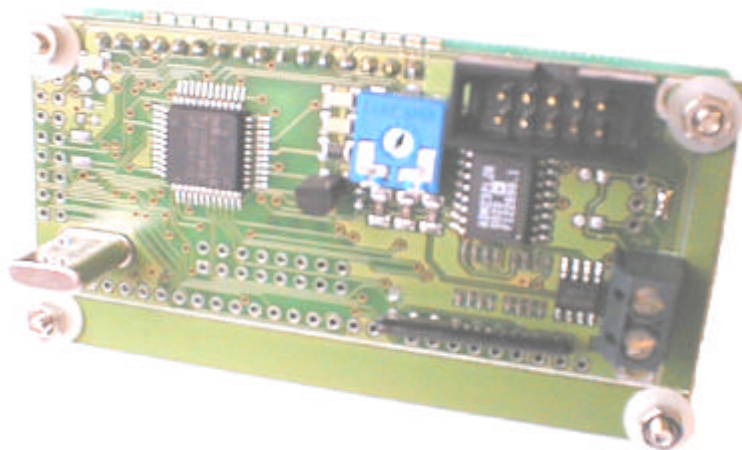


Handbuch
Mini Terminal MT
16x1 / 16x2 / 40x4



Ansteuerung mit Festtextspeicher für Character LCD-Module

Inhalt

Allgemeines	2
Inbetriebnahme des MiniTerminals	3
Steckerbelegungen	8
Spannungsversorgung	11
Befehlssatz	13
Software	18

Allgemeines

LCD-Charakter-Module sind relativ preiswert und eignen sich hervorragend für einfache Benutzerführung und Statusanzeigen. Das *MiniTerminal* vereinfacht den Einsatz dieser LCD-Charakter-Module indem der Anschluß an eine Standardschnittstelle (RS232) ermöglicht und eine Reihe zusätzliche Funktionen bereitgestellt wird.

- Ansteuerung gängiger 16x1, 16x2 und 40x4 LCD-Charakter-Module über eine serielle (RS232)-Schnittstelle.
- Standardmäßiger Festtextspeicher von 2 kByte (aufrüstbar auf 6 kByte).
- Die Festtexte können über die serielle Schnittstelle und/oder über 7 TTL-Leitungen (CN4) abgerufen werden.
- Die Ausgabe von Zeichen auf dem Display ist unabhängig von der Festtextausgabe und kann mit dieser beliebig gemischt werden (siehe Demo-Programm).
- Mechanisch kompatibel zu 16x1, 16x2 und 40x4-Displays.
- Anschluss einer Matrixtastatur mit bis zu 32 Tasten und einstellbaren Tastencodes.
- Die Versorgung erfolgt standardmäßig über **stabilisierte +5 Volt** .

Das Produkt wird ständig weiter entwickelt; Änderungen an Hard- und Software, insbesondere Lage und Belegung der Anschlüsse und Steckverbindungen, Befehlssatz sind jederzeit möglich. Vor einem Einsatz ist daher auf jeden Fall die Eignung für den vorgesehenen Verwendungszweck zu prüfen.

Beachten Sie auch unsere Homepage www.bue.de. Hier finden Sie Antworten auf häufige Fragen (FAQ) sowie ggf. Software-Updates.

Werden umfangreichere Anzeigemöglichkeiten benötigt, ist ein passendes Produkt „MiniTerminal MTGR“ auch für grafische LCD-Displays mit unterschiedlichsten Auflösungen bis hin zu VGA erhältlich.

Änderungsstand Handbuch

Datum	Änderungen
19.01.01	Für MiniTerminal Version 1.6 Neue Befehle f. Tastaturkonfiguration und Versionsabfrage, Beschriftung der Pins des Schnittstellenanschlusses CN6 zur Verdeutlichung der Zählweise
16.02.01	Belegung CN6 überarbeitet, Ergänzung Transparent-Mode
27.04.01	berichtigte Beispiele des ESC&P-Befehls
17.05.01	Berichtigung Schnittstellenparameter (Anzahl Stop-Bits) Anpassung an Platinen-Rev.1

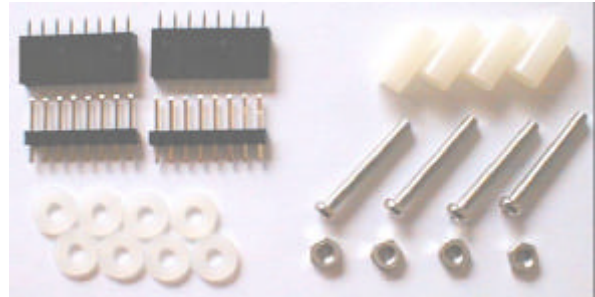
Inbetriebnahme des MiniTerminals

Die Inbetriebnahme soll anhand eines 16x2-Displays beispielhaft erklärt werden.

1. benötigtes Material

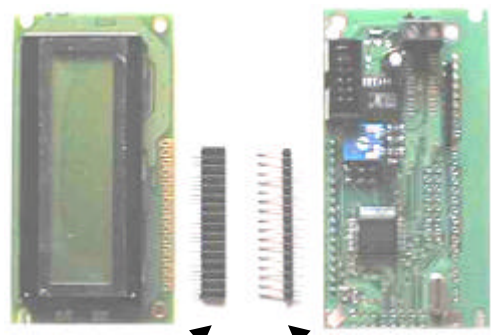
- *MiniTerminal*
- Zubehörsatz (im Lieferumfang des *MiniTerminal*)
- Bestückungsplan (siehe Seite 6)
- Display 16x2
- Lötkolben, Lötzinn
- Schraubendreher
- Spannungsversorgung, 5V stabilisiert
- Schnittstellenkabel
- PC mit TerminalProgramm, z.B. HyperTerminal oder separat erhältliche Software „MITAS“, siehe Seite 18

mitgeliefertes Zubehör *MiniTerminal*



2. Einlöten der benötigten Steckverbindungen

- Buchsenleiste in Display einlöten; nicht direkt mit *MiniTerminal* verlöten !
- je nach verwendetem Display Stiftleiste jeweils auf Lötseite einlöten :
 16x2 CN2, 16pol einreihig
 für andere Displays :
 16x1 CN1, 16pol einreihig
 40x4 CN7, 16pol zweireihig
- die Steckverbindungen passen jeweils genau wenn beide Platinen mit der Unterseite gegeneinander zusammengebaut werden.
- keine Zwischenkabel verwenden



Buchsenleiste

Stiftleiste

3. Festlegung der Spannungsversorgung

- 5V
(optional mit Spannungsregler)

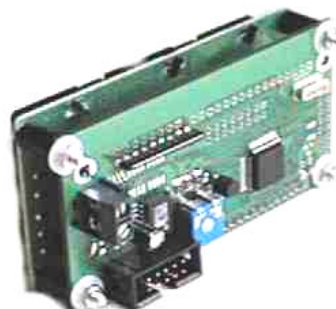
Brücke J1 **geschlossen**, Siehe Seite 11
 Brücke J1 **offen**, Siehe Seite 11)

4. Festlegung der Betriebsart

- Brücke J9 für die gewünschte Betriebsart einstellen (offen='Festtext', geschlossen= 'Bus'). Für einen ersten Test ist die Betriebsart ‚Bus‘ vorzuziehen; hier bleibt die Einschaltmeldung des *MiniTerminal* sichtbar und der Kontrast kann richtig eingestellt werden. In der Betriebsart ‚Festtext‘ dagegen wird die Einschaltmeldung nur kurz angezeigt, danach ein entsprechender Festtext, welcher aber im Lieferzustand möglicherweise noch leer ist.

5. Zusammenbau

- Je nach verwendeten Displaytyp können überstehende Ränder der Terminal-Platine an den vorgesehenen Stellen **vorsichtig** abgebrochen werden
- Montieren Sie das Display mit Hilfe der mitgelieferten Abstandshalter, Schrauben und Unterlegscheiben (diese auf der Terminalseite verwenden um Kurzschlüsse auf dem *MiniTerminal* zu vermeiden)



fertig aufgebautes Modul

6. Erste Inbetriebnahme

- Schließen Sie die stabilisierte Versorgungsspannung an die Klemme an, achten Sie auf richtige Polung und Spannung; ist kein Spannungsregler eingesetzt, sind 5V stabilisiert erforderlich
- Stellen Sie das Potentiometer P1 so ein, dass die Anzeige gut ablesbar ist. Dort sollten die Konfigurationsdaten zu lesen sein :

V<Softwareversion> **T**<Displaytyp> **B**<Baudrate>
Beispiel (Grundeinstellung): **V 1.6 T1 B0**

7. Verbindung mit dem PC

- Verbinden Sie die 10polige Buchse CN6 über das Anschlusskabel und ggf. einem weiteren Verlängerungskabel (kein Nullmodemkabel) mit einer seriellen Schnittstelle (COM1 oder COM2) des PC
- Starten Sie das Terminalprogramm (Übertragungsparameter : 9600/N/8/2, Terminalemulation „Keine“ bzw. „TTY“). Geben Sie nun einen Text ein, so wird dieser am Mini *Terminal* angezeigt.

8. Das Mini *Terminal* ist jetzt betriebsbereit

Wichtig :

- Wird ein 40x4 oder 16x1-Display eingesetzt, so muss das Mini *Terminal* zuerst entsprechend konfiguriert werden, sonst erfolgt die Anzeige nicht korrekt :
 - um ein 40x4-Display korrekt anzusteuern, ist die Befehlssequenz ‚ESC &S [12]‘ einzugeben, wobei ESC und [12] als Byte einzugeben sind (nicht die ESC-Taste und nicht die Zahl 12 eingeben sondern die Alt-Taste halten und dann auf der Zehnertastatur die Zahl 27 für ESC bzw. 12 für [12] eingeben. Auf dem Bildschirm sind diese Zeichen nicht sichtbar !)
 - Für ein 16x1-Display ist die Befehlsfolge ‚ESC &S[0]‘ (siehe auch Befehlssatz, Seite 13)
- Bei Verwendung eines Spannungsreglers siehe Seite 11
- Änderungen an der Beschaltung nur im stromlosen Zustand ausführen

Datenübertragung

Die Datenübertragung erfolgt über die RS232 Schnittstelle des MiniTerminals.
 Standardeinstellung des MiniTerminals: 9600 Baud / 8 Data-Bits / No Parity / 2 Stop-Bits
 Falls 2 Stop-Bit Senderseitig nicht einstellbar sein sollten ist die Funktion in aller Regel auch mit 1 Stop-Bit gewährleistet.
 Einstellungsmöglichkeiten: 1200, 2400, 4800, 9600 Baud (siehe unter Befehlssatz)

Die serielle Schnittstelle kann **nicht** gleichzeitig senden und empfangen. Bei Verwendung der Tastatur ist daher das RTS-Signal des MiniTerminals (bzw. CTS-Signal auf der PC-Seite) zu beachten.

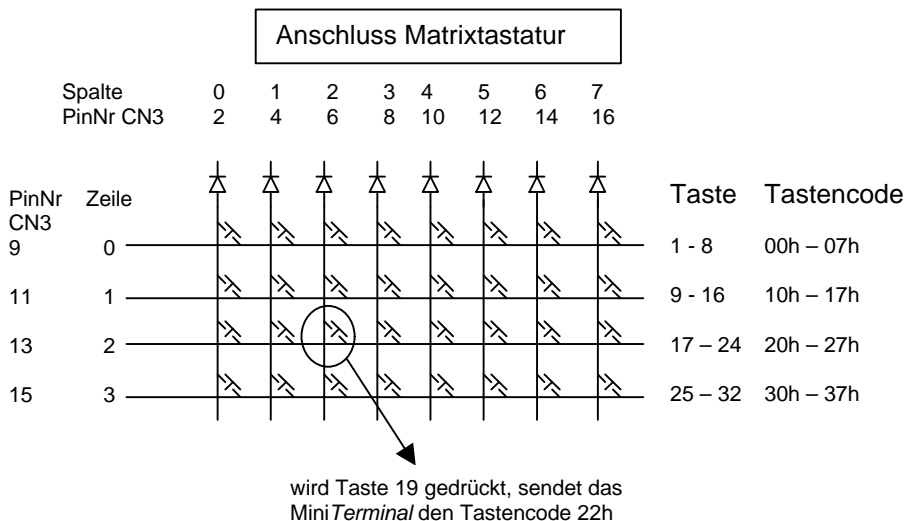
Adressierung

Es ist möglich, bis zu 16 MiniTerminals an eine serielle Schnittstelle anzuschließen. In diesem Fall dürfen die TX und CTS-Leitungen **NICHT** angeschlossen werden (die Programmierung von Festtexten ist daher vor dem Zusammenschalten vorzunehmen da dies eine angeschlossene TX-Leitung erfordert, siehe Befehlssatz „ESC R“).
 Um dennoch jedes Terminal gezielt ansprechen zu können, werden Adressen vergeben welche über CN4, Pin 1-4 eingestellt werden. In der Standardeinstellung „hört“ das MiniTerminal auf alle Adressen, es ist also keine Kodierung für CN4, Pin 1-4 nötig.

CN4	Funktion	
Pin 1	Adr0	binär kodiert, offener Anschluss ist High / 1
Pin 2	Adr1	
Pin 3	Adr2	
Pin 4	Adr3	
J9	geschlossen = Busmode offen = Auto Festtext	Standard

Tastatur

Die Matrix-Tastatur kann aus maximal 8 Spalten und 4 Zeilen (32 Tasten) bestehen. Sie wird am Stecker CN3 angeschlossen. Wird eine Taste gedrückt, sendet das MiniTerminal den Tasten-Code an den PC bzw. den Steuerrechner. Da gleichzeitiges Senden und Empfangen nicht möglich ist, wird vor dem Senden des Tastencodes die CTS-Leitung gesetzt um Kollisionen zu vermeiden.
 Bei gleichzeitigem Drücken von mehreren Tasten können die Tastatúrausgangsports des µP kurzgeschlossen werden. Dies kann Fehlfunktionen des µP verursachen.
 Aus diesem Grund sollte man an jedem Spalteneingang der Matrix-Tastatur eine Sperrdiode vorsehen (siehe Abb.)



Betriebsarten

Grundsätzlich sind 2 Betriebsarten möglich:

BUS
J9 = geschlossen

Auto Festtext
J9 = offen

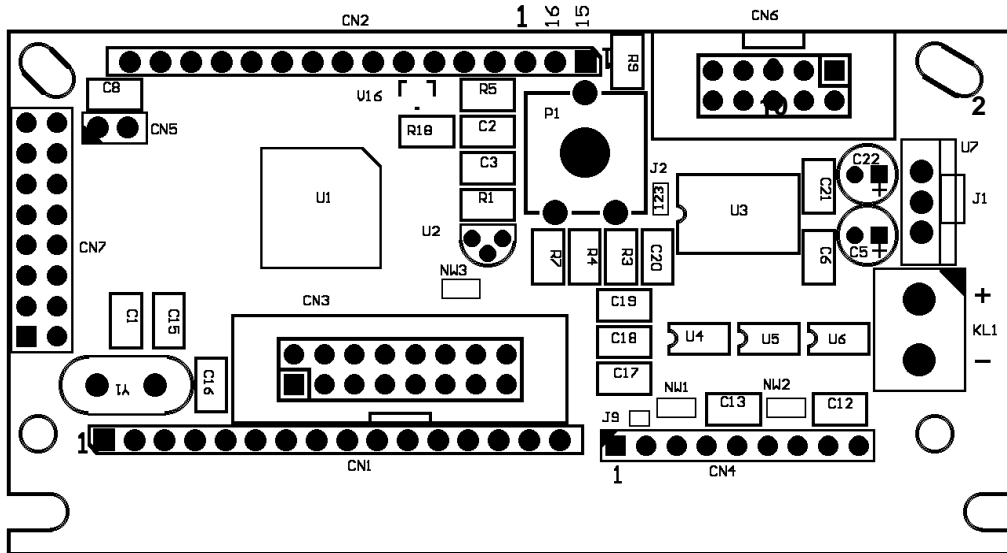
In dieser Betriebsart erfolgen sämtliche Steuerbefehle und Festtextaufrufe nur über die serielle Schnittstelle. Die gewünschte Adresse wird an CN4, Pin 1-4 eingestellt

In dieser Betriebsart können die gespeicherten Festtexte über Pins CN4, Pin 1-7 aufgerufen werden

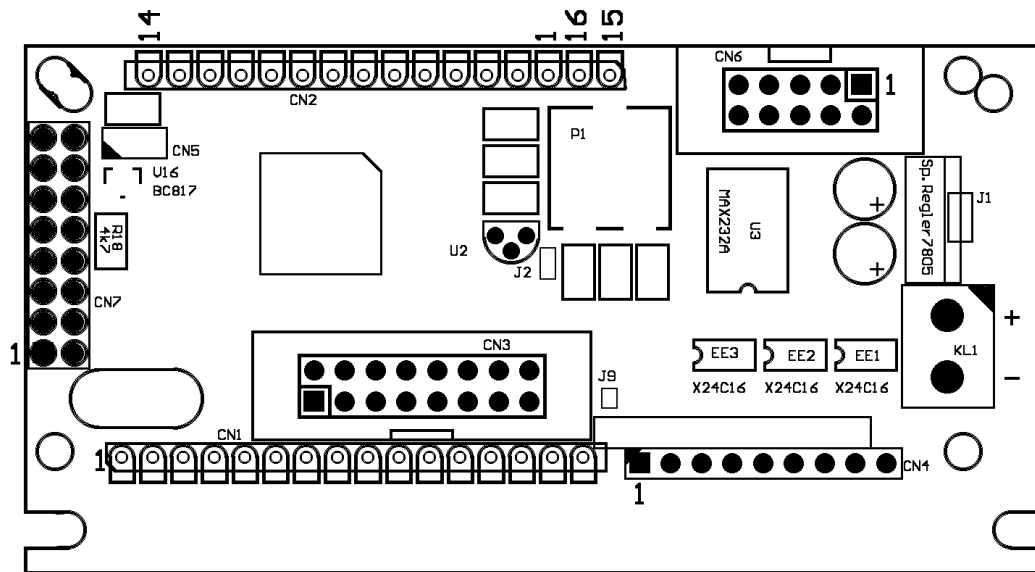
Beachten: offene Anschlüsse zählen als High / 1
ist CN4 unbelegt so ist Adresse 15 bzw Festtext 127 eingestellt

Bestückungsplan

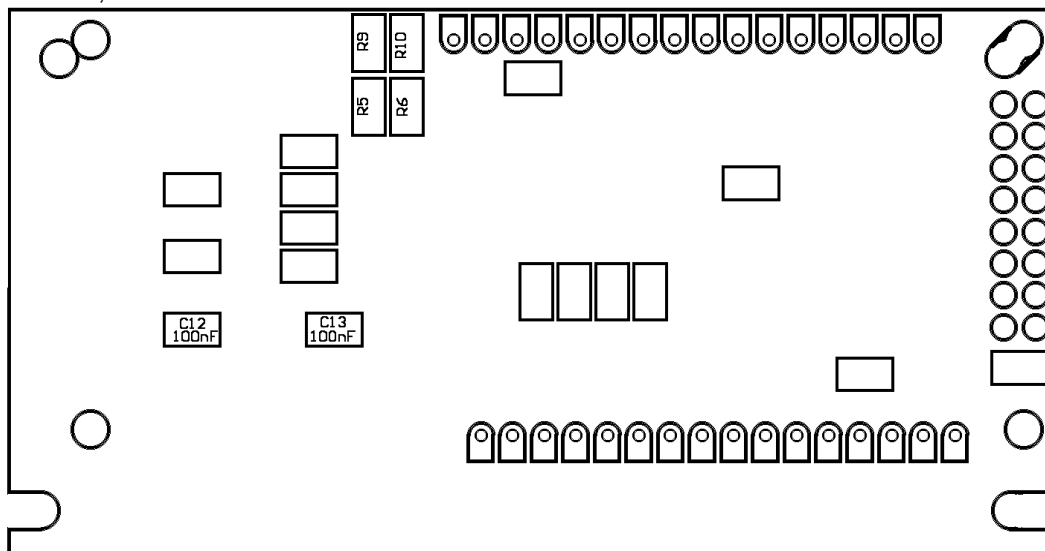
Bestückungsseite, Platinen Rev. 1



Bestückungsseite, Platinen Rev. 0



Lötseite, Platinen Rev.0



Steckerbelegungen

KL1

Spannungsversorgung

Pin	Belegung
1	GND
2	+UB J1 offen : Einsatz eines Spannungsreglers J1 geschlossen : 5V geregelt

CN1

Anschluss für LCD 16x1

Pin	Belegung	Pin	Belegung
1	GND,0V	9	DB2
2	+ 5V	10	DB3
3	VEE (Kontrastspannung)	11	DB4
4	RS (Register Select)	12	DB5
5	R/W (Read/Write)	13	DB6
6	E (Enable Signal)	14	DB7
7	DB0 (Data Bus)	15	(Backlight +UB über R9/R10)
8	DB1	16	(Backlight GND über R5/R6)

Bitte beachten: Werden Anschlusskabel für LCD's verwendet, müssen Pull-Up-Widerstände an allen Steuer- und Datenleitungen (Pin4 - Pin14) angebracht werden.

Die Widerstände R5/6 bzw. R9/10 liegen jeweils parallel und sind entsprechend dem eingesetzten Backlight einzulöten. Dabei die Verlustleistung beachten !

Auf der Platine Rev 1 ist nur R5 verfügbar.

Beispiele

Backlight +UB	Stromaufnahme	benötigter Vorwiderstand (OHM)	mögliche Bestückung					
			R5	R6	R9	R10		
4.2V	ca. 170mA	4,7	4,7	-	Brücke	-	standard	
z.B. für anderen Displaytyp mögliche Variante :								
4,2V	ca. 180mA	4,5	10	8,2	Brücke	-		
"	"	4,5	2,7	-	1,8	-		

mindestens jeweils ein Widerstand aus dem Block R5/6 bzw. R9/10 ist zu bestücken bzw. zu brücken !

CN2

Anschluss für LCD 16x2

spezielle Nummerierung beachten, bei Verwendung eines 16x2 ohne Backlight bleiben die ersten 2 Pins, also Pin15 und 16, frei. Siehe auch Bestückungsplan

Pin	Belegung	Pin	Belegung
15	Backlight +UB über R9/10	7	DB0 (Data Bus)
16	Backlight GND über R5/6	8	DB1
1	GND,0V	9	DB2
2	+ 5V	10	DB3
3	VEE (Kontrastspannung)	11	DB4
4	RS (Register Select)	12	DB5
5	R/W (Read/Write)	13	DB6
6	E (Enable Signal)	14	DB7

Bestückung der Vorwiderstände wie bei CN1

CN7

Anschluss für LCD 40x4

Pin	Belegung
1	DB7(Data Bus)
2	DB6
3	DB5
4	DB4
5	DB3
6	DB2
7	DB1
8	DB0

Pin	Belegung
9	E1 (Enable Signal)
10	R/W (Read/Write)
11	RS (Register Select)
12	VEE (Kontrastspannung)
13	GND,0V
14	+ 5V
15	E2
16	VEE-Out

Wird ein Display verwendet welches die negative Kontrastspannung selbst erzeugt, so muss ggf. J2 umbelegt werden um den Kontrastbereich ausnutzen zu können :

J2	Funktion
1-2	Kontrastspannung vom Display verwenden
2-3	Kontrastspannung vom Mini Terminal verwenden

CN3

Anschluss Matrixtastatur

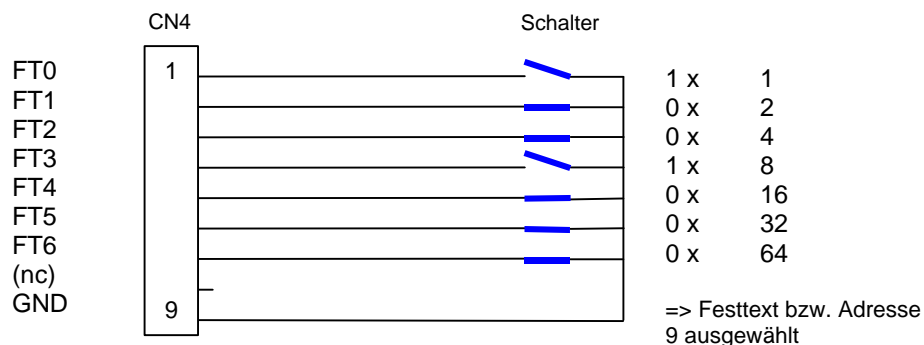
Zuordnung der Zeilen	PinNr CN3	Zuordnung der Spalten
Spalte 0 ->	2	1
Spalte 1 ->	4	3
Spalte 2 ->	6	5
Spalte 3 ->	8	7
Spalte 4 ->	10	9 <- Zeile0
Spalte 5 ->	12	11 <- Zeile1
Spalte 6 ->	14	13 <- Zeile2
Spalte 7 ->	16	15 <- Zeile3

CN4

Anschluss Festtext-Auswahl / Adressauswahl

Pin	FT-Modus J9=1	Beschreibung	Bus-Modus J9=0	Beschreibung
1	FT0	Nummer des Festtextes der ausgegeben werden soll (0 127 binär codiert) B.: für Festtext 1 müssen FT1-FT6 auf ,0' und FT0 auf ,1' oder offen) gelegt werden.	Adr0	Binäre Adresscodierung, Adresse. 0....15
2	FT1		Adr1	
3	FT2		Adr2	
4	FT3		Adr3	
5	FT4			
6	FT5			
7	FT6			
8	n.c.	(nicht belegt)		
9	GND			

Beispiel : Anschlussbeispiel für Festtextabfrage über CN4



CN5 Anschluss für Piezo-Buzzer (optional)

Pin	Belegung
1	+
2	GND

dazu sind V16 = BC817 sowie R18 = 4,7kOhm nachzubestücken

Es handelt sich dabei um einen geschalteten Ausgang (max. 150 mA) welcher durch das BEL-Steuerzeichen (07h) für ca. 30 ms aktiviert wird.

CN6 Anschluss für serielle Schnittstelle

Pin	Belegung	zugehöriges Pin Sub-D-Buchse =PC-Seite
1,2,4,7,8	(nicht belegt)	
3	TXD	2 (=RXD)
5	RXD	3 (=TXD)
6	RTS	8 (=CTS)
9	GND	5 (=GND)
10	+5V	-

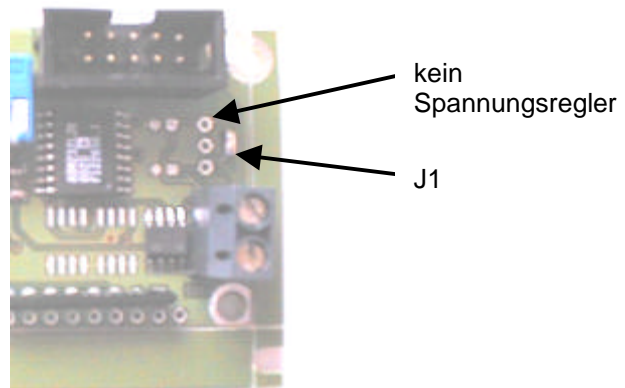


Der Stecker ist so belegt, dass zu einem 9-pol Sub-D-Schneid/Klemm-Buchse mittels eines Flachbandkabels eine 1:1-Verbindung hergestellt werden kann.

Die +5V an Pin 10 können als Ausgang zur Spannungsversorgung weiterer Komponenten verwendet werden. Die Verlustleistung eines verwendeten Spannungsreglers ist ggf. zu beachten.

Spannungsversorgung

- +5V (standard)
- J1 ist gebrückt
 - kein Spannungsregler bestückt



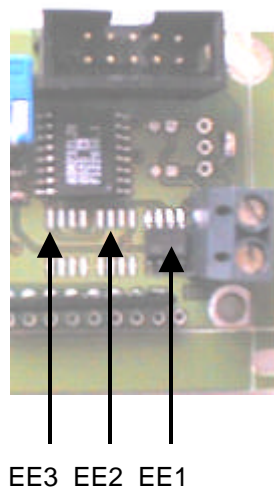
- +9-15V
- mit Spannungsregler 7805
 - J1 ist offen
- Achtung, bei Einsatz von LCDs mit Backlight unbedingt Verlustleistung des Spannungsreglers beachten !
Bei 12V Betriebsspannung und ca. 300mA Stromaufnahme ergeben sich bereits 2W Verlustleistung !

Speichererweiterung

Es können max. 2 EEPROM-Speicher (Typ 24C16) nachgerüstet werden. Dabei sind folgende Kombinationen möglich :

Speicher	EE 1	EE 2	EE 3
2kB	bestückt	(-)	(-)
4kB	bestückt	bestückt	(-)
6kB	bestückt	bestückt	bestückt

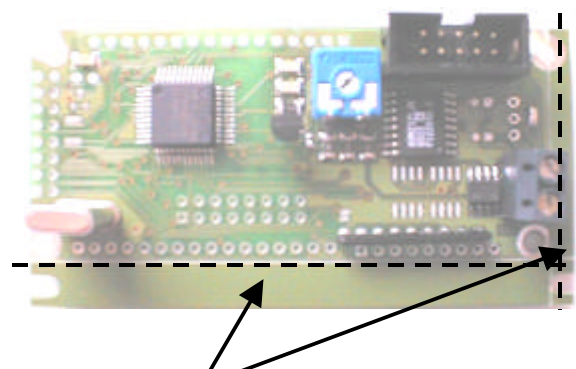
Bei Bestückung eines Speicher-IC ist auf richtige Polung zu achten sowie jeweils ein weiterer Siebkondensator C12/C13, je 100nF, nachzubestücken



EE3 EE2 EE1

Konfiguration

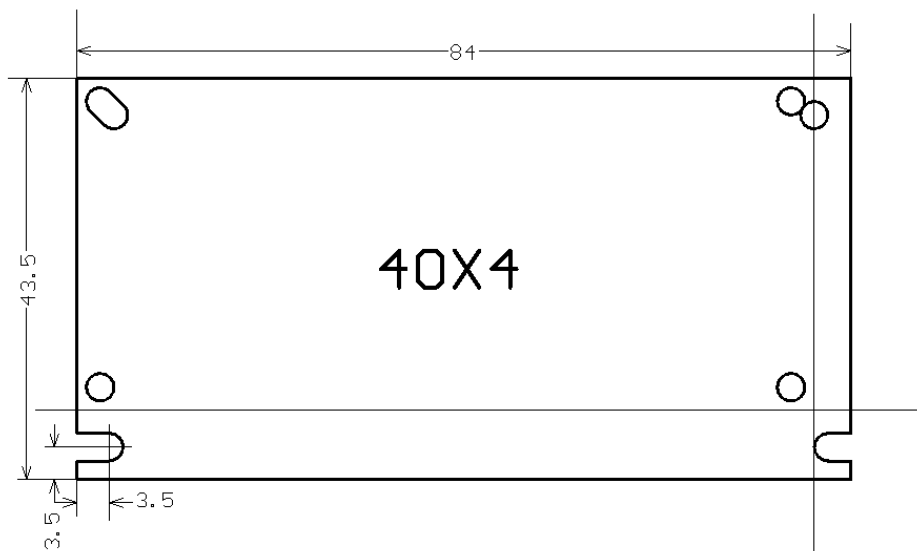
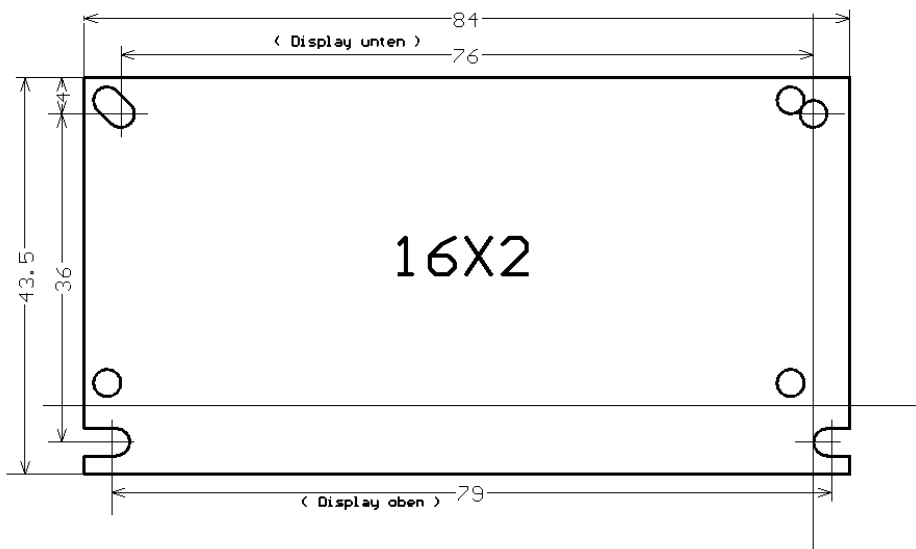
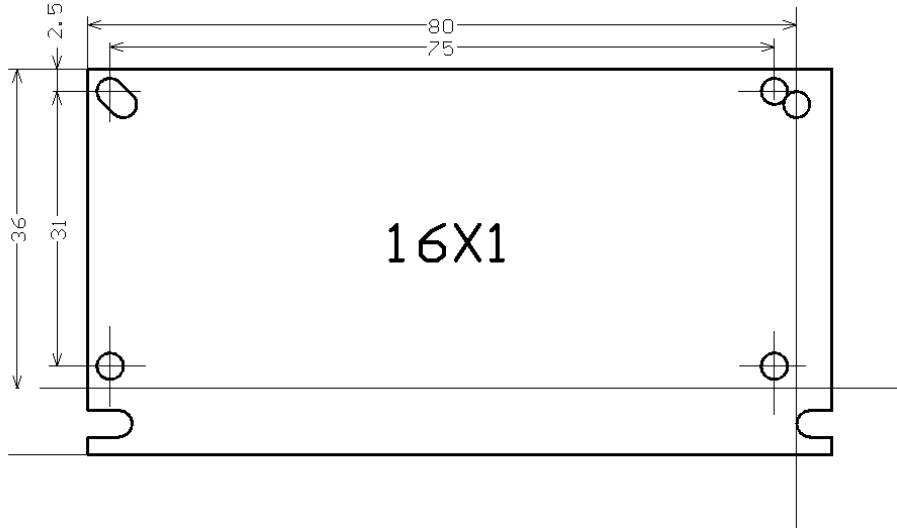
Standardmäßig ist das Miniterminal als 16x2 konfiguriert. Bei Verwendung als 16x1 oder 40x4 muss die Software umkonfiguriert werden (siehe Befehlssatz, ESC & S n)
Zur mechanischen Anpassung an die verschiedenen Displays kann seitlich ein Platinenstreifen vorsichtig (!) abgebrochen werden.



Randstreifen je nach Display ggf .entfernen

Maße

jeweils für die verwendeten Displaytypen



Befehlssatz

verwendete Abkürzungen : **CR** Wagen-Rücklauf / Neue Zeile 0d (hex) / 13 (dezimal)
ESC 1B (hex) / 27 (dezimal)

ESC [(Zeile) ; (Spalte) H | Cursor setzen

Zeile 0.. 3 / Spalte 0..39 Alle Zahlen ohne führende Nullen !

Beispiel: **PRINT #1,CHR\$(27);"[";CHR\$(2);";" ;CHR\$(1);CHR\$(9);"H";**
 setzt den Cursor in die 20. Spalte der 3. Zeile
 (Beachten: 2-stellige Spaltenzahlen auch getrennt ausgeben)

ESC & # | Display löschen / Cursor Home

Beispiel: **PRINT #1,CHR\$(27);"&#";**

ESC & C | Cursor ein

Beispiel: **PRINT #1,CHR\$(27);"&C";** schaltet den Cursor ein

ESC & D | Cursor aus

Beispiel: **PRINT #1,CHR\$(27);"&D";** schaltet den Cursor aus

ESC [K | Bis zum Ende der Zeile löschen

Beispiel: **PRINT #1,CHR\$(27);"[K";** löscht alle Zeichen bis zum Ende der aktuellen Zeile

ESC & S n | Display-Type auswählen

<i>n</i> =	0	(^@)	16 x 1 / 8 x 2
	1	(^A)	16 x 2
	2	(^B)	16 x 4
	3	(^C)	20 x 1
	4	(^D)	20 x 2
	5	(^E)	20 x 4
	6	(^F)	24 x 1
	7	(^G)	24 x 2
	8	(^H)	32 x 1
	9	(^I)	32 x 2
	10	(^J)	40 x 1
	11	(^K)	40 x 2
	12	(^L)	40 x 4

Beispiel: **PRINT #1,CHR\$(27);"&S";CHR\$(2);** Initialisiert das MiniTerminal für ein 16 x 4 Display
 Der ausgewählte Display-Typ wird im EEPROM abgespeichert

ESC & A <i>adresse</i>	Adresse vorgeben
ESC & A Z	alle selektiert (Default-Wert)

adresse = 0...15

Beispiel : **Print #1, CHR\$(27); "&A";CHR\$(1)** nur *MiniTerminal* mit zugeordneter Adresse **1** wertet Daten aus
 Print #1, CHR\$(27); "&AZ"; alle angeschlossenen *MiniTerminals* werten Daten aus (Grundeinstellung)

ESC & Bn	Einstellen der Baudrate
---------------------	-------------------------

Diese Einstellung wird im EEPROM permanent abgespeichert

n = **0** (9600), **1** (4800), **2** (2400), **3** (1200)

Beispiel: **PRINT #1,CHR\$(27);"&B";CHR\$(2);** setzt die Baudrate auf 2400 Bits pro Sekunde

Tastencode ändern

ESC & u Byte0 Byte31	Belegung des Tasten-Codes ändern
---------------------------------------	----------------------------------

Der originale Tasten-Code wird der Reihe nach durch die dem "u" folgenden 32 Byte ersetzt. Die Tasten-Codes werden, wie die Festtexte, ebenfalls im EEPROM permanent abgespeichert.

Beispiel: **PRINT #1,CHR\$(27);"&u";**
 TEXT\$ = "abcdefghijklmnopqrstuvwxy,,:;?" ' neuer Tasten-Code
 L = LEN(TEXT\$)
 FOR I=1 TO L
 PRINT #1,MID\$(TEXT\$,I,1);
 R\$ = INPUT\$(1,#1)
 IF R\$ <> MID\$(TEXT\$,I,1) THEN GOTO fehler ' Fehler beim Speichern
 NEXT I
 fehler:
 PRINT "Fehler beim Speichern"
 ende:
 END

Grundeinstellung

Taste	Tastencode (hexadezimal)
1 - 8	00h – 07h
9 - 16	10h – 17h
17 - 24	20h – 27h
25 – 32	30h – 37h

zu Beachten :

Die Einstellungen der Baudrate und der Umkodiertabelle bleiben auch bei Spannungsausfall gespeichert. Diese Einstellungen werden nur durch die o.g. Escape-Sequenzen geändert.

Tastaturkonfiguration

ESC & P n	Tastaturbetriebsart einstellen
----------------------	---------------------------------------

Dieser Befehl legt die Betriebsart der Tastatur fest :

- n=0 Polling-Modus, d.h. der Code der zuletzt gedrückten Taste wird nur auf Anfrage gesendet, und zwar im Anschluss an die Übermittlung dieses Befehls. Wurde seit der letzten Abfrage keine Taste betätigt, liefert das MiniTerminal den Code 255 / FF
- n=1 bei Tastendruck wird der Tastencode einmal unaufgefordert gesendet, auch wenn die Taste gedrückt gehalten wird
- n=2 ..7 Autorepeat-Modus, es wird solange der Tastencode gesendet, bis die Taste losgelassen wird. Dabei sind unterschiedliche Geschwindigkeiten möglich, abhängig von n :
2 : sehr schnell bis 7 : ca. 2 Sekunden-Takt

nach dem Einschalten ist Autorepeat, Stufe 2 aktiv

Beispiel: **PRINT #1,CHR\$(27);"&P0";** Tastatur in Pollingbetrieb und letzten Tastencode abfragen
PRINT #1,CHR\$(27);"&P5"; Tastatur in Autorepeat-Modus schalten mit ca. 1 Sekunde Abstand zwischen zwei gesendeten Tastencodes

Zeichen umdefinieren

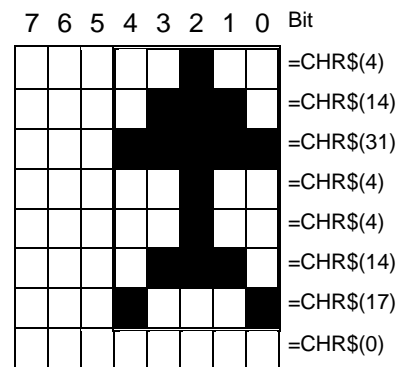
ESC & L CG Zeichen 8x Muster	bestehendes Zeichen umdefinieren
---	---

- CG Zu verwendende Zeichen-Adresse von 1-8.
Es stehen 8 Adressen für benutzerdefinierte Zeichen zur Verfügung.
- Zeichen Zeichen, welches umdefiniert werden soll. Hier kann ein beliebiges Zeichen des Character-Fonts angegeben werden
- 8x Muster Hier muss das Bit-Muster des neuen Zeichens angegeben werden. Ein Zeichen besteht aus einer 5 x 7 Punkt-Matrix. Nur Bit0 - Bit4 der 7 anzugebenden Bytes sind relevant.
Als 8. Byte wird immer 'CHR\$(0)' angegeben.

Beispiel-Programm (Qbasic) :

```

^Schnittstelle definieren
OPEN "COM1:9600,N,8,2,cs0,ds0,cd0,rs" AS 1
'Zeichen-Muster
pattern$ = CHR$(4)+CHR$(14)+CHR$(31)+CHR$(4)+CHR$(4)+CHR$(14)+CHR$(17)+CHR$(0)
INPUT "Umzudefinierendes Zeichen :"; zch$
INPUT "Zu verwendende CG-RAM-Adresse (1-8) :"; CGAdr%
'ESC-Sequenz zur Zeichenumdefinierung
PRINT #1, CHR$(27) + "&L";
PRINT #1, CHR$(CGAdr% - 1);
strg$ = zch$ + pattern$
l = LEN(strg$)
IF l <> 0 THEN
  FOR n = 1 TO l
    PRINT #1, MID$(strg$, n, 1);
    R$ = INPUT$(1, #1)
    IF R$ <> MID$(strg$, n, 1) THEN
      PRINT "Fehler beim Speichern": BEEP
    EXIT FOR
  END IF
NEXT n
END IF
  
```



Beispiel

Löschen von umdefinierten Zeichen:

Für umzudefinierendes Zeichen (zch\$) einen Wert zwischen CHR\$(10) - CHR\$(18) eingeben (nicht darstellbares Zeichen) und zu löschende CG-RAM-Adresse (CGAdr%) angeben.

Festtexte

- Die Festtexte sind mit 0 .. 127 durchnummeriert.
- Vor der Ausgabe eines Festtextes wird das Display gelöscht.
- Ein Festtext darf das Sonderzeichen CR enthalten.

Festtexte auslesen :

ESC W	Festtexte über die serielle Schnittstelle ausgeben
--------------	--

Das MiniTerminal gibt alle gespeicherten Festtexte der Reihe nach über die serielle Schnittstelle aus. Jeder Festtext wird durch 'CHR\$(0)' abgeschlossen. Die Übertragung wird durch 2x 'CHR\$(0)' abgeschlossen (siehe Speicherformat unter 'Festtext laden').

Beispiel: **PRINT #1,CHR\$(27);"W";**

Festtext laden :

ESC R	Festtexte im EEPROM abspeichern
--------------	---------------------------------

Das MiniTerminal schickt jedes Zeichen nach dem Programmieren zurück. Wurde beim Programmieren ein Fehler festgestellt (z.B. EEPROM defekt oder Speicherplatz erschöpft), wird das Zeichen invertiert. Da gleichzeitiges Senden und Empfangen nicht möglich ist, muss unbedingt das vom MiniTerminal gesendete Zeichen abgewartet werden bevor das nächste gesendet wird ! (Fehlerüberprüfung siehe Demo-Programm Zeile 230-250.)

Das Speichern der Texte dauert ungefähr 1 Sec pro 100 Zeichen.

Beispiel: PRINT #1,CHR\$(27);"R"; PRINT #1,"T"; INPUT\$(1,#1) PRINT #1,CHR\$(0); PRINT #1,"K"; INPUT\$(1,#1) PRINT #1,CHR\$(0); PRINT #1,CHR\$(0);	1. Zeichen("T") des 1. Festtextes warten, bis das Zeichen zurückgeschickt wird Ende des 1. Festtextes 1. Zeichen("K") des 2. Festtextes warten, bis das Zeichen zurückgeschickt wird Ende des n. Festtextes Ende der Übertragung
---	--

Achtung: Alle im EEPROM gespeicherten Festtexte werden bei der Übertragung überschrieben.

Geeignete Vorgehensweise bei Abänderung oder Erweiterung bestehender Festtexte:

1. Bestehende Festtexte mit Befehl 'ESC W' auslesen und in Datei abspeichern (pro Zeile 1 Festtext)
2. Festtexte in Datei mit ASCII-Editor bearbeiten (Speicherformat beachten)
3. Festtexte aus Datei zeilenweise einlesen und mit Befehl 'ESC R' im EEPROM abspeichern

-> Das im Kapitel Software erwähnte MiniTerminal-Testprogramm beinhaltet diese Funktionen.

Speicherformat : 'FT0'+Chr\$(0)+ 'FT1'+Chr\$(0)+ 'FT2'+Chr\$(0) 'FTn'+Chr\$(0)+Chr\$(0)
(im EEPROM)

Festtext ausgeben :

ESC	D	<i>nr</i>	Festtext mit Nummer ' <i>nr</i> ' anzeigen (' <i>nr</i> ' von 0 .. 127)
------------	----------	-----------	---

Dieser Befehl überschreibt die Einstellung am Stecker CN4.

Ändert sich der Wert am Stecker CN4, wird wieder der Festtext, der dem Wert am Stecker CN4 zugeordnet ist, angezeigt. Das bedeutet eine Festtextnummer wird immer dann angewählt, wenn entweder über die serielle Schnittstelle eine Nummer vorgegeben wird oder sich am Stecker CN4 die Einstellung ändert (FT-Modus).

Beispiel: **PRINT #1,CHR\$(27);"D";CHR\$(2);** Das MiniTerminal löscht das Display und zeigt den 3. Festtext an.

Hinweis: Die Festtext-Befehle können unabhängig vom Wert des Pins '**FT/Bus**' verwendet werden. D.h. über die serielle Schnittstelle können in **beiden** Betriebsarten ('Auto Festtext' oder 'Bus') Festtexte angezeigt werden.

Transparent Mode

ESC & T	in Transparent-Mode wechseln
--------------------	------------------------------

Im Transparent-Mode hat der Host direkten Zugriff auf den Controller-Chip des Displays (HD44780 oder ähnlich). Dieser Modus sollte nur von Kennern des Controllers angewendet werden !

x0 data	Write CMD (Adresse 0)
x1 data	Write Data (Adresse 1)
x2	Read CMD
x3	Read Data
x4 n	Display-Controller wählen
sonst	Transparent-Mode verlassen

data Zeichen in Hex
n Nr des auszuwählenden Display-Controllers in Hex (0=standard, 1=zweiter Controller)
x don't care (die oberen 4 Bits sind ohne Bedeutung)

Beispiel: **PRINT #1,CHR\$(27);"&T";** Enter Transparent Mode
 PRINT #1,CHR\$(0);CHR\$(2); 02h in Controller-Adresse 0 schreiben
 PRINT #1,CHR\$(255); Transparent-Mode verlassen

Konfiguration abfragen

ESC & V	Konfiguration auslesen
--------------------	------------------------

Die Konfiguration wird über die serielle Schnittstelle ausgegeben; Das Format entspricht der Einschaltmeldung und ist 10 Byte lang :

V<Softwareversion><Leerzeichen> **T**<Displaytyp><Leerzeichen> **B**<Baudrate>
Beispiel (Grundeinstellung): **V1.6 T1 B0**

Zu Bestimmung von Displaytyp und Baudrate siehe jeweils Befehle zum Setzen dieser Einstellungen. Vor dem Senden weiterer Befehle ist das Ende der Übertragung abzuwarten!

Software

(A) Kostenloses QBASIC-Programm

Systemvoraussetzungen:

- RS 232 Schnittstelle (COM1 oder Com2)
- EGA oder VGA-Grafikfähig
- DOS 5.0 oder höher
- Internetzugang: Das Programm ist ausschließlich über die Homepage www.bue.de zu beziehen.

Dieses Programm ist ein im Vergleich zum abgedruckten Demoprogramm stark erweitertes Testprogramm in QBasic-Quelltext. Es bietet einem Anwender die einfache Möglichkeit den Funktionsumfang des *MiniTerminal*-Controllers kennenzulernen und gleich auch anzuwenden.

(B) MITAS

Die *MiniTerminal*AnsteuerSoftware zur einfachen Konfiguration des *MiniTerminals*.

Dieses Programm bietet dem Anwender die einfache Möglichkeit den Funktionsumfang des *MiniTerminals* kennenzulernen und anzuwenden. Es bietet vor allem die Möglichkeit, Festtexte abzuspeichern, zu erweitern oder abzuändern sowie den Zeichensatz zu konfigurieren.

Das Programm ist durch die Menübedienung selbsterklärend.

Systemvoraussetzungen:

- Win 3.xx
- Win 95 oder höher
- Win NT 4.0 oder höher

Demo Programm für Q-Basic

```

10 ' Demo-Programm für die Festtextanzeige des MiniTerminals
20 '
30 OPEN "COM1:9600,N,8,2,cs0,ds0,cd0,rs" AS 1
40 PRINT
50 PRINT "1 - Festtexte in das MiniTerminal laden"
60 PRINT "2 - Festtexte auf dem PC anzeigen"
70 PRINT "3 - Festtexte auf dem MiniTerminal anzeigen"
80 PRINT "4 - Text auf dem Miniterminal anzeigen"
90 PRINT "5 - Ende "
100 INPUT CMD
110 ON CMD GOSUB 140,350,490,560,670
120 GOTO 40
130 '-----
140 PRINT "Festtexte in das MiniTerminal laden"
150 PRINT "Ein Festtext wird mit 'ENTER' abgeschlossen"
160 PRINT "Enthält ein Festtext keine Zeichen mehr, "
170 PRINT "wird die Übertragung beendet"
180 PRINT #1,CHR$(27);"R"; : ' Kommando an MiniTerminal schicken
190 INPUT "Text eingeben: ";TEXT$
200 L = LEN(TEXT$)
210 IF L=0 THEN GOTO 310 : ' Übertragung beenden
220 FOR I=1 TO L
230 PRINT #1,MID$(TEXT$,I,1);
240 R$ = INPUT$(1,#1)
250 IF R$ <> MID$(TEXT$,I,1) THEN RETURN : ' Fehler beim Speichern
260 NEXT I
270 PRINT #1,CHR$(0); : ' Text-Ende-Zeichen
280 R$ = INPUT$(1,#1) : ' 0 lesen
290 GOTO 190
300 '
310 PRINT #1,CHR$(0); : ' Text-Ende-Zeichen
320 R$ = INPUT$(1,#1) : ' 0 lesen
330 RETURN
340 '-----
350 PRINT "Im MiniTerminal gespeicherte Texte auf dem PC anzeigen"
360 PRINT
370 PRINT #1,CHR$(27);"W"; : ' Kommando an MiniTerminal schicken
380 NULLFLAG=0
390 A$ = INPUT$(1,#1)
400 IF ASC(A$)=0 GOTO 440
410 NULLFLAG=0
420 PRINT A$;
430 GOTO 390
440 IF NULLFLAG=1 THEN RETURN ' 2. Null --> Ende der Übertragung
450 NULLFLAG=1
460 PRINT
470 GOTO 390
480 '-----
490 PRINT "Fest-Text auf dem MiniTerminal anzeigen"
500 PRINT "Nummer (999 = Ende): ";
510 INPUT NUMMER
520 IF NUMMER=999 THEN RETURN
530 PRINT #1,CHR$(27);"D";CHR$(NUMMER);
540 GOTO 500
550 '-----
560 PRINT "Texte auf dem MiniTerminal ausgeben"
570 PRINT "Der Text wird nach der Eingabe von 'ENTER' ausgegeben"
580 PRINT "Enthält ein Text keine Zeichen mehr, wird die Funktion beendet"
590 INPUT "Text eingeben: ";TEXT$
600 L = LEN(TEXT$)
610 IF L=0 THEN RETURN : ' Übertragung beenden
620 FOR I=1 TO L
630 PRINT #1,MID$(TEXT$,I,1);
640 NEXT I
650 GOTO 590
670 END : ' Programm-Ende

```

Character-Font-Tabelle : (darstellbare Zeichen im Hex-Code)

		Higher 4 bit (D4 to D7) of Character Code (Hexadecimal)																			
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F				
Lower 4 bit (D0 to D3) Character Code (Hexadecimal)	0	CG RAM (1)			0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
	1	CG RAM (2)			!	1	!@	Q	a	q				u	7	7	4	ä	q		
	2	CG RAM (3)			"	2	B	R	b	r				Γ	イ	ウ	×	ρ	θ		
	3	CG RAM (4)			#	3	C	S	c	s				┘	ウ	テ	E	ε	ω		
	4	CG RAM (5)			\$	4	D	T	d	t				\	I	t	†	μ	Ω		
	5	CG RAM (6)			%	5	E	U	e	u				•	才	ナ	1	ε	Ü		
	6	CG RAM (7)			&	6	F	V	f	v				ヲ	カ	ニ	ε	ρ	Σ		
	7	CG RAM (8)			'	7	G	W	w				フ	キ	ヌ	ウ	g	π			
	8	CG RAM (1)			(8	H	X	h	x				イ	ウ	ホ	リ	フ	又		
	9	CG RAM (2))	9	I	Y	i	y				ウ	ケ	ル	レ	ウ			
	A	CG RAM (3)			*	*	J	Z	j	z				エ	コ	ル	レ	j	キ		
	B	CG RAM (4)			+	+	K	C	k	c				オ	サ	ヒ	ロ	*	ク		
	C	CG RAM (5)			,	<	L	¥	l	¥				ホ	シ	フ	ワ	φ	円		
	D	CG RAM (6)			-	=	M]m)				ユ	ズ	レ	シ	ト	÷			
	E	CG RAM (7)			.	>	N	^n	+				ヨ	セ	ホ	レ	ニ				
	F	CG RAM (8)			/	?	O	_o	+				ウ	リ	マ	°	ö				