13
3.6.3	Trail	13
3.7	Beschreibung der Data-Unit in den Telegrammen	14
3.7.1	Online-Texte	15
3.7.1.1	Auswählen des Zeichensatzes	16
3.7.1.2	Setzen des Cursors	16
3.7.1.3	Setzen der Attribute	16
3.7.2	Grafiken, Texte, Variablen, Bargraphen	17
3.7.2.1	Grafikaufruf	17
3.7.2.2	Textaufruf	17
3.7.2.3	Geschwindigkeit der Lauftexte setzen	17
3.7.2.4	Variablenaufruf	18
3.7.2.5	Variable setzen	18
3.7.2.6	Variable erhöhen/erniedrigen	19
3.7.2.7	Variablenposition setzen	19
3.7.2.8	Bargraph (Wertebalken) ein-/ausblenden	20
3.7.2.9	Bargraph (Wertebalken) - Wert setzen	20
3.7.3	Direkte Grafikansteuerung	23
3.7.3.1	Anzeige löschen, füllen	23
3.7.3.2	Punkt setzen	23
3.7.3.3	Punkt aus Display lesen	23
3.7.3.4	Rechteck zeichnen	24
3.7.3.5	Scrollen der Anzeige	25
3.7.3.5.1	Anzeigen mit vertikaler Auflösung < 64 Pixel	25
3.7.3.5.2	Anzeigen mit vertikaler Auflösung > 64 Pixel	26
3.7.4	Allgemeine Funktionen	27
3.7.4.1	Blinkperiodendauer setzen	27
3.7.4.2	Helligkeitseinstellung	27

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

3.7.5	Digitale Ein-/Ausgänge	28
3.7.6	Makros	29
3.7.6.1	Makroausführung starten	29
3.7.6.2	Eingangsabhängiger Makrosprung	30
3.7.6.3	Warten bei Makroausführung	32
3.7.6.4	Makroausführung stoppen	32
<b>3.8</b>	<b>Antworttelegramm</b>	<b>33</b>
<b>3.9</b>	<b>Mehrfache ESC-Sequenzen</b>	<b>34</b>
<b>3.10</b>	<b>Beispiele</b>	<b>35</b>
<b>3.11</b>	<b>Parallele Schnittstelle</b>	<b>36</b>
3.11.1	Timing der parallelen Eingänge	36
3.11.2	Eingangsspiegel der parallelen Eingänge	36
3.11.3	Textaufruf	37
3.11.4	Grafikaufruf	38
3.11.5	Variable setzen	39
3.11.6	Makroaufruf	40
<b>3.12</b>	<b>CANopen-Interface</b>	<b>41</b>
3.12.1	Allgemeines	42
3.12.2	Gesamtspezifikation	42
3.12.3	CANopen-Protokoll	43
3.12.3.1	NMT-Telegramme	43
3.12.3.2	Nodeguard-Telegramm	44
3.12.3.3	Heartbeat-Telegramm	44
3.12.3.4	SDO-Telegramme	45
3.12.3.5	Receive-PDO-Telegramm	45
3.12.3.6	Transmit-PDO-Telegramm	45
3.12.4	RS-Datentelegramm senden (CAN -> serielle Schnittstelle)	46
3.12.5	RS-Datentelegramm empfangen (serielle Schnittstelle -> CAN)	50
3.12.6	Anwendungsbeispiele	54
3.12.7	Hinweis	57
3.12.8	DIP-Schalter / LEDs	58
<b>4</b>	<b>ANSCHLUßBELEGUNG</b>	<b>61</b>
4.1	Schnittstellenkonfiguration / LEDs	65
<b>5</b>	<b>ANHANG</b>	<b>67</b>
5.1	Darstellbare Zeichen (ASCII-Tabelle)	67
5.2	Allgemeine Hinweise	68
5.3	Konformitätserklärung	69
5.4	Gewährleistung / Haftung	70
5.5	Versionsübersicht	71

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 1 Allgemeines

Die grafikfähige Großanzeige ist als Produktionsanzeige oder Informationstafel universell einsetzbar.

Der modulare Aufbau erlaubt kostengünstige Ausführungen in verschiedenen Größen, mit unterschiedlichen Schrifthöhen und Stellenzahlen.

In der Mehrfarbenausführung (MC) können wichtige Informationen zusätzlich farbig hervorgehoben werden.

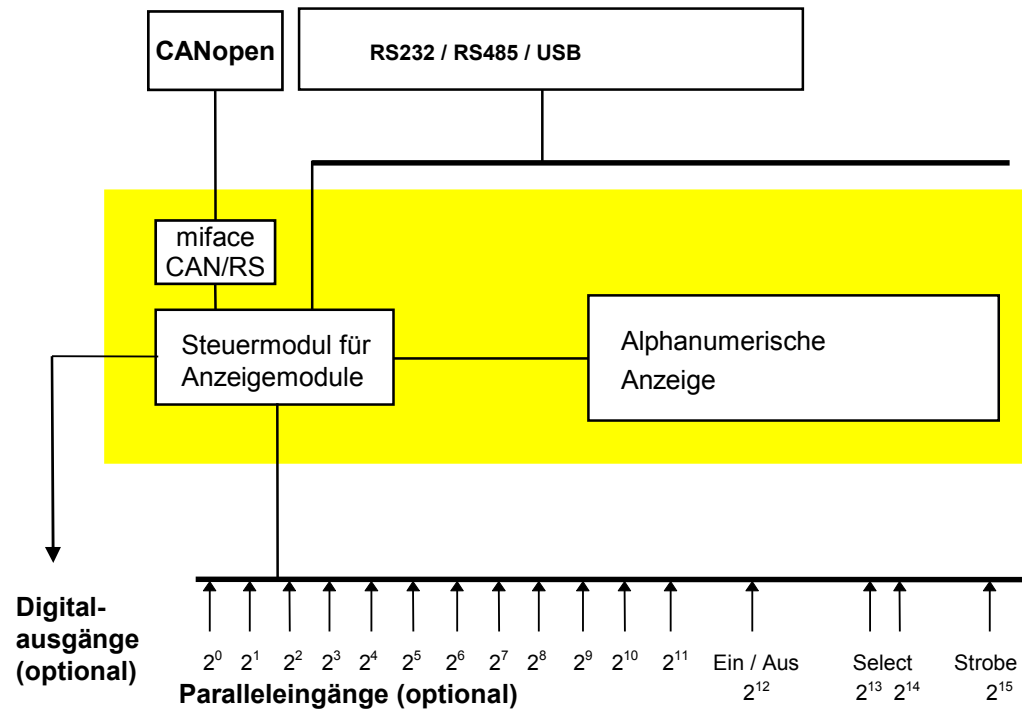
### Funktionen der Anzeige :

- Datenübertragung: CANopen, serielle Schnittstelle RS232, RS485, USB oder parallele Schnittstelle
- Konfiguration durch PC-Software (über serielle Schnittstelle)
- Darstellung von Text (verschiedene Schriftgrößen und Schriftarten) und Grafik
- Standschrift, Blinkschrift, Laufschrift, Scrollen, Inversdarstellung
- Monitoranzeige, Aufrufen gespeicherter Texte und Grafiken, Variableneinblendung, Ausführung von Makros

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 2 Übersicht



# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3 Technische Information

### 3.1 Gesamtspezifikation

Displayart:	LED-Punktmatrix-Anzeige (max. 256x128)(HxV)
Anzeige:	ASCII-Zeichen (Windows-Zeichensätze), Grafiken
Leuchtfarbe:	Typ SC: einfarbig, Typ MC: mehrfarbig
Ansicht:	einseitig oder zweiseitig
Betriebsspannung:	230 V / 50 Hz, 110 V / 60 Hz oder 24 VDC +/-20 %
Schnittstelle:	CANopen, seriell, parallel (optional)
Gehäuse:	Alu, pulverbeschichtet
Gehäusegröße:	siehe Kapitel „Gerätekonfiguration“
Befestigung:	Gelenkmontage, hängende Montage, Befestigungswinkel für Wandmontage
Schutzart:	IP54 oder IP65
Betriebstemperatur:	0...+50 °C (optional - 20...+50 °C)
Lagertemperatur:	-25...+70 °C
Grafiken:	max. 1000
Texte:	max. 1000 (max. 255 Lauftexte)
Variablen:	max. 1000
Zeichensätze:	max. 100
Bargraphen:	max. 1000
Makrobefehle:	max. 1000

Je nach vertikaler Auflösung der Großanzeige steht folgender FLASH-Speicher für Grafiken, Texte, Variable, Zeichensätze und Makros zur Verfügung:

- Vertikale Auflösung ≤ 64 Pixel: 64 KByte
- Vertikale Auflösung > 64 Pixel: 448 KByte

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.2 Gerätekonfiguration

Artikelnummer: \_\_\_\_\_

Pixelanzahl (horizontal x vertikal): \_\_\_\_\_ x \_\_\_\_\_

**Anwendung:**

für Innenbereich       für Außenbereich

**Leuchtfarbe:**

rot                       grün                       gelb  
 weiß                       blau

**Ansicht:**

einseitig                       zweiseitig

**Betriebsspannung:**

230 V / 50 Hz       110 V / 60 Hz       24 V DC

**Schutzart:**

IP40       IP54       IP65       IP \_\_\_\_\_

**Betriebstemperatur:**

bei Typ Innenanwendung:	bei Typ Außenanwendung:	Spezialvariante:
<input type="checkbox"/> 0...+50 °C (Standard)	<input type="checkbox"/> -20...+50 °C (Standard)	<input type="checkbox"/> _____ °C
	<input type="checkbox"/> -25...+50 °C (optional mit Heizung)	

**Gehäuseabmessung:**

\_\_\_\_\_ x \_\_\_\_\_ x \_\_\_\_\_ mm

**Gehäusematerial:**

Aluprofil                       Edelstahl                       Stahlblech

**Interface:**

RS232       RS485       USB  
 parallele Schnittstelle  
 CANOpen

Voreinstellungen bei Auslieferung:

RS-Baudrate: 19,2 kBaud  
 Datenbits: 8, Stopbits: 1, Parität: gerade  
 Geräteadresse: 1

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.3 Besondere Hinweise

- Beim Anstecken der Spannungsversorgung muß folgende Reihenfolge eingehalten werden:
  - Spannungsversorgungskabel mit der Anzeige verbinden.
  - Spannungsversorgungskabel mit der Spannungsquelle verbinden.
- Beim Abstecken der Spannungsversorgung muß folgende Reihenfolge eingehalten werden:
  - Spannungsversorgungskabel von der Spannungsquelle abstecken.
  - Spannungsversorgungskabel von der Anzeige abstecken.
- Achten Sie auf die richtige Schriftfarbe.  
Beispiel: Bei einer einfarbig roten Anzeige darf keine grüne Schriftfarbe verwendet werden (sonst keine Anzeige).
- Bei der Positionierung von x- und y-Koordinaten ist darauf zu achten, daß die Position auch tatsächlich in der Anzeige vorhanden ist (Auflösung in Pixel).
- Grafiken, Texte, Variablen und Bargraphen müssen so positioniert werden, daß sie vollständig auf die Anzeige passen (sonst werden sie nicht angezeigt).  
Beachten Sie bei Texten und Variablen, daß die „Unterlänge“ auch mit auf die Anzeige passen muß.
- Wollen Sie die Anzeige über die serielle Schnittstelle ansteuern, so wird empfohlen zuerst die Beispiele in Kapitel „Beispiele“ auszuprobieren.
- Wollen Sie die Anzeige über das CANopen-Interface ansteuern, so wird empfohlen zuerst Kapitel „CANopen-Interface“ zu lesen. Dort sind Beispiele enthalten, die sofort ausprobiert werden können (Kapitel „Anwendungsbeispiele“).



# migra MPB CAN

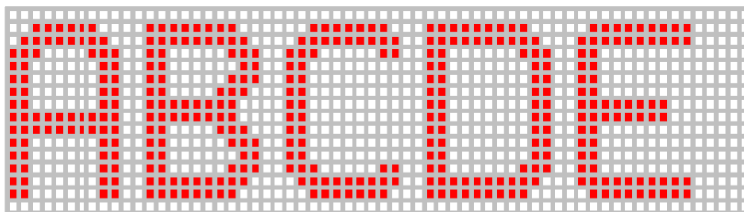
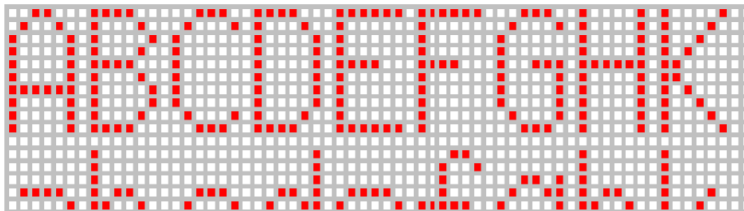
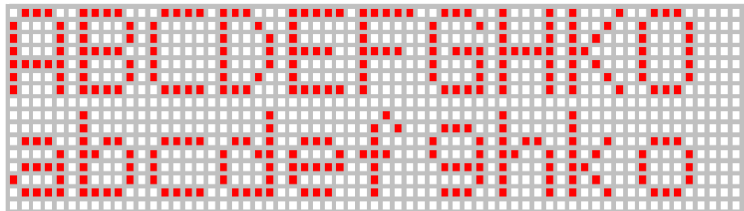
Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.4 Anzeigeelemente

Als Anzeigeelemente dienen alphanumerische Anzeigemodule mit einer 16 x 16, 64 x 8 oder 64 x 16 Punktmatrix.

Achtung: Softwaretechnisch gibt es keine Unterscheidung zwischen „16-Pixelzeilen-Modulen“ und „8-Pixelzeilen-Modulen“. Bei „8-Pixelzeilen-Modulen“ sind lediglich die untersten 8 Zeilen nicht sichtbar.

Im Folgenden ist ein „64 x 16-Pixelmodul“ mit drei verschiedenen Ziffernhöhen (Beispiele) dargestellt:



# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.5 System- bzw. Gerätestart

Die Großanzeige führt während des Hochlaufs interne Speicher- und Funktionstests durch (Dauer: weniger als 1 Sekunde).

Bleibt die Anzeige dunkel (und blinkt die geräteinterne Funktions-LED mit ca. 0,5Hz), so befindet sich das Gerät im Bootmodus. Das bedeutet, dass sich zurzeit keine vollständige Software im Flash befindet. Dies kann passieren, falls ein vorheriger Upload abgebrochen wurde. Der Upload muss in diesem Fall wiederholt werden (mit Hilfe der PC-Software MKS).

Es erscheint die Einstellung der seriellen Schnittstelle:

- \* Geräte-Adresse (ID)
- \* Baudrate
- \* Anzahl der Datenbits
- \* Art des Paritätsbits
- \* Anzahl der Stopbits.

Nach dem Hochlauf wird automatisch die Makroausführung beim ersten Makro gestartet (falls vorhanden). Soll die Anzeige sofort wieder gelöscht werden, so muss ein entsprechendes Makrokommando vorhanden sein!

Anschließend wird auf gültige Nutzdaten vom Anwender gewartet (über eine der Schnittstellen).

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.6 Telegramm-Aufbau

Das Telegramm an die Großanzeige besteht aus 3 Teilen:

<b>Header</b>	<b>Data-Unit</b>	<b>Trail</b>
---------------	------------------	--------------

Telegramme an die Großanzeige werden geräteintern erst 3 bis 240 ms nach Erhalt des letzten Telegrammbytes ausgewertet (= „Empfangstimeout“). Der Abstand der einzelnen Telegrammbytes untereinander darf diese Zeitdauer nicht überschreiten!

Nach Erhalt des Antworttelegramms kann sofort das nächste Telegramm gesendet werden.

Wird ohne Antworttelegramm gearbeitet, so ist die Großanzeige erst nach Abarbeitung des letzten Telegramminhalts wieder bereit, ein neues Telegramm zu empfangen. Wird also z. B. eine große Grafik eingeblendet, so muß länger gewartet werden, als zur Ausgabe eines „Online-Zeichens“. In der Regel ist ein Telegrammabstand von („Empfangstimeout“ + 150 ms) ausreichend.

Sollen nacheinander unterschiedliche Großanzeigen (mit unterschiedlicher Geräte-Adresse) am selben RS485-Bus angesprochen werden, so darf ebenfalls der „Empfangstimeout“ (3-240 ms) nicht unterschritten werden.

Für die Zusammenarbeit der PC-Software (MKS) mit einer RS485-Schnittstelle muß in der Regel ein „Empfangstimeout“ von mindestens 30 ms eingestellt werden.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.6.1 Header

STX	DA	SA	FC	LEN-H (optional)	LEN-L (optional)
Start of Text	Zieladresse	Quelladresse	Frame Control	Highnibble Anzahl der Datenbytes	Lownibble Anzahl der Datenbytes
00000010 <sub>b</sub> (02 <sub>h</sub> )	1XXXXXXXX <sub>b</sub>	1XXXXXXXX <sub>b</sub>	1XXXXXXXX <sub>b</sub>	1111XXXX <sub>b</sub>	1111XXXX <sub>b</sub>

- STX:** Start of Text: 02<sub>h</sub>
- DA:** Destination Address: Zieladresse  
Bit 7 muß gesetzt sein ⇒ mögliche Adressen: 0 - 126<sub>d</sub>, 127<sub>d</sub> für Broadcast
- SA:** Source Address: Quelladresse  
Bit 7 muß gesetzt sein ⇒ mögliche Adressen: 0 - 126<sub>d</sub>
- FC:** Frame Control: Steuerung des Kommunikationsablaufes  
Bit 7: fest auf 1  
Bit 6-2: reserviert (0)  
Bit 1: 0 -> keine Prüfsumme verwenden (LEN-H/L, CHK-H/L nicht verwenden)  
1 -> Prüfsumme verwenden (LEN-H/L, CHK-H/L verwenden)  
Bit 0: 0 -> keine Antwort senden  
1 -> Antwort senden
- LEN-H:** Längen-Highnibble: Highnibble der Anzahl der Datenbytes, Bits 4-7 müssen gesetzt sein  
(z.B. Anzahl der Datenbytes = 26<sub>h</sub> ⇒ Highnibble = 2 ⇒ 11110010<sub>b</sub>)
- LEN-L:** Längen-Lownibble: Lownibble der Anzahl der Datenbytes, Bits 4-7 müssen gesetzt sein  
(z.B. Anzahl der Datenbytes = 26<sub>h</sub> ⇒ Lownibble = 6 ⇒ 11110110<sub>b</sub>).

Nur bei Verwendung einer Prüfsumme muß LEN-H und LEN-L zusätzlich gesendet werden. Wird Bit 1 im FC-Byte nicht gesetzt, so dürfen LEN-H und LEN-L nicht mitgesendet werden!

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.6.2 Data-Unit

Data-Unit
Anzeigedaten
1B <sub>h</sub> , 0A <sub>h</sub> , 0D <sub>h</sub> , 20 <sub>h</sub> -FF <sub>h</sub>

**Data-Unit:** Datenbytes (ASCII-Zeichen, Steuerkommandos).

## 3.6.3 Trail

Falls im FC-Byte Bit 1 **nicht** gesetzt ist (**keine** Prüfsumme verwenden):

ETX
End of Text
00000011 <sub>b</sub> (03 <sub>h</sub> )

**End of Text:** 03<sub>h</sub>.

Falls im FC-Byte Bit 1 gesetzt ist (Prüfsumme verwenden):

CHK-H	CHK-L	ETX
Highnibble der Summe aller vorangegangenen Bytes (ohne STX)	Lownibble der Summe aller vorangegangenen Bytes (ohne STX)	End of Text
1111XXXX <sub>b</sub> (FX <sub>h</sub> )	1111XXXX <sub>b</sub> (FX <sub>h</sub> )	00000011 <sub>b</sub> (03 <sub>h</sub> )

**Prüfsumme =LOW-Byte der Summe aus DA, SA, FC, LEN-H, LEN-L und allen Datenbytes**

**CHK-H:** Highnibble der Prüfsumme: Bits 3-0  
Bits 4-7 müssen gesetzt (1) sein.

**CHK-L:** Lownibble der Prüfsumme: Bits 3-0  
Bits 4-7 müssen gesetzt (1) sein.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.7 Beschreibung der Data-Unit in den Telegrammen

Die Konfiguration der Anzeige muß durch die PC-Software (micon 5) erfolgen (Grafiken, Texte, Variablen, Zeichensätze, Bargraphen und Makros definieren). Die einzelnen Elemente der in die Anzeige geladenen Konfiguration können dann von den in diesem Kapitel beschriebenen Telegrammen verwendet werden.

Die Anzeige ist im Auslieferungszustand mit einer Konfiguration vorprogrammiert. Entsprechend Ihrer Anwendung können Sie eigene Konfigurationen mit der PC-Software erstellen und zur Anzeige senden. Beachten Sie dabei, daß die bestehende Konfiguration in der Anzeige überschrieben wird.

Alle Indexe sind 0-basiert: d.h., um den ersten Text aufzurufen, wird „000“ übergeben. Die Position 0 / 0 (x / y) ist der Punkt in der linken oberen Ecke des Displays. Alle Grafiken, Texte, Variablen und Bargraphen werden von der angegebenen x- / y-Position ausgehend in das Display nach rechts unten geschrieben. Dabei dürfen die Anzeigegrenzen nicht überschritten werden (ansonsten keine Anzeige).

### Hinweis:

**Die Indexe von Grafiken, Texten, Variablen, Zeichensätzen und Bargraphen sind sowohl in der PC-SW als auch bei der Ansteuerung 0-basiert!**

Telegramme, die keinen „Online Text“ (ASCII-Codes 20<sub>h</sub> bis FF<sub>h</sub>, 0A<sub>h</sub>, 0D<sub>h</sub>) beinhalten, beginnen mit einem Escape-Zeichen (1B<sub>h</sub>) als erstes Byte in der Data-Unit. Es muß für jede Escape-Sequenz ein eigenes Telegramm gesendet werden.

Bei Verwendung von Antworttelegrammen kann das nächste Telegramm sofort nach Erhalt der Antwort gesendet werden. Hierbei kann es jedoch zu Verzögerungen bei der Ausführung von Makros, Laufschriften und dem „Scrollen“ kommen, falls die Telegrammfolge zu schnell wird.

Bei Verzicht auf Antworttelegramme muß die in 3.6 erwähnte Verzögerung zwischen den Telegrammen eingehalten werden.

Die Datenbytes der Data-Unit müssen in der Regel ASCII-codiert werden!

Beispiel „Positionsangabe“:

Zur Angabe der Position 123<sub>d</sub> muß ... 31<sub>h</sub> 32<sub>h</sub> 33<sub>h</sub> ... gesendet werden!  
(ASCII-Zeichen „1“, „2“ und „3“.)

Eine ASCII-Tabelle befindet sich in Kapitel „Darstellbare Zeichen“.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.7.1 Online-Texte

Ein Online-Text wird ohne Escape-Sequenz übertragen (ASCII-Codes  $20_{\text{h}}$ - $\text{FF}_{\text{h}}$ ,  $0\text{A}_{\text{h}}$ ,  $0\text{D}_{\text{h}}$ ).

Die übertragenen ASCII-Zeichen werden mit dem aktuellen Zeichensatz an der aktuellen Cursorposition unter Berücksichtigung der aktuellen Attribute angezeigt.

Der Zeilenumbruch erfolgt mit Hilfe der ASCII-Codes  $0\text{A}_{\text{h}}$  oder  $0\text{D}_{\text{h}}$ , oder durch Senden der ESCAPE-Sequenz zum Setzen der Cursorposition. (ESC-„C“.)

Beim Überschreiten der Anzeigegrenzen wird die Anzeige in der folgenden Zeile bzw. in der ersten Zeile fortgesetzt.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.7.1.1 Auswählen des Zeichensatzes

Byte 1	Byte 2	Byte 3	Byte 4
ESC	Funktion	Zeichensatz-Nr. 10er	Zeichensatz-Nr. 1er
1Bh	„Z“ (5Ah): normale Zeichenbreite „Z“ (7Ah): Sperrschrift (gleiche Breite aller Zeichen erzwingen)	„0“-„9“ (30h - 39h)	„0“-„9“ (30h - 39h)

Im Auslieferungszustand sind in der Anzeige 3 Zeichensätze gespeichert:

- 50 mm Zeichensatz (Zeichensatz „Z00“, „z00“)
- 75 mm Zeichensatz (Zeichensatz „Z01“, „z01“)
- 100 mm Zeichensatz (Zeichensatz „Z02“, „z02“).

Wird eine neue Konfiguration in die Anzeige geladen, so werden die vorhandenen Zeichensätze mit den neuen Zeichensätzen überschrieben!

## 3.7.1.2 Setzen des Cursors

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
ESC	Funktion	x-Position 100er	x-Position 10er	x-Position 1er	y-Position 100er	y-Position 10er	y-Position 1er
1Bh	„C“ (43h)	„0“-„9“ (30h-39h)	„0“-„9“ (30h-39h)	„0“-„9“ (30h-39h)	„0“-„9“ (30h-39h)	„0“-„9“ (30h-39h)	„0“-„9“ (30h-39h)

## 3.7.1.3 Setzen der Attribute

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
ESC	Funktion	Vordergrundfarbe	Hintergrundfarbe	Blinken
1Bh	„A“ (41h)	„0“ (30h): schwarz „1“ (31h): grün „2“ (32h): rot „3“ (33h): gelb	„0“ (30h): schwarz „1“ (31h): grün „2“ (32h): rot „3“ (33h): gelb „T“ (54h): transparent	„0“ (30h): statisch „1“ (30h): blinkend



# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.7.2 Grafiken, Texte, Variablen, Bargraphen

### 3.7.2.1 Grafikaufruf

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
ESC	Funktion	Ein/Ausblenden	Grafik Nr. 100er	Grafik Nr. 10er	Grafik Nr. 1er
1B <sub>h</sub>	„G“ (47 <sub>h</sub> )	„+“ (2B <sub>h</sub> ): Einblenden „-“ (2D <sub>h</sub> ): Ausblenden	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )

Beim Ausblenden wird die von der Grafik beanspruchte Fläche mit der aktuellen Online-Hintergrundfarbe (vom letzten „ESC-A“-Telegramm) überschrieben! Falls diese „transparent“ gewählt ist, so wird „schwarz“ verwendet!

### 3.7.2.2 Textaufruf

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
ESC	Funktion	Ein/Ausblenden	Text Nr. 100er	Text Nr. 10er	Text Nr. 1er
1B <sub>h</sub>	„T“ (54 <sub>h</sub> )	„+“ (2B <sub>h</sub> ): Einblenden „-“ (2D <sub>h</sub> ): Ausblenden	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )

Beim Ausblenden wird die vom Text beanspruchte Fläche mit der aktuellen Online-Hintergrundfarbe (vom letzten „ESC-A“-Telegramm) überschrieben! Falls diese „transparent“ gewählt ist, so wird „schwarz“ verwendet!

### 3.7.2.3 Geschwindigkeit der Lauftexte setzen

Byte 1	Byte 2	Byte 3
ESC	Funktion	Laufgeschwindigkeit
1B <sub>h</sub>	„L“ (4C <sub>h</sub> )	„0“ (30 <sub>h</sub> ): statisch „1“ (31 <sub>h</sub> ): 1,8 sek : „9“ (39 <sub>h</sub> ): 0,2 sek

Alle Lauftexte werden mit dieser Geschwindigkeit bewegt. Nach dem Einschalten des Gerätes ist „9“ (0,2 Sekunden pro Laufschrift) voreingestellt.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.7.2.4 Variablenuufruf

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
ESC	Funktion	Ein/Ausblenden	Var. Nr. 100er	Var. Nr. 10er	Var. Nr. 1er
1B <sub>h</sub>	„V“ (56 <sub>h</sub> )	„+“ (2B <sub>h</sub> ): Einblenden „-“ (2D <sub>h</sub> ): Ausblenden	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )

Beim Ausblenden wird die von der Variable beanspruchte Fläche mit der aktuellen Online-Hintergrundfarbe (vom letzten „ESC-A“-Telegramm) überschrieben! Falls diese „transparent“ gewählt ist, so wird „schwarz“ verwendet!

## 3.7.2.5 Variable setzen

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7...133
ESC	Funktion	Setzen	Var. Nr. 100er	Var. Nr. 10er	Var. Nr. 1er	Variablen- werte
1B <sub>h</sub>	„V“ (56 <sub>h</sub> )	„=“ (3D <sub>h</sub> ): Setzen	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	(20 <sub>h</sub> -FF <sub>h</sub> )

Eine Variable darf max. 127 Zeichen beinhalten (die Länge der Variable wird bei der Konfiguration festgelegt).

Es werden lediglich soviele Zeichen der Variable überschrieben, wie im Telegramm übergeben werden.

Um ein „Flackern“ zu vermeiden, wird vor der Neuanzeige nicht gelöscht!

Für eine ordnungsgemäße Anzeige darf die Hintergrundfarbe der Variable nicht „transparent“ sein und es sollte ein Sperrschrift-Zeichensatz (mit konstanter Zeichenbreite) verwendet werden. Anderenfalls muß die Variable vor der Veränderung erst ausgeblendet und anschließend wieder eingeblendet werden!

Der Variableninhalt wird lediglich in RAM gespeichert. Nach dem Aus-/Einschalten enthält die Variable wieder ihren konfigurierten Wert.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.7.2.6 Variable erhöhen/erniedrigen

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
ESC	Funktion	Erhöhen / Erniedrigen	Var. Nr. 100er	Var. Nr. 10er	Var. Nr. 1er
1B <sub>h</sub>	„V“ (56 <sub>h</sub> )	„I“ (49 <sub>h</sub> ): Inkrement oder „D“ (44 <sub>h</sub> ): Dekrement	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )

Es werden nur numerische Zeichen verändert. Buchstaben, Kommas usw. werden übersprungen. Die numerischen Zeichen werden als eine zusammengehörende Dezimalzahl interpretiert. Diese Dezimalzahl wird um eins erhöht oder erniedrigt.

Um ein „Flackern“ zu vermeiden, wird vor der Neuanzeige nicht gelöscht!

Für eine ordnungsgemäße Anzeige darf die Hintergrundfarbe der Variable nicht „transparent“ sein und es sollte ein Sperrschrift-Zeichensatz (mit konstanter Zeichenbreite) verwendet werden. Anderenfalls muß die Variable vor der Veränderung erst ausgeblendet und anschließend wieder eingeblendet werden!

Der Variableninhalt wird lediglich in RAM gespeichert. Nach dem Aus-/Einschalten enthält die Variable wieder ihren konfigurierten Wert.

## 3.7.2.7 Variablenposition setzen

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
ESC	Funktion	Setzen	Var. Nr. 100er	Var. Nr. 10er	Var. Nr. 1er
1B <sub>h</sub>	„V“ (56 <sub>h</sub> )	„P“ (50 <sub>h</sub> ): Pos. setzen	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )

Byte 7	Byte 8	Byte 9	Byte 10	Byte 11	Byte 12
x-Position 100er	x-Position 10er	x-Position 1er	y-Position 100er	y-Position 10er	y-Position 1er
„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )

Die Position der Variable wird im RAM abgelegt. Nach dem Aus-/Einschalten befindet sich die Variable wieder an ihrer konfigurierten Position.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.7.2.8 Bargraph (Wertebalken) ein-/ausblenden

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
ESC	Funktion	Ein-/Ausblenden	Bargraph Nr. 100er	Bargraph Nr. 10er	Bargraph Nr. 1er
1Bh	„W“ (57h)	„+“ (2Bh): Einblenden „-“ (2Dh): Ausblenden	„0“-„9“ (30h-39h)	„0“-„9“ (30h-39h)	„0“-„9“ (30h-39h)

Beim Einblenden wird der zuletzt geschickte Wert (bzw. der Bezugswert nach RESET) zur Anzeige des Bargraphen verwendet. Ist eine Variable mit dem Bargraphen verbunden, so wird auch diese entsprechend angezeigt.

Beim Ausblenden wird die vom Bargraphen beanspruchte Fläche mit der aktuellen Online-Hintergrundfarbe (vom letzten „ESC-A“-Telegramm) überschrieben! Falls diese „transparent“ gewählt ist, so wird „schwarz“ verwendet! Ist eine Variable mit dem Bargraphen verbunden, so wird auch diese ausgeblendet.

Es sind maximal 255 Bargraphen möglich (Nr. 0 - 254).

Die maximale Variablenlänge zugeordneter Variablen beträgt 127 Zeichen.

## 3.7.2.9 Bargraph (Wertebalken) - Wert setzen

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
ESC	Funktion	Setzen	Bargraph Nr. 100er	Bargraph Nr. 10er	Bargraph Nr. 1er	Art der übergebenen Daten
1Bh	„W“	„=“: Setzen	„0“-„9“	„0“-„9“	„0“-„9“	„A“: ASCII-codierter Dezimalwert

Byte 8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13
Vorzeichen	Dezimalwert 10000er	Dezimalwert 1000er	Dezimalwert 100er	Dezimalwert 10er	Dezimalwert 1er
„+“, „-“	„0“-„9“	„0“-„9“	„0“-„9“	„0“-„9“	„0“-„9“

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

Sollte der Bargraph noch nicht eingeblendet sein, so wird dies automatisch nachgeholt.

Der Bargraph-Balken wird entsprechend seiner Lage zwischen den in der Konfiguration festgelegten MIN- und MAX-Grenzen angezeigt. Der Balkenanfang liegt dabei immer beim in der Konfiguration festgelegten Bezugswert. Das Balkenende liegt beim übergebenen Wert.

An der Stelle des Bezugswerts wird der Balken in seiner konfigurierten Farbe gezeichnet.

Überschreitet der Balken eine der Farbgrenzen (ausgehend vom Bezugswert), so wird nach diesem Punkt die mit der Farbgrenze festgelegte Farbe angezeigt.

Vier Farbgrenzen sind definiert. Jede davon muss im Bereich MIN-Grenze bis MAX-Grenze liegen:

MIN-Grenze  $\leq$  Farbgrenze 1  $\leq$  Farbgrenze 2  $\leq$  Farbgrenze 3  $\leq$  Farbgrenze 4  $\leq$  MAX-Grenze.

Die Einhaltung dieser Reihenfolge ist durch die PC-Software sichergestellt !

Neben dem Standard-Bargraphen (mehrfarbiger Balken) ist auch noch die Darstellung als einfarbiger Balken bzw. als einfarbige Marke möglich (abhängig von den Konfigurationsdaten - siehe PC-Software). Die Balken- bzw. Markenfarbe entspricht dann der Farbe des Balkenendes des Standard-Bargraphen.

Wird die MIN- bzw. MAX-Grenze unter- bzw. überschritten, so wird an der jeweiligen Grenze eine blinkende Marke angezeigt.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

Ist eine Variable mit dem Bargraphen verbunden, so wird auch diese entsprechend verändert:

Von rechts beginnend werden alle aus dem Zeichen „#“ oder „\*“ bestehenden Stellen mit dem neuen Wert überschrieben.

Wird eine Variable mit „#“ vorbelegt, so werden führende Nullen unterdrückt (durch Leerzeichen ersetzt).

Bei Vorbelegung mit „\*“ werden auch führende Nullen mit angezeigt.

Sollte eine Stelle aus dem Zeichen „\$“ bestehen, so wird diese mit dem neuen Vorzeichen überschrieben.

<b>Beispiel:</b>	Variablenvorbelegung	= „\$ #*, * m/s“
	Wert	= -9 = „-00009“
	=> Anzeige	= „- 0,9 m/s“

Sollte eine Bereichsüberschreitung bzw. -unterschreitung vorliegen (MIN-/MAX-Grenzen), so wird der aktuelle Wert blinkend angezeigt.

Für eine ordnungsgemäße Anzeige darf die Hintergrundfarbe der Variable nicht „transparent“ sein und es muß ein Sperrschrift-Zeichensatz (mit konstanter Zeichenbreite) verwendet werden.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.7.3 Direkte Grafikansteuerung

### 3.7.3.1 Anzeige löschen, füllen

Byte 1	Byte 2	Byte 3
ESC	Funktion	Farbe
1B <sub>h</sub>	„F“ (46 <sub>h</sub> )	„0“ (30 <sub>h</sub> ): schwarz „1“ (31 <sub>h</sub> ): grün „2“ (32 <sub>h</sub> ): rot „3“ (33 <sub>h</sub> ): gelb

### 3.7.3.2 Punkt setzen

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9
ESC	Funktion	Farbe	x-Pos. 100er	x-Pos. 10er	x-Pos. 1er	y-Pos. 100er	y-Pos. 10er	y-Pos. 1er
1B <sub>h</sub>	„P“ (50 <sub>h</sub> )	„0“ (30 <sub>h</sub> ): schwarz „1“ (31 <sub>h</sub> ): grün „2“ (32 <sub>h</sub> ): rot „3“ (33 <sub>h</sub> ): gelb	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )

### 3.7.3.3 Punkt aus Display lesen

Anfrage (FC-Byte, Bit 0 = 1):

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9
ESC	Funktion	Abfrage- Kennung	x-Pos. 100er	x-Pos. 10er	x-Pos. 1er	y-Pos. 100er	y-Pos. 10er	y-Pos. 1er
1B <sub>h</sub>	„P“ (50 <sub>h</sub> )	„2“ (3F <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )

Antwort:

Farbinfo (mit Header und Trail)

Byte 1	Byte 2	Byte 3
ESC	Funktion	Farbe
1B <sub>h</sub>	„P“ (50 <sub>h</sub> )	„0“ (30 <sub>h</sub> ): schwarz „1“ (31 <sub>h</sub> ): grün „2“ (32 <sub>h</sub> ): rot „3“ (33 <sub>h</sub> ): gelb

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.7.3.4 Rechteck zeichnen

Byte 1	Byte 2	Byte 3	Byte 4
ESC 1B <sub>h</sub>	Funktion „R“ (52 <sub>h</sub> )	Vordergrundfarbe (Rahmen) „0“ (30 <sub>h</sub> ): schwarz „1“ (31 <sub>h</sub> ): grün „2“ (32 <sub>h</sub> ): rot „3“ (33 <sub>h</sub> ): gelb	Hintergrundfarbe (Füllung) „0“ (30 <sub>h</sub> ): schwarz „1“ (31 <sub>h</sub> ): grün „2“ (32 <sub>h</sub> ): rot „3“ (33 <sub>h</sub> ): gelb „T“ (54 <sub>h</sub> ): transparent

Position linke obere Ecke:

Byte 5	Byte 6	Byte 7	Byte 8	Byte 9	Byte 10
x-Position 100er	x-Position 10er	x-Position 1er	y-Position 100er	y-Position 10er	y-Position 1er
„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )

Position rechte untere Ecke:

Byte 11	Byte 12	Byte 13	Byte 14	Byte 15	Byte 16
x-Position 100er	x-Position 10er	x-Position 1er	y-Position 100er	y-Position 10er	y-Position 1er
„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )

Das Rechteck wird mit der Vordergrundfarbe gezeichnet.  
Das Rechteck wird mit der Hintergrundfarbe ausgefüllt.



# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.7.3.5 Scrollen der Anzeige

### 3.7.3.5.1 Anzeigen mit vertikaler Auflösung < 64 Pixel

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9
ESC	Funktion	Richtung	Geschwindigkeit	Schrittweite	Startzeile 10er	Startzeile 1er	Endzeile 10er	Endzeile 1er
1Bh	„S“	„0“: aus „1“: oben „2“: unten	„0“: statisch „1“: 1,8 sek „9“: 0,2 sek	„1“: 1Pixel „9“: 9 Pixel „0“: nicht scrollen	„0“-„9“	„0“-„9“	„0“-„9“	„0“-„9“

Scrollt einen Bildschirmausschnitt von der Start- bis zur Endzeile einmal (Geschwindigkeit = „statisch“) oder zyklisch um 1 bis 9 Pixel.

Die Start-/Endzeile gibt die y-Position der ersten/letzten Pixel-Zeile des Scrollbereichs an (Endzeile > Startzeile!). Es wird immer nur der zuletzt eingestellte Scrollbereich „gescrollt“!

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.7.3.5.2 Anzeigen mit vertikaler Auflösung > 64 Pixel

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9	Byte 10	Byte 11
ESC	Funktion	Richtung	Geschwindigkeit	Schrittweite	Startzeile 100er	Startzeile 10er	Startzeile 1er	Endzeile 100er	Endzeile 10er	Endzeile 1er
1Bh	„S“	„0“: aus „1“: oben „2“: unten	„0“: statisch „1“: 1,8 sek „9“: 0,2 sek	„1“: 1 Pixel „9“: 9 Pixel „0“: nicht scrollen	„0“-„9“	„0“-„9“	„0“-„9“	„0“-„9“	„0“-„9“	„0“-„9“

Scrollt einen Bildschirmausschnitt von der Start- bis zur Endzeile einmal (Geschwindigkeit = „statisch“) oder zyklisch um 1 bis 9 Pixel.  
Die Start-/Endzeile gibt die y-Position der ersten/letzten Pixel-Zeile des Scrollbereichs an (Endzeile > Startzeile!). Es wird immer nur der zuletzt eingestellte Scrollbereich „gescrollt“!

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.7.4 Allgemeine Funktionen

### 3.7.4.1 Blinkperiodendauer setzen

Byte 1	Byte 2	Byte 3
ESC	Funktion	Periodendauer
1B <sub>h</sub>	„B“ (42 <sub>h</sub> )	„0“ (30 <sub>h</sub> ): 2 Sekunden : „9“ (39 <sub>h</sub> ): 0,2 Sekunden

Alle blinkenden Texte blinken mit dieser Blinkperiodendauer.  
Nach dem Einschalten des Geräts ist „9“ (0,2 Sekunden) voreingestellt.

### 3.7.4.2 Helligkeitseinstellung

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
ESC	Funktion	Farbe	Helligkeit 100er	Helligkeit 10er	Helligkeit 1er
1B <sub>h</sub>	„H“ (48 <sub>h</sub> )	„1“-„2“ (31 <sub>h</sub> -32 <sub>h</sub> )	„0“-„1“ (30 <sub>h</sub> -31 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )

Jeweils im Bereich von 0...100 %.  
Farbe „1“ = grün  
Farbe „2“ = rot.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.7.5 Digitale Ein-/Ausgänge

Die Großanzeige verfügt über bis zu 16 digitale Eingänge und 16 digitale Ausgänge (optional).

Anfrage (FC-Byte, Bit 0 = 1):

Byte 1	Byte 2	Byte 3	Byte 2+n	Byte 18
ESC	Funktion	Output Bit 1	Output Bit n	Output Bit 16
1Bh	„D“ (44h)	„0“ oder „1“ (30h oder 31h)	„0“ oder „1“ (30h oder 31h)	„0“ oder „1“ (30h oder 31h)

„0“ -> Ausgang löschen

„1“ -> Ausgang setzen

sonst -> Ausgang in altem Zustand lassen.

Antwort:

Byte 1	Byte 2	Byte 3	Byte 2+n	Byte 18
ESC	Funktion	Input Bit 1	Input Bit n	Input Bit 16
1Bh	„D“ (44h)	„0“ oder „1“ (30h oder 31h)	„0“ oder „1“ (30h oder 31h)	„0“ oder „1“ (30h oder 31h)

„0“ -> Eingang gelöscht

„1“ -> Eingang gesetzt.

Die Ausgangsbits werden zum Schalten der digitalen Ausgänge (soweit hardwaremäßig vorhanden) verwendet.

Nach dem Einschalten sind alle Ausgänge gelöscht (0).

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.7.6 Makros

Makros sind in der Konfiguration vordefinierte Befehlssequenzen. Sie entsprechen der Data-Unit in den Ansteuer-Telegrammen.

Nach dem Einschalten der Anzeige wird die Makroausführung beim ersten Makrokommando gestartet (falls vorhanden). Alle 100 ms wird jeweils das nächste Makro ausgeführt. Nach Erreichen des letzten Makros stoppt die Makroausführung.

### 3.7.6.1 Makroausführung starten

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
ESC	Funktion	Makro Nr. 100er	Makro Nr. 10er	Makro Nr. 1er
1B <sub>h</sub>	„M“ (4D <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )

Die Makroausführung startet beim angegebenen Makro.

Durch Verwendung dieses Kommandos in einer Makrosequenz sind auch „Sprünge“ bei der Makroausführung möglich.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.7.6.2 Eingangsabhängiger Makrosprung

### Variante A:

Makroausführung an der Stelle „Makro Nr.“ („000“-„999“) fortsetzen, falls der angegebene Eingang („0“-„F“) gleich „0“ oder „1“ ist:

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
ESC	Funktion	Makro Nr. 100er	Makro Nr. 10er	Makro Nr. 1er	„E“ (fest)
1B <sub>h</sub>	„M“ (4D <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„E“ (45 <sub>h</sub> )

→

Byte 7	Byte 8	Byte 9
Nr. des abzufragenden Eingangs (0-15)	„=“ (fest)	abgefragter Zustand (0/1)
„0“-„F“ (30 <sub>h</sub> -46 <sub>h</sub> )	„=“ (3D <sub>h</sub> )	„0“ oder „1“ (30 <sub>h</sub> oder 31 <sub>h</sub> )

### Variante B:

Makroausführung an der Stelle „Makro Nr.“ („000“-„999“) fortsetzen, falls der angegebene Eingang („0“-„F“) seit der letzten Abfrage (mit dem Kommando „ESC-MXXXE...“) mindestens einmal mit dem Zustand „0“ oder „1“ eingelesen wurde.

Das Einlesen der Digitaleingänge erfolgt in der Regel alle 100 ms.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
ESC	Funktion	Makro Nr. 100er	Makro Nr. 10er	Makro Nr. 1er	„E“ (fest)
1B <sub>h</sub>	„M“ (4D <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„E“ (45 <sub>h</sub> )

→

Byte 7	Byte 8	Byte 9
Nr. des abzufragenden Eingangs (0-15)	„#“ (fest)	abgefragter Zustand (0/1)
„0“-„F“ (30 <sub>h</sub> -46 <sub>h</sub> )	„#“ (23 <sub>h</sub> )	„0“ oder „1“ (30 <sub>h</sub> oder 31 <sub>h</sub> )

### **Hinweise:**

Falls einer dieser Befehle (Variante A oder B) aus der Makroliste heraus aufgerufen wurde, wird sofort das nächste Makro ausgeführt (ohne 100 ms zu warten und ohne die Digitaleingänge neu einzulesen).

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

Spätestens nachdem 30 dieser Makros („ESC-MXXE...“) in der Makroliste hintereinander ausgeführt wurden, wird eine Pause von 100 ms eingeschoben und die Digitaleingänge werden neu eingelesen.

## Beispiel:

	<b>Makro Nr.</b>	<b>Makrokommando</b>
1	000	ESC-M005E0=1
2	001	ESC-M007E1#1
3	002	ESC-M009E2=1
4	003	ESC-w010
5	004	ESC-M000
6	005	ESC-T+000
7	006	ESC-M000
8	007	ESC-T+001
9	008	ESC-M000
10	009	ESC-F0
11	010	ESC-M000

- \* Ca. alle 1 s werden die Eingänge 0 bis 2 abgefragt.
- \* Bei gesetztem Eingang 0 wird Text „000“ angezeigt.
- \* Bei gesetztem Eingang 1 wird Text „001“ angezeigt.  
Dies geschieht auch dann, wenn der Eingang nur kurzfristig (mindestens 100 ms) -z.B. während der Wartezeit (1 s)- gesetzt war.
- \* Bei gesetztem Eingang 2 wird die Anzeige gelöscht.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.7.6.3 Warten bei Makroausführung

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
ESC	Funktion	Wartezeit 100er	Wartezeit 10er	Wartezeit 1er
1B <sub>h</sub>	„W“ (77 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )	„0“-„9“ (30 <sub>h</sub> -39 <sub>h</sub> )

Setzt die Wartezeit bis zur Ausführung des nächsten Makros (Einheit: 100 ms.).

Normalerweise wird alle 100 ms. ein Makro ausgeführt, bis das letzte Makro erreicht ist.

Nach der Sequenz ESC + „w...“ wird die Makroausführung für die angegebene Zeit ausgesetzt.

Diese Sequenz kann sowohl Bestandteil eines Makros als auch Bestandteil eines Empfangstelegramms sein.

## 3.7.6.4 Makroausführung stoppen

Byte 1	Byte 2
ESC	Funktion
1B <sub>h</sub>	„E“ (45 <sub>h</sub> )



# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.8 Antworttelegramm

Das Antworttelegramm wird nur gesendet, falls das entsprechende Bit (Bit 0) im FC-Byte gesetzt war und als Zieladresse nicht die Broadcastadresse (127) verwendet wurde.

Beispiel :      Großanzeigenadresse = 1,  
                  Adresse des ansteuernden Gerätes = 0

Antworttelegramm der Großanzeige zum ansteuernden Gerät (MASTER):

STX	DA	SA	FC	Data-Unit	ETX
Start of Text	Zieladresse	Quelladresse	Frame Control	Fehlercode	End of Text
00000010 <sub>b</sub>	10000000 <sub>b</sub>	10000001 <sub>b</sub>	10000000 <sub>b</sub>		00000011 <sub>b</sub>
02 <sub>h</sub>	80 <sub>h</sub>	81 <sub>h</sub>	80 <sub>h</sub>	„0“-„5“ (30 <sub>h</sub> -35 <sub>h</sub> )	03 <sub>h</sub>

Bedeutung des Fehlercodes:

Wert (ASCII Zeichen)	Bedeutung
„0“ (30 <sub>h</sub> )	kein Fehler
„1“ (31 <sub>h</sub> )	falsche Prüfsumme
„2“ (32 <sub>h</sub> )	reserviert
„3“ (33 <sub>h</sub> )	falsche Anzahl der Datenbytes (LEN-H/L), ungültige ESCAPE-Sequenz
„4“ (34 <sub>h</sub> )	Element (Text, Variable, Grafik, Zeichensatz oder Makro) nicht vorhanden, ungültiger Parameter
„5“ (35 <sub>h</sub> )	Flash ungültig

Bei Telegrammen, die Antwortdaten anfordern („Punkt aus Display auslesen“, „Digitale Ein-/Ausgänge“), wird statt des Fehlercodes „0“ die angeforderte Information zurückgeliefert.

Der Fehlercode im Antworttelegramm bezieht sich immer auf das letzte Teiltelegramm (bei mehreren ESC-Sequenzen).

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.9 Mehrfache ESC-Sequenzen

Es ist möglich, mehrere Teiltelegramme zu einem Gesamttelegramm zusammenzufassen. Dies gilt sowohl bei der normalen Ansteuerung, als auch in der Makroliste.

Die Teiltelegramme können jeweils eine ESC-Sequenz oder ein Onlinetext sein.

Soll nach einer ESC-Sequenz ein Onlinetext folgen, so ist dieser durch das „Trennzeichen“  $31_D = 1F_H$  von der ESC-Sequenz zu trennen. Das Trennzeichen selbst wird nicht ausgewertet.

Die maximal erlaubte Gesamttelegramm-Länge der Data-Unit beträgt 230 Zeichen.

### Beispiel:

Data-Unit =  $1B_H$  „Z01“  $1B_H$  „C002003“  $1B_H$  „A301“  $1F_H$  „Online-Text“

=> Es wird ein Online-Text mit Zeichensatz 1, an Cursorposition  $x=2$ ,  $y=3$  mit Vordergrundfarbe „gelb“ und Hintergrundfarbe „schwarz“ angezeigt (blinkend).

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.10 Beispiele

Online-Text an eine Großanzeige mit Adresse 1 ausgeben:

### 1. Ohne Prüfsumme, mit Antwort

STX	DA	SA	FC	Data-Unit	ETX
Start of Text	Adresse Großanzeige	Quelladresse	Frame Control	„Hallo Welt“	End of Text
00000010 <sub>b</sub>	10000001 <sub>b</sub>	10000000 <sub>b</sub>	10000001 <sub>b</sub>		00000011 <sub>b</sub>
02 <sub>h</sub>	81 <sub>h</sub>	80 <sub>h</sub>	81 <sub>h</sub>	...	03 <sub>h</sub>

Antwort der Großanzeige, falls keine Fehler vorliegen :

STX	DA	SA	FC	Data-Unit	ETX
Start of Text	Zieladresse	Quelladresse	Frame Control	Fehlercode	End of Text
00000010 <sub>b</sub>	10000000 <sub>b</sub>	10000001 <sub>b</sub>	10000000 <sub>b</sub>		00000011 <sub>b</sub>
02 <sub>h</sub>	80 <sub>h</sub>	81 <sub>h</sub>	80 <sub>h</sub>	„0“ (30 <sub>h</sub> )	03 <sub>h</sub>

### 2. Mit Prüfsumme und Antwort (in Kurzform, HEX-Format)

Auf der Großanzeige mit Adresse 1 soll ein Zeichen „A“ dargestellt werden:

Master :           02 81 80 83 F0 F1 **41** FA F6 03  
 Großanzeige:    02 80 81 80 30 03

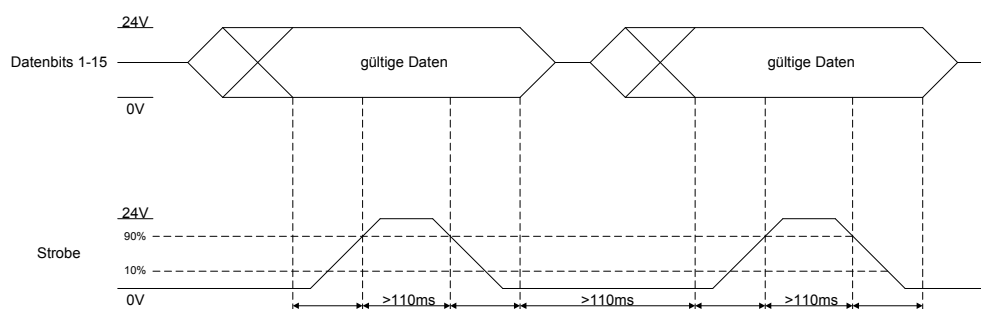
# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.11 Parallele Schnittstelle

Über die 16 Digitaleingänge können Funktionen analog den Online-Telegrammen der seriellen Schnittstelle aufgerufen werden.

### 3.11.1 Timing der parallelen Eingänge



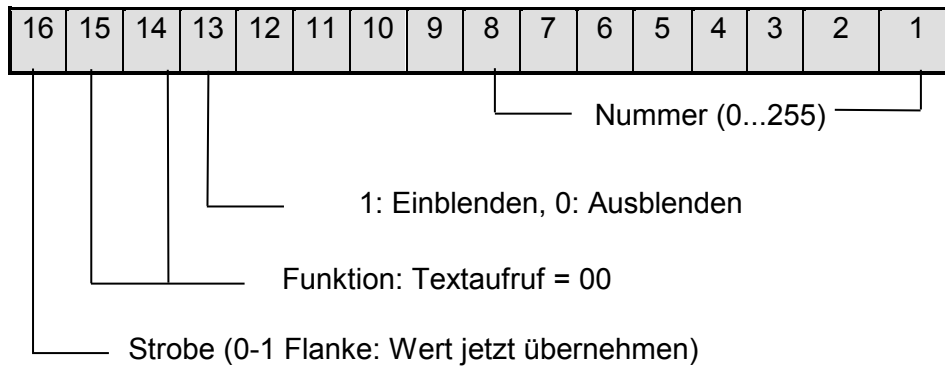
### 3.11.2 Eingangspiegel der parallelen Eingänge

Pegel	Spannungsbereich
U (low)	+ 0...1,6 VDC
U (high)	+ 18...30 VDC

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.11.3 Textaufruf



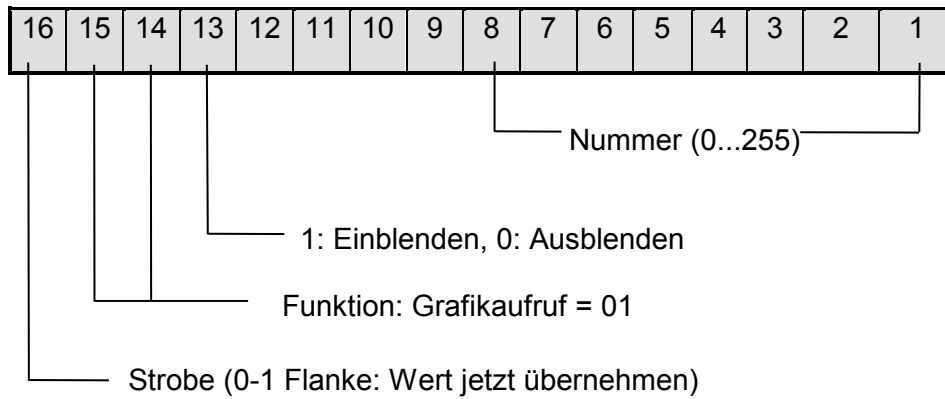
Um z.B. den Text 12 einzublenden, sollte folgende Belegung gesetzt werden:

16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
1	0	0	1	0	0	0	0	0	0	0	0	1	1	0	0

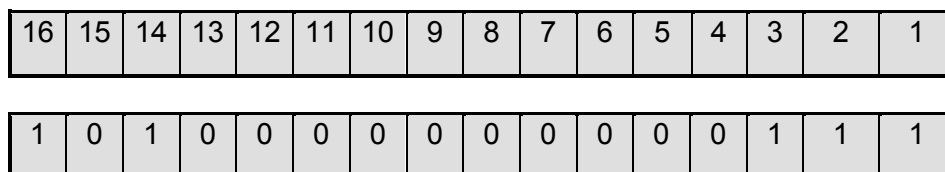
# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.11.4 Grafikaufruf



Um z.B. die Grafik 7 auszublenden, sollte folgende Belegung gesetzt werden:

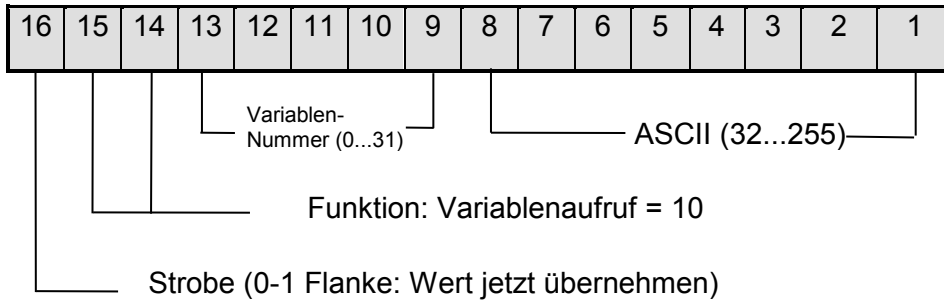


# migra MPB CAN

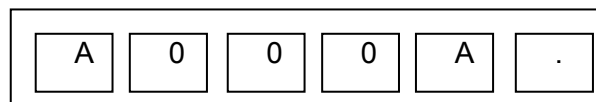
Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.11.5 Variable setzen

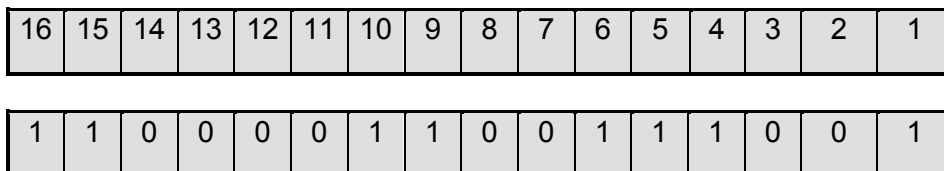
Variablen mit einem Zeichen können direkt gesetzt werden:



**Beispiel:** Bei einer mit 6 Variablen beschriebenen Anzeige soll die Variable an der vierten Stelle geändert werden.



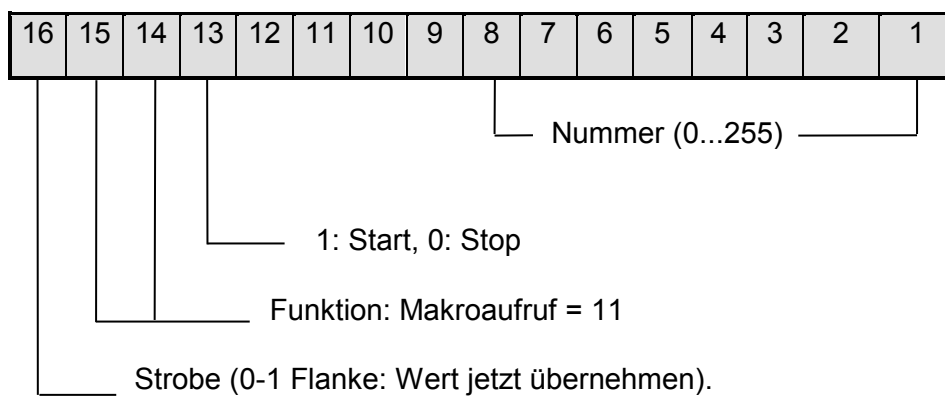
Jede Stelle wird durch eine Variable implementiert (Nummer 0...5). Um nun als vierte Ziffer eine „9“ darzustellen, muß der Variablen 3 (4. Stelle) der ASCII-Wert 39<sub>h</sub> zugewiesen werden. Dazu müssen folgende Eingänge anliegen:



# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.11.6 Makroaufruf





# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.12 CANopen-Interface

Bei der **migra SC/MC 5/3 CANopen** handelt es sich um eine erweiterte Ausführung der **migra SC/MC 5/3 Seriell**.

Die Anzeige kann sowohl über die serielle Schnittstelle, als auch über die CANopen-Schnittstelle angesteuert werden (jedoch nicht zur selben Zeit).

Hierzu wurde das „**miface CAN/RS**“-Interface in die Anzeige integriert. Dieses erlaubt die Ausgabe bzw. den Empfang von seriellen Telegrammen über die CANopen-Schnittstelle

Über das CANopen-Interface werden die selben Telegramme zur MIGRA übertragen wie über die serielle Schnittstelle.

Zur Zusammenarbeit MIGRA-Steuerplatine / CANopen-Interface sind folgende Einstellungen nötig (= Werkseinstellungen):

### CANopen-Interface:

DIP-Switches RS (S2)	1	2	3	4	5	6	7	8	9	10
Zustand (1=ON; 0=OFF)	0	0	1	1	0	0	0	0	0	0

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.12.1 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.

## 3.12.2 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>	 RS-Schnittstelle (intern fest mit der MIGRA verbunden !)
Baudrate:	19,2 kBaud
Datenbreite:	8 Bit
Parität:	gerade
Stopbits:	1
Max. Telegrammlänge:	200 Bytes in Senderichtung 200 Bytes in Empfangsrichtung
Telegrammendeerkennung:	über „Empfangs-Timeout“

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.12.3 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.

### 3.12.3.1 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.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.12.3.2 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.

## 3.12.3.3 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.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.12.3.4 SDO-Telegramme

	COB-ID	B.1	B.2	B.3	B.4	B.5	B.6	B.7	B.8
„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.

## 3.12.3.5 Receive-PDO-Telegramm

	COB-ID	B.1	B.2	B.3	B.4	B.5	B.6	B.7	B.8
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.

## 3.12.3.6 Transmit-PDO-Telegramm

	COB-ID	B.1	B.2	B.3	B.4	B.5	B.6	B.7	B.8
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.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

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

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

Byte	Funktion
1	Funktionsbyte: * Bit 7:       Endeбит * Bit 6...5:   reserviert (=0) * Bit 4:       Togglebit * Bit 3:       reserviert (=0) * Bit 2...0:   Teil-Telegramm-Länge
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.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 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.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## Beispiel zur Aussendung eines RS-Telegramms

(RS-Telegramm "ABCDEFGH12345" 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.



# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 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>n</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!

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

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

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

Byte	Funktion
1	Funktionsbyte: * Bit 7:       Endebeit * Bit 6...5:   reserviert (=0) * Bit 4:       Togglebit * Bit 3:       reserviert (=0) * Bit 2...0:   Teil-Telegramm-Länge
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

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 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

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 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

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 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).

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.12.6 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:

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

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 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:

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:

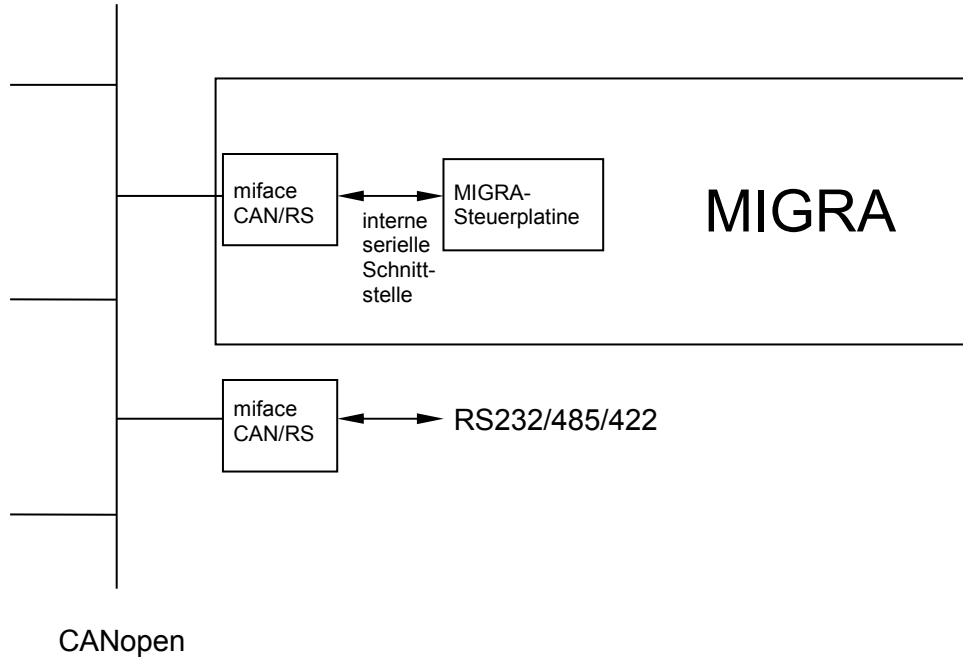
02<sub>h</sub> 80<sub>h</sub> 81<sub>h</sub> 80<sub>h</sub> 30<sub>h</sub> 03<sub>h</sub>

- \* 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)

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3. 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>) ab-  
 \* warten: 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>) ab-  
 \* warten: 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>
- \* alle Knoten starten: NMT-Telegramm (COB-ID = 000<sub>h</sub>) senden: 01<sub>h</sub> 00<sub>h</sub>



# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.12.7 Hinweis

Das Gerät muss sich bereits beim Einschalten an einem aktiven CAN-Bus befinden!

Anderenfalls entsteht beim Aussenden der Boot-Up-Message ein CAN-Fehler und es kann nicht (sofort) nach "OPERATIONAL" gewechselt werden!

Kann jedoch nicht sichergestellt werden, daß sich bereits beim Einschalten mindestens ein aktiver Teilnehmer (z.B. die SPS) am CAN-Bus befindet, wird folgende Vorgehensweise zum "Hochlauf" des CAN-Geräts empfohlen (Beispiel für Knoten-ID=1):

**a)** Senden Sie "NMT, Start remote node":

TX: ID=000, LEN=2, RTR=0, DATA=01 01

**b)** Fordern Sie das "Nodeguard-Telegramm" an:

TX: ID=701, LEN=1, RTR=1

RX: ID=701, LEN=1, RTR=0, DATA=05/85 oder 7F/FF

**c)** Falls DATA=7F/FF war (=> noch PRE-OPERATIONAL, rote ERROR-LED leuchtet):

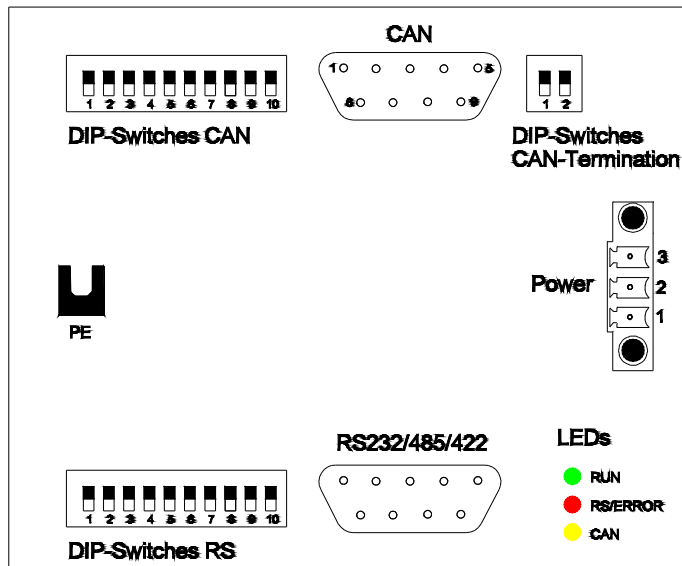
- Wiederholen Sie Schritt **b)** 40mal (=> rote ERROR-LED geht AUS)

- Führen Sie Schritt **a)** erneut aus (=> Knoten wird jetzt erfolgreich gestartet)

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 3.12.8 DIP-Schalter / LEDs



### 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

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 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 dem Einschalten) 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

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 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

\*: Einstellung für die Kommunikation mit der MIGRA (nicht verstellen !)

## Parität / Empfangs-Timeout:

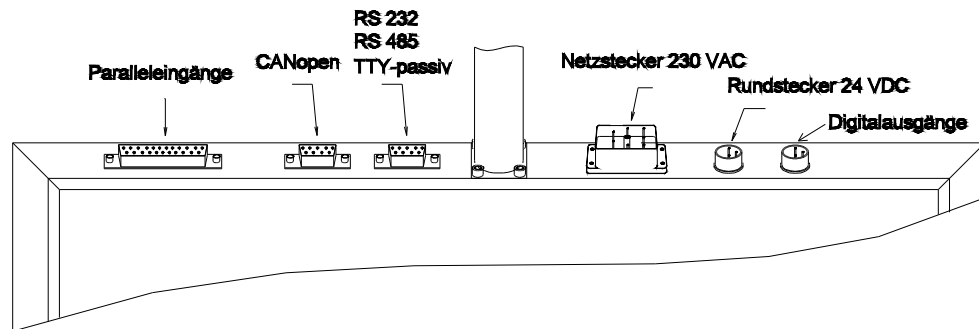
DIP-Switches RS			Parität	Empfangs-Timeout
DIP				
6	5	4		
0	0	0	keine	Standard-Timeout
0*	0*	1*	gerade	Standard-Timeout
0	1	0	ungerade	Standard-Timeout
0	1	1	keine	30 ms
1	0	0	keine	3 x Standard-Timeout
1	0	1	gerade	3 x Standard-Timeout
1	1	0	ungerade	3 x Standard-Timeout
1	1	1	keine	60 ms

\*: Einstellung für die Kommunikation mit der MIGRA (nicht verstellen !)

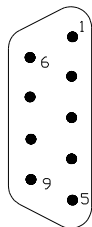
# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 4 Anschlußbelegung



### 9pol. Sub-D Stiflleiste (CANopen)



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

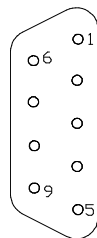
# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

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

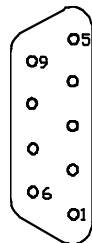
Es ist je nach RS-Schnittstelle (Bestelloption) folgende Belegung gültig:

### 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):



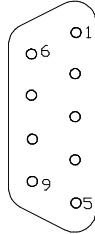
Pin	RS485
1	n.c.
2	n.c.
3	Rx+ / Tx+
4	n.c.
5	GND *
6	+5 VDC *
7	n.c.
8	Rx- / Tx-
9	n.c.

\* Falls ein externer Busabschluss benötigt wird, können diese Pins verwendet werden.

# migra MPB CAN

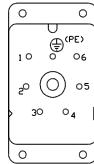
Grafikfähige LED-Großanzeige mit CANopen-Interface

## TTY-passiv (Stiftleiste):



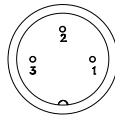
Pin	TTY
1	GND
2	n.c.
3	n.c.
4	n.c.
5	RxD +
6	TxD +
7	RxD -
8	TxD -
9	n.c.

## 7pol. Netzstecker (230 VAC)



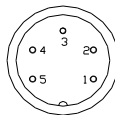
PIN	Belegung
1	L1
2	N
(PE)	PE

## 3pol. Rundstecker (24 VDC, optional)



PIN	Belegung
1	GND
2	+24 VDC
3	PE

## 5pol. Rundstecker (nur bei Option Digitalausgänge)



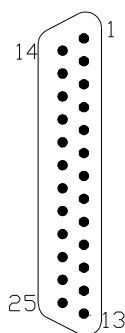
PIN	Belegung
1	Relais 1, Schließer-a
2	Relais 1, Schließer-b
3	Relais 2, Schließer-a
4	Relais 2, Schließer-b
5	n.c.

Relais 1 / 2 wird über Digitalausgang 1 / 2 angesteuert. (Telegramm ESC+“D.”)

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 25pol. Sub-D Stiftleiste (nur bei Option Paralleleingang)



Pin 1	Binär-Daten 2 <sup>0</sup>	
Pin 2	Binär-Daten 2 <sup>1</sup>	
Pin 3	Binär-Daten 2 <sup>2</sup>	
Pin 4	Binär-Daten 2 <sup>3</sup>	
Pin 5	Binär-Daten 2 <sup>4</sup>	
Pin 6	Binär-Daten 2 <sup>5</sup>	
Pin 7	Binär-Daten 2 <sup>6</sup>	
Pin 8	Binär-Daten 2 <sup>7</sup>	
Pin 9		Variablennr.: 2 <sup>0</sup>
Pin 10		Variablennr.: 2 <sup>1</sup>
Pin 11		Variablennr.: 2 <sup>2</sup>
Pin 12		Variablennr.: 2 <sup>3</sup>
Pin 13	Ein-/Ausblenden (1: Ein, 0: Aus)	Variablennr.: 2 <sup>4</sup>
Pin 14	Funktionsauswahl 2 <sup>0</sup>	
Pin 15	Funktionsauswahl 2 <sup>1</sup>	
Pin 16	Strobe (1: Wert jetzt übernehmen)	
Pin 17-24	n.c.	
Pin 25	Gnd	

- Binär-Daten:** Daten mit einem Wert von 0...255 als Textnummer, Grafiknummer, Makronummer oder als ASCII-Zeichen einer Variablen, die angezeigt werden soll.
- Ein-/ Ausblenden:** Blendet das ausgewählte Objekt ein oder aus.
- Textaufruf:** Blendet den ausgewählten Text ein/aus.
- Grafikaufruf:** Blendet die ausgewählte Grafik ein/aus.
- Variablenaufruf:** Zeigt das über die Binärdaten eingestellte ASCII-Zeichen an der Position der eingestellten Variable an.
- Makroaufruf:** Startet/stoppt die Makroausführung beim ausgewählten Makro.
- Strobe:** Durch ein „HIGH“-Signal“ an Pin 16 (>110 ms) werden die eingestellten Daten übernommen (Ausführung des Kommandos spätestens 100 ms nach der steigenden Flanke, außer wenn die Anzeige noch mit der Abarbeitung eines anderen Kommandos beschäftigt ist).

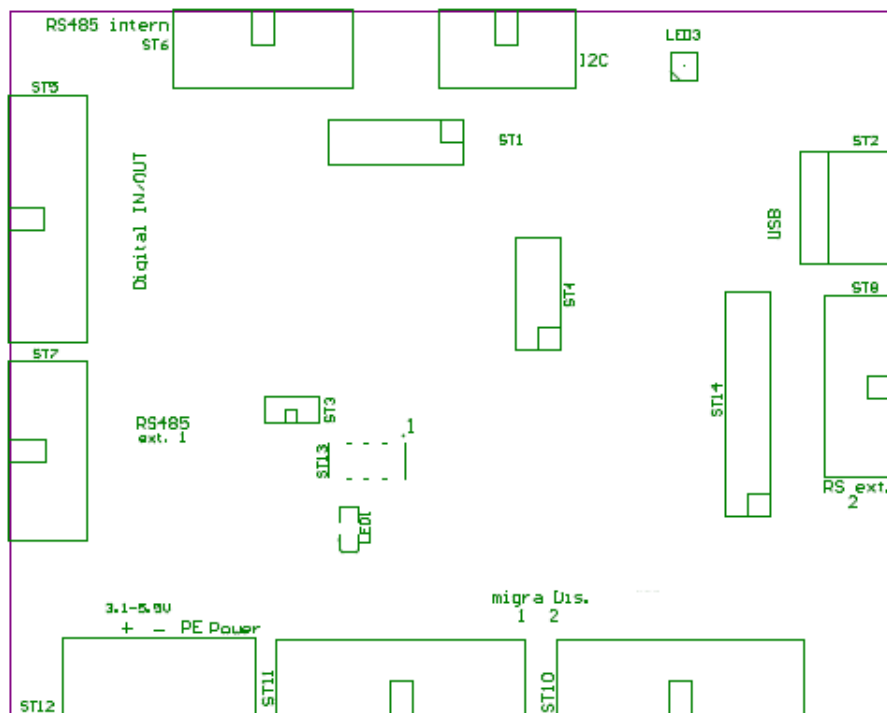
Details zu den Paralleleingängen finden sich in Kapitel „Parallele Schnittstelle“.



# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 4.1 Schnittstellenkonfiguration / LEDs



Folgende Standardschnittstellenparameter sind bei Auslieferung voreingestellt:

- \* Baudrate: 19200 Baud
- \* Datenbit: 8
- \* Parität: even
- \* Stopbit: 1

Schnittstellenparameter werden über die PC Software MKS konfiguriert.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

LED	Funktion / Beschreibung	
LED 1 (grün)	Hochlauf:	blinkt mit ca 2,5 Hz
	Normalbetrieb:	blinkt mit ca. 5 Hz
	Bootmodus:	blinkt mit ca. 0,5 Hz
	Softwareupdate:	Dauerhaftes flackern
	Konfiguration:	
	MKS fehlt:	1x blinken (Fehlercode)
LED 3 (blau)	Micon fehlt:	2x blinken (Fehlercode)
	Bildsignal erkannt:	an
	Bildsignal fehlerhaft:	aus

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 5 Anhang

### 5.1 Darstellbare Zeichen (ASCII-Tabelle)

Die Datenbytes werden ASCII-codiert.

Zeichensatz: alle ASCII-Zeichen im Bereich von 20H bis FFH.

Beispielzeichensatz:

		Higher		bin	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
Lower	bin	hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F		
	bin xxxx0000	hex 0	X	X		0	@	P	`	p	X	X	X	X	X	X	X	X	X	X
	xxxx0001	1	X	X	!	1	A	Q	a	q	ü	X	X	X	X	X	X	X	X	X
	xxxx0010	2	X	X	"	2	B	R	b	r	ß	X	X	X	X	X	X	X	X	X
	xxxx0011	3	X	X	#	3	C	S	c	s	X	X	X	X	X	X	X	X	X	X
	xxxx0100	4	X	X	\$	4	D	T	d	t	ä	ö	X	X	Ä	X	ä	X	X	X
	xxxx0101	5	X	X	%	5	E	U	e	u	X	X	X	X	X	X	X	X	X	X
	xxxx0110	6	X	X	&	6	F	V	f	v	X	X	X	X	X	Ö	X	ö	X	X
	xxxx0111	7	X	X	'	7	G	W	g	w	X	X	X	X	X	X	X	X	X	X
	xxxx1000	8	X	X	(	8	H	X	h	x	X	X	X	X	X	X	X	X	X	X
	xxxx1001	9	X	X	)	9	I	Y	i	y	X	ö	X	X	X	X	X	X	X	X
	xxxx1010	A	<CR>*	X	*	:	J	Z	j	z	X	ü	X	X	X	X	X	X	X	X
	xxxx1011	B	X	X	+	;	K	[	k	{	X	X	X	X	X	X	X	X	X	X
	xxxx1100	C	X	X	,	<	L	\	l		X	X	X	X	X	Ü	X	ü	X	X
	xxxx1101	D	<CR>*	X	-	=	M	]	m	}	X	X	X	X	X	X	X	X	X	X
	xxxx1110	E	X	X	.	>	N	^	n	~	Ä	X	X	X	X	X	X	X	X	X
	xxxx1111	F	X	X	/	?	O	_	o	■	X	X	X	X	X	ß	X	X	X	X

X bedeutet nicht verfügbar

\*Carriage Return: Cursor an den Anfang der nächsten Zeile

Es kann jeder beliebige Windows-Zeichensatz oder selbstdefinierte Zeichensatz verwendet werden.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 5.2 Allgemeine Hinweise

Beachten Sie bitte folgende Hinweise:

- Achten Sie bei der Montage der Anzeige darauf, daß auch in montiertem Zustand das Gehäuse zu Einstell- oder Wartungsarbeiten geöffnet werden kann. Lassen Sie bei der Befestigung der Anzeige an der Rückseite/Vorderseite/Oberseite einen entsprechenden Abstand frei, um eine ausreichende Belüftung zu gewährleisten (falls vorhanden).
- Direkte Bestrahlung durch helle Lichtquellen oder direkte Sonneneinstrahlung vermindern die Ablesqualität.
- Zum Reinigen muß die Anzeige ausgeschaltet sein.
- Schützen Sie die Anzeige vor übermäßiger Feuchtigkeit, starken Vibrationen, direkter Sonneneinstrahlung und extremen Temperaturen. Nichtbeachtung kann zu Funktionsstörungen oder zur Zerstörung führen. Darüber hinaus besteht unter Umständen die Gefahr von Stromschlag, Brand oder Explosion. Informationen zu den bestimmungsgemäßen Umgebungsbedingungen, insbesondere zu empfohlenen Temperaturbereichen finden Sie im Kapitel "Technische Information".
- Die Anzeige darf bei erkannter Beschädigung am Gerät und / oder der Netzleitung nicht verwendet werden.
- Versuchen Sie nicht, das Gerät selbst zu reparieren. Jeder Fremdeingriff durch unautorisierte Personen führt zum Garantieverlust.
- Beachten Sie alle Hinweise und Vorgaben, die in diesem Benutzerhandbuch enthalten sind.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 5.3 Konformitätserklärung

# EU-Konformitätserklärung

## EU Declaration of Conformity

**Produktbezeichnung:** migra  
*Product name:*

**Typenreihe:** migra CAN  
*Type code:*

**Hersteller:** microSYST Systemelectronic GmbH  
*Manufacturer:* Am Gewerbepark 11  
 92670 Windischeschenbach

<b>Das bezeichnete Produkt stimmt mit der folgenden Europäischen Richtlinie überein:</b> <i>We herewith confirm that the above mentioned product meets the requirements of the following standard:</i>		<b>Die Übereinstimmung des bezeichneten Produk- tes mit den Vorschriften der angewandten Richt- linie(n) wird nachgewiesen durch die Einhaltung folgender Normen / Vorschriften:</b> <i>The conformity of the product described above with the provisions of the applied Directive(s) is demon- strated by compliance with the following standards / regulations:</i>	
<b>Richtlinien / Directives</b>		<b>Europäische Norm / Standard</b>	
<b>EMV Richtlinie</b> <i>EMC Directive</i>	<b>2014/30/EU</b>	EN61000-6-2:2005	
		EN61000-6-4:2007 +A1:2011	
<b>Niederspannungs- Richtlinie</b> <i>Low Voltage Directive</i>	<b>2014/35/EU</b>	EN60950-1:2006 +A11:2009 +A1:2010 +A12:2011 +A2:2013	
<b>RoHS Richtlinie</b> <i>RoHS Directive</i>	<b>2011/65/EU</b>	EN50581:2012	

Windischeschenbach, 16.11.2017



Manuel Raß

**Geschäftsführer / General Manager**

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 5.4 Gewährleistung / Haftung

Für das gelieferte Produkt wird gemäß unserer Allgemeinen Geschäftsbedingungen die Haftung für bereits bei Übergabe vorhandener Mängel übernommen.

Technisch bedingte Änderungen sowie Irrtum bleiben vorbehalten. Ein Anspruch auf Lieferung eines neuen Produkts besteht nicht. Der Erwerber hat die erhaltene Ware unverzüglich zu überprüfen und offensichtliche Mängel bis spätestens 24 Stunden nach deren Wahrnehmung anzuzeigen. Bei Verletzung der Rügepflicht gilt der betreffende Mangel als genehmigt. Nicht sofort sichtbare Mängel sind ebenfalls unverzüglich nach deren Wahrnehmung anzuzeigen.

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 zur Behebung des Mangels 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 Geschäftsbedingungen der microSYST Systemelectronic GmbH in der aktuellen Fassung.

# migra MPB CAN

Grafikfähige LED-Großanzeige mit CANopen-Interface

## 5.5 Versionsübersicht

Ver.	Datum	Bemerkungen, Beschreibungen
1.00	18.04.02	Kreuzer, Nickl: Dokument angelegt
1.10	16.12.02	Kreuzer: Neues Logo
1.20	23.01.03	Kreuzer: Neue Bezeichnung Rx/Tx (+/-)
1.30	08.10.03	Kreuzer: Neue Steuerplatine
1.40	02.02.04	Kreuzer: Pinbelegung CAN geändert (ohne CAN_Shield)
1.50	31.01.05	Kreuzer: Texte, Variablen, usw. sind 0-basiert
1.60	07.03.06	Kreuzer: max. Auflösung auf 4x12 Displaymodule erweitert, mehrere ESC-Sequenzen hintereinander möglich, „Signed-Integer“-codierter Dezimalwert bei Bargraph entfällt, Trennzeichen zwischen mehreren Teiltelegrammen
1.70	16.08.10	Technische Daten aktualisiert
1.80	25.03.13	Firmenanschrift, Konformitätserklärung, Gewährleistung
1.90	17.10.13	Logo
2.00	13.03.15	SC/MC entfernt, Beschreibung Steuerplatine
2.10	29.04.16	Konformitätserklärung
3.00	24.11.16	migra → migra MPB
3.10	13.11.17	Änderung der Adresse und Titel MPB
3.20	18.04.18	Hinweis zum Geräte-Hochlauf

Zertifiziert nach **DIN EN ISO 9001**.