

CAN-DISP

Dok-Rev. 1.5 vom 17.06.2008
Hardware-Rev. 1.0 vom 06.08.2007

Inhaltsverzeichnis

1	Allgemeine Hinweise.....	4
1.1	Handhabung	4
1.2	Installation	4
1.3	Erklärung	4
1.4	Reparaturen	4
2	Technische Daten.....	5
2.1	Umgebungsbedingungen	5
2.2	Mechanische Abmessungen	5
2.3	Technische Daten	5
3	Inbetriebnahme.....	6
3.1	Einbau	6
3.2	Spannungsversorgung	6
3.3	Lage der Jumper und Steckverbinder	6
3.4	Beschreibung der Jumper	7
3.4.1	ST1 – Baudrate der seriellen Schnittstelle	7
3.4.2	ST4, ST5 – Reserviert	7
3.4.3	ST14, ST15 – Pegel der seriellen Schnittstelle	7
3.5	TR1 – Kontrastregelung	7
3.6	SW1 – Einstellen der Kommunikationsparameter	7
3.6.1	Serielle Parameter	7
3.6.2	CAN-Parameter	8
3.7	Betriebszustand	9
4	Hardwarebeschreibung.....	10
4.1	ST2 – Serielle Debug-Schnittstelle	10
4.2	Steckverbinder	10
4.2.1	Display	10
4.2.2	serielle Schnittstelle	11
4.2.3	Matrixtastatur	12
4.2.4	ST19 – Touchscreen	12
4.2.5	ST6 – CAN-Bus	12
4.2.6	ST25 – LCD-Beleuchtung	13
4.2.7	ST20 – Lautsprecher	13
4.2.8	ST18 – 24 Volt Versorgung	13
5	Kommunikation	14
5.1	Serielle Schnittstelle	14

5.2 CAN-Bus	14
5.2.1 Basis-ID (CANDISP_OUTPUT)	14
5.2.2 Basis-ID + 1 (CANDISP_FLOW_OUT)	15
5.2.3 Basis-ID + 2 (CANDISP_INPUT)	15
5.2.4 Basis-ID + 3 (CANDISP_FLOW_IN)	15
6 Befehlssatz	16
6.1 Befehlsliste numerisch	16
6.2 Befehlsliste alphabetisch	17
6.3 Befehlsbeschreibung	17
6.4 Zeichensätze	26
6.5 Scancodes	26
6.6 Touchscreen-Tasten	28

Revisionsliste:

Rev.	Datum	Na.	Änderung
1.0	30.08.2007	Ko	Erstellung
1.1	03.09.2007	Ko	div. Änderungen
1.2	12.02.2008	Ha	Softwarebeschreibung
1.3	18.02.2008	Ha	Beschreibung box_c korrigiert, rbox_c und CAN-Protokoll ergänzt
1.4	21.02.2008	Ha	Beschreibung rbox_c korrigiert
1.5	18.06.2008	Ha	SW1 erweiterte Funktion

1 Allgemeine Hinweise

1.1 Handhabung

1. Lesen Sie bitte zuerst sorgfältig diese Dokumentation bevor Sie die Hardware auspacken und einschalten. Sie sparen Zeit und vermeiden Probleme.
2. Beachten Sie bitte die Vorsichtsmaßnahmen bei der Handhabung elektrostatisch gefährdeter Hardware.
3. Wenn die Hardware Batterien enthält, legen Sie sie nicht auf elektrisch leitfähige Unterlagen. Die Batterie könnte kurzgeschlossen werden und Schäden verursachen.
4. Achten Sie bitte darauf, daß der spezifizierte Temperaturbereich nicht verlassen wird.

1.2 Installation

1. Überprüfen Sie, ob alle Jumper entsprechend Ihrer Anwendung gesetzt sind.
2. Schalten Sie die Spannungsversorgung der externen Anschlüsse ab, bevor Sie eine Verbindung herstellen.
3. Wenn Sie sicher sind, daß alle Verbindungen korrekt installiert sind, schalten Sie die Spannungsversorgung ein.

1.3 Erklärung

Wir behalten uns das Recht vor, Änderungen, die einer Verbesserung der Schaltung oder des Produktes dienen, ohne besondere Hinweise vorzunehmen. Trotz sorgfältiger Kontrolle kann für die Richtigkeit der hier gegebenen Daten, Schaltpläne, Programme und Beschreibungen keine Haftung übernommen werden. Die Eignung des Produktes für einen bestimmten Einsatzzweck wird nicht zugesichert.

1.4 Reparaturen

Sollte das Produkt defekt sein, so senden Sie es bitte frei in geeigneter Verpackung mit folgender Beschreibung an uns zurück:

- Fehlerbeschreibung
- Trat der Fehler nur unter bestimmten Bedingungen auf?
- Was war angeschlossen?
- Wie sahen die angeschlossenen Signale aus?
- Garantiereparatur oder nicht?

2 Technische Daten

2.1 Umgebungsbedingungen

Umgebungstemperatur (Betrieb)	0-50° C
Umgebungstemperatur (Lagerung)	-20-85° C
rel. Luftfeuchte	max. 95%, nicht kondensierend
Höhe	-300m bis +3000m

2.2 Mechanische Abmessungen

Platinengröße	100 x 160 mm
---------------	--------------

2.3 Technische Daten

Versorgungsspannung:	5 Volt DC, 0.2 A alternativ 24 Volt DC, 0.1 A ohne Display, 5V/0.5 A mit Backlight
Prozessor	MB90F347
Displayanschluß:	für Hitachi LMG7420 / SP14N001 und weitere Displays mit Toshiba T6963C CCFL- und LED-Beleuchtung unterstützt
Tastaturanschluß:	Matrixtastatur mit bis zu 8x8 Tasten
Touchscreen:	4-Draht Anschluß X-X/Y-Y für resistive Touchscreens
serielle Schnittstelle:	RS-232-/TTL-Pegel, 9600/19200 Baud
CAN	1x CAN, Baudrate 500 kBit

3 Inbetriebnahme

3.1 Einbau

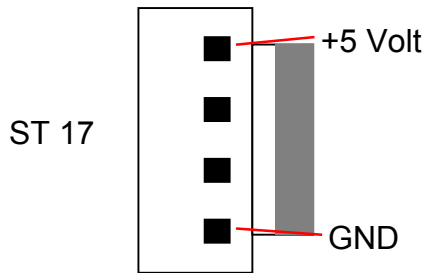
Die CAN-DISP ist zum Einbau in Schaltschränke oder ähnliche EMV-dichte Gehäuse bestimmt. Die Verkabelung ist EMV-gerecht mit abgeschirmten Kabeln durchzuführen.

3.2 Spannungsversorgung

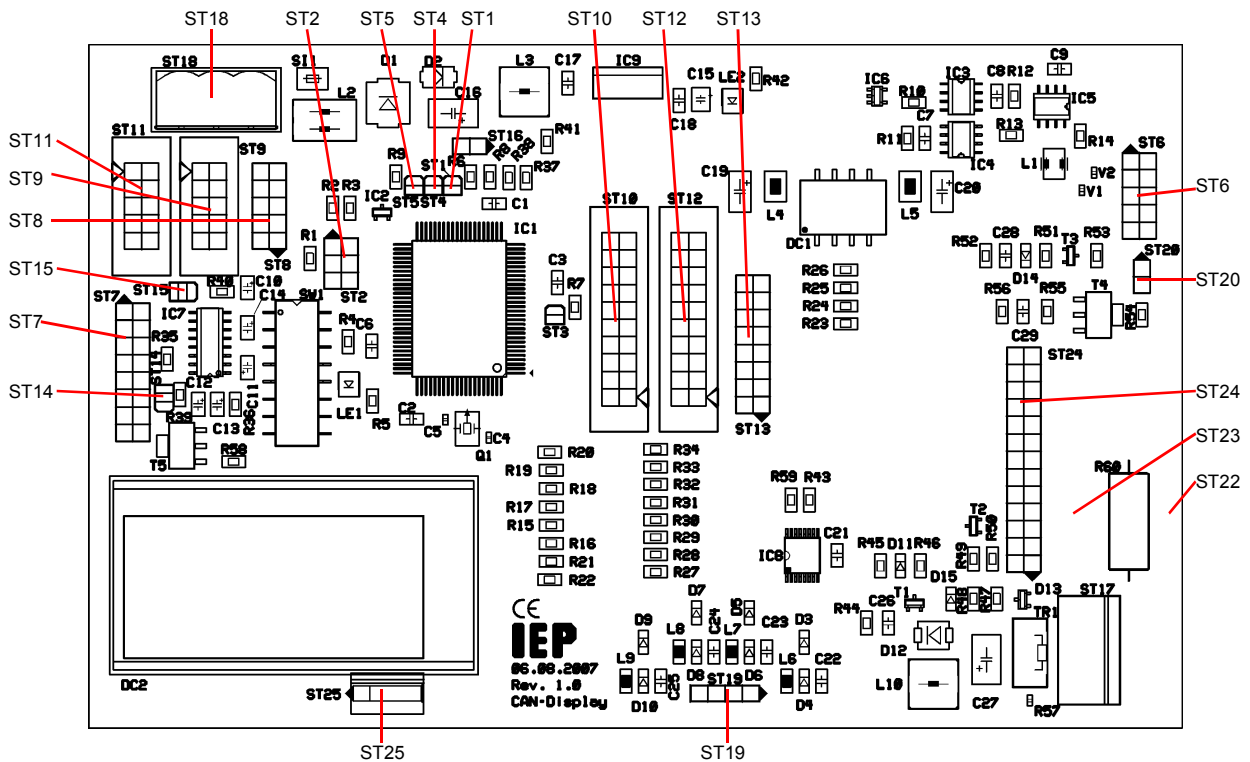
Die CAN-DISP wird mit 5 Volt DC versorgt. Bitte beachten Sie, dass eine Verpolung der Spannung zur Zerstörung der CAN-DISP führen kann!



Die Anschlußbelegung sieht folgendermaßen aus:

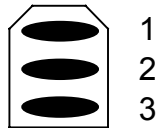


3.3 Lage der Jumper und Steckverbinder



3.4 Beschreibung der Jumper

Die Löt-Jumper werden folgendermaßen gezählt:



3.4.1 ST1 – Baudrate der seriellen Schnittstelle

ST1 wird zur Parametereinstellung der seriellen Schnittstelle verwendet, s.

3.4.2 ST4, ST5 – Reserviert

Die Jumper dürfen nicht geschlossen werden.

3.4.3 ST14, ST15 – Pegel der seriellen Schnittstelle

Soll die serielle Schnittstelle mit RS-232-Pegeln betrieben werden, müssen ST14/15 auf Pos. 2-3 gejumpert werden (default). Nur beim Betrieb mit TTL-Pegeln (Anschluß ST7) dürfen die Jumper auf Pos. 1-2 gesetzt werden.

3.5 TR1 – Kontrastregelung

Der Kontrast des LCD-Displays kann mit dem Trimmer TR1 eingestellt werden. Per Software erfolgt die Feineinstellung, ein NTC sorgt für die Temperaturkompensation.

3.6 SW1 – Einstellen der Kommunikationsparameter

Über den DIP-Switch SW1 werden die Kommunikationsparameter der CAN-DISP eingestellt. Die Einstellung betrifft sowohl die serielle wie auch die CAN-Bus-Kommunikation.

In den folgenden Erläuterungen wird die Schalterstellung "ON" (bzw. Jumper geschlossen) als 1, die Schalterstellung "OFF" (bzw. Jumper offen) als 0 dargestellt,

3.6.1 Serielle Parameter

Die serielle Schnittstelle der CAN-DISP arbeitet grundsätzlich voll duplex mit 8 Datenbits und 1 Stopbit sowie RTS/CTS-Hardware-Handshake. Zur Parametereinstellung werden Jumper ST1 sowie Switch SW1 ausgewertet:

Schalter	ST1	S1	S2	S3	S4	S5	S6	S7	S8
	Baudrate			Parität		Busbetrieb			
1200	1	1	1						
2400	1	1	0						
4800	1	0	1						
9600	1	0	0						
115200	0	1	1						

57600	0	1	0				
38400	0	0	1				
19200	0	0	0				
Parity aus		0	x				
Parity even		1	0				
Parity odd		1	1				
Busbetrieb aus			0	0	0	0	

Der Busbetrieb ist z.Zt. noch nicht unterstützt, S5 ... S8 müssen auf 0 (Off) gestellt sein, andernfalls ist die Kompatibilität mit ggf. folgenden Software-Versionen nicht gegeben.

3.6.2 CAN-Parameter

Über SW1 werden Baudrate und Basis-ID des Moduls eingestellt. Von der CAN-DISP werden 4 aufeinander folgende Identifier (Standard: 11 Bit Länge) belegt.

Schalter	S1	S2	S3	S4	S5	S6	S7	S8
	Baudrate		Basisadresse		Basis-Offset im Adressbereich			
1 MBaud	1	1						
500 kBaud	1	0						
125 kBaud	0	1						
50 kBaud	0	0						
	1152	0x480	1	1				
	896	0x380	1	0				
	640	0x280	0	1				
	384	0x180	0	0				
				0	0	0	0	0
				1	0	0	0	1
				2	0	0	1	0
						
				15	1	1	1	1

Die Basisadresse der CAN-DISP ergibt sich aus Adressbereich und Offset zu:

$$BasisID = Basisadresse + 4 * Basisoffset$$

Beispiel: Am DIP-Schalter ist 0x9C (10011100) eingestellt. Damit ergibt sich:

Baudrate: 10, also 500 kBaud

Basisadresse: 01, also 640

Basis-Offset: 1100, dezimal 12

Als Basis-ID ergibt sich

$$BasisID = 640 + 4 * 12 = 688$$

Die CAN-DISP belegt dabei die Adressen 688, 689, 690 und 691 (s. cap. 5.2).

3.7 Betriebszustand

Die grüne LED LE2 zeigt an, dass die Betriebsspannung des Moduls vorhanden ist. Die gelbe LED LE1 blinkt, wenn Daten übertragen werden.

4 Hardwarebeschreibung

4.1 ST2 – Serielle Debug-Schnittstelle

Die serielle Schnittstelle dient nur der Programmierung des Controllers:

ST2	PIN	PIN	ST2
Vcc	1	2	MD2
Rx	3	4	Tx
GND	5	6	MD1

Nach dem Einschalten gibt die CAN-DISP eine Versions-Meldung mit 38400 Baud aus.

ACHTUNG: Die Schnittstelle stellt nur TTL-Pegel zur Verfügung. Der Anschluß einer normalen RS-232-Schnittstelle führt zur Zerstörung des Moduls.



4.2 Steckverbinder

4.2.1 Display

Hitachi LMG7420:

ST22	PIN
GND	1
Vcc	2
V0	3
CD	4
/WR	5
/RD	6
D0	7
D1	8
D2	9
D3	10
D4	11
D5	12
D6	13
D7	14
/CE	15
/RET	16
-15V	17
/DOFF	18
FS	19
REV	20

Hitachi SP14N001:

ST21	PIN
GND	1
Vcc	2
V0	3
CD	4
/WR	5
/RD	6
D0	7
D1	8
D2	9
D3	10
D4	11
D5	12
D6	13
D7	14
/CE	15
/RET	16
-15V	17
/DOFF	18
FS	19
REV	20
nc	21
nc	22
nc	23
nc	24
nc	25
nc	26

Toshiba:

ST23	PIN	PIN	ST23
nc	1	2	GND
Vcc	3	4	V0
/WR	5	6	/RD
/CE	7	8	CD
nc	9	10	/RET
D0	11	12	D1
D2	13	14	D3
D4	15	16	D5
D6	17	18	D7
FS	19	20	REV

Densitron:

ST24	PIN	PIN	ST24
GND	1	2	Vcc
V0	3	4	CD
/RD	5	6	/WR
D0	7	8	D1
D2	9	10	D3
D4	11	12	D5
D6	13	14	D7
/CE	15	16	/RET
-15V	17	18	REV
FS	19	20	GND
Y+	21	22	X+
Y-	23	24	X-
LED-Vcc	25	26	LED-GND

Die CAN-DISP unterstützt monochrome Displays verschiedener Hersteller mit dem Controller T6963 (Toshiba) in einer Auflösung von 240x128 Pixel.

4.2.2 serielle Schnittstelle

PC-Belegung:

ST9	PIN	PIN	ST9
--	1	2	--
RxD	3	4	RTS
TxD	5	6	CTS
--	7	8	--
GND	9	10	--

MOCS A1:

ST8	PIN	PIN	ST8
GND	1	2	--
TxD	3	4	--
RxD	5	6	--
RTS	7	8	--
CTS	9	10	--

Spezialbelegung:

ST11	PIN	PIN	ST11
--	1	2	CTS
--	3	4	RTS
--	5	6	RxD
--	7	8	TxD
--	9	10	GND

MOCS A2:

ST7	PIN	PIN	ST7
GND	1	2	--
TxD	3	4	--
RxD	5	6	--
RTS	7	8	--
CTS	9	10	--
--	11	12	--
GND	13	14	--
--	15	16	--

A C H T U N G: Diese Schnittstelle führt TTL-Pegel und kann bei Anschluß normaler RS-232-Pegel zerstört werden! Jumper ST14/15 beachten!

4.2.3 Matrixtastatur

Die Tasten müssen zwischen Row x und Col y angeordnet werden.

ST10:

ST10	PIN	PIN	ST10
C 2	1	2	R 1
R 2	3	4	R 3
R 4	5	6	C 1
R 1	7	8	R 2
R 3	9	10	R 4
R 4	11	12	R 3
C 3	13	14	C 4
C 5	15	16	C 6
R 1	17	18	R 2
--	19	20	--

ST12:

ST12	PIN	PIN	ST12
C 1	1	2	R 1
R 2	3	4	R 3
R 4	5	6	C 2
R 1	7	8	R 2
R 3	9	10	R 4
R 4	11	12	R 3
C 3	13	14	C 4
C 5	15	16	C 6
R 1	17	18	R 2
R 5	19	20	R 6

ST13:

ST13	PIN	PIN	ST13
R 1	1	2	C 1
R 2	3	4	C 2
R 3	5	6	C 3
R 4	7	8	C 4
R 5	9	10	C 5
R 6	11	12	C 6
R 7	13	14	C 7
R 8	15	16	C 8

4.2.4 ST19 – Touchscreen

4 polige Stiftleiste:

ST19	PIN
X+	1
Y+	2
X-	3
Y-	4

4.2.5 ST6 – CAN-Bus

2x5 polige Stiftleiste:

ST6	PIN	PIN	ST6
--	1	2	GND
CANL	3	4	CANH
GND	5	6	--
--	7	8	--
--	9	10	--

4.2.6 ST25 – LCD-Beleuchtung

4 polige Stiftleiste:

ST25	PIN
~900 V	1
--	2
--	3
~900 V	4

4.2.7 ST20 – Lautsprecher

2 polige Stiftleiste:

ST20	PIN
LS +	1
LS -	2

4.2.8 ST18 – 24 Volt Versorgung

Phoenix 3 polig :

ST18	PIN
+24 V	1
-24 V	2
PE	3

Wenn dieser Anschluß genutzt wird, darf die CAN-DISP nicht mit 5 Volt (ST17) versorgt werden!



5 Kommunikation

Die CAN-DISP wird wahlweise über eine serielle Schnittstelle oder den CAN-Bus mit Befehlen und Daten versorgt. Neben den darzustellenden Zeichen kann dieser Datenstrom auch Steuerbefehle enthalten.

Steuerbefehle werden mit einem Sonderzeichen eingeleitet und enthalten neben einem Befehlscode ggf. optionale Parameter:

```
ESC befehl [parameter]
```

Default: ESC ist das ASCII-ESC (0x1B). befehl sowie parameter sind binäre Daten.

Ein Empfangspuffer von 1024 Byte entkoppelt die Datenübertragung von der Darstellung. Erreicht der Empfangspuffer eine Füllung von 75%, so signalisiert die CAN-DISP dies dem Sender über ein Handshake-Protokoll. Zeichen, die bei gefülltem Empfangspuffer empfangen werden, gehen verloren.

Der Ausgabedatenstrom der CAN-DISP enthält sowohl Antworten auf Befehle als auch Tastencodes für Matrixtastatur und Touchscreen. Auf der Ausgabeseite berücksichtigt die CAN-DISP ebenfalls ein Handshake-Protokoll. Ist der Ausgabepuffer der CAN-DISP gefüllt (Ausgabe über Handshake blockiert, oder Erzeugung von Ausgabedaten ist schneller, als über die Kommunikationsschnittstelle ausgebar), so wird die Verarbeitung der Befehle wie auch Abfrage von Tastatur und Touchscreen angehalten.

Eine Synchronisation des ausgehenden Rechners auf die Abarbeitungsgeschwindigkeit der CAN-DISP kann z.B. mit dem Befehl `stp_kbd` erfolgen. Die CAN-DISP antwortet auf diesen Befehl mit dem Zeichen NUL (hex 0x00). Da dieses Zeichen von der CAN-DISP nicht als Tastaturdatum gesendet wird, kann der Ausgaberechner eindeutig erkennen, wann die CAN-DISP diesen Befehl abgearbeitet hat.

5.1 Serielle Schnittstelle

Die Betriebsparameter der ser. Schnittstelle werden über ST1 (s. cap. 3.4.1) eingestellt.

Die CAN-DISP unterstützt ein RTS/CTS-Handshake.

5.2 CAN-Bus

Die CAN-DISP verwendet 4 aufeinanderfolgende kurze Identifier, beginnend mit einem Basis-Identifier. Der Basis-Identifier wird über SW1 eingestellt (s. cap. 3.6)

5.2.1 Basis-ID (CANDISP_OUTPUT)

Auf der Basis-ID empfängt die CAN-DISP ihren Eingangsdatenstrom. Die Datenlänge der Empfangspakete ist beliebig, eine Blockung in zusammengehörende Befehle ist nicht erforderlich.

5.2.2 Basis-ID + 1 (CANDISP_FLOW_OUT)

Über diese ID empfängt die CAN-DISP das Handshake-Protokoll des Ausgaberechners. Die CAN-DISP berücksichtigt nur Eingangstelegramme mit einer Datenlänge von 1 Byte und folgendem Dateninhalt:

- 0x00 (CAN_FLOW_STOP) hält die Ausgabe der CAN-DISP an.
- 0x01 (CAN_FLOW_START) erlaubt der CAN-DISP die Ausgabe von Daten.

5.2.3 Basis-ID + 2 (CANDISP_INPUT)

Über diese ID sendet die CAN-DISP ihre Ausgabedaten mit einer Datenlänge von 1..8 Byte je Telegramm. Die Daten sind als Datenstrom zu interpretieren, die CAN-DISP führt keine inhaltliche Blockung durch.

5.2.4 Basis-ID + 3 (CANDISP_FLOW_IN)

Über diese ID signalisiert die CAN-DISP dem Ausgaberechners das Handshake-Protokoll. Die CAN-DISP sendet nur Telegramme mit einer Datenlänge von 1 Byte mit folgendem Dateninhalt:

- 0x00 (CAN_FLOW_STOP) hält die Ausgabe des Ausgaberechners an.
- 0x01 (CAN_FLOW_START) erlaubt dem Ausgaberechner die Ausgabe von Daten.

6 Befehlssatz

Der Befehlssatz orientiert sich an den Möglichkeiten des Controllers T6963.

Die CAN-DISP bietet zwei unterschiedliche Darstellungsmöglichkeiten:

1) Textmodus

Mit einem 6x8-Zeichensatz kann Text mit 16 Zeilen und 40 Spalten dargestellt werden.

2) Grafikbetrieb

Im Grafikbetrieb kann neben pixelgenau positionierbarem Text in variabler Grösse auch Grafik dargestellt werden.

6.1 Befehlsliste numerisch

Zur Befehlsbeschreibung: klicken auf das Mnemo; zurück: klicken auf den Text "Mnemo"

Dez.	Hex.	Mnemo	Funktion
0	00	cls_code	Text löschen
1	01	gsz_code	Textzeiger setzen
4	04	atr_code	Textattribut setzen
5	05	grf_code	Graf OR-TEXT
7	07	plt_code	Pixel setzen x y
8	08	drw_code	Linie zeichnen
9	09	clrs_c	Grafik löschen
18	12	no_tx_ctr	Emulation aus
19	13	text_ctr	Emulation an
20	14	esc_cod_c	ESC setzen
21	15	set_grf_c	Grafikzeiger auf x y
22	16	g_sig_opt	Grafikoption setzen
23	17	mov_cur_c	Grafikcursor relativ
29	1D	wtx_mod_c	Zeichen als Text
30	1E	wgr_mod_c	Zeichen als Grafik
36	24	grf_xor_c	Grafik EXOR Text
44	2C	chr_siz_c	Zeichengrösse
47	2F	cur_chr_c	Grafikcursor def.

48	30	versio_c	Ausgabe Version
55	37	def_key_c	Tastaturcode setzen
61	3D	sound_c	Sound Frequenz
63	3F	str_kbd	Tastaturscan AN
64	40	stp_kbd	Tastaturscan AUS
66	42	kwait_c	Autorepeat setzen
67	43	contr_c	Kontrast einstellen
68	44	brght_c	Helligkeit einstellen
73	49	laut_c	Lautstärke
74	4A	touch_c	Touch Definition
75	4B	tmode_c	Touch Betriebsart
76	4C	tcal_c	Touch kalibrieren
77	4D	scal_c	Touch Kalibrierdaten
78	4E	circle_c	Kreis zeichnen
79	4F	box_c	Box zeichnen
80	50	rbox_c	Box gerundet

6.2 Befehlsliste alphabetisch

Mnemo	Dez.	Hex.	Funktion
atr_code	4	04	Textattribut setzen
box_c	79	4F	Box zeichnen
brght_c	68	44	Helligkeit einstellen
chr_siz_c	44	2C	Zeichengrösse
circle_c	78	4E	Kreis zeichnen
clrs_c	9	09	Grafik löschen
cls_code	0	00	Text löschen
contr_c	67	43	Kontrast einstellen
cur_chr_c	47	2F	Grafikcursor def.
def_key_c	55	37	Tastaturcode setzen
drw_code	8	08	Linie zeichnen
esc_cod_c	20	14	ESC setzen
g_sig_opt	22	16	Grafikoption setzen
grf_code	5	05	Graf OR-TEXT
grf_xor_c	36	24	Grafik EXOR Text
gsz_code	1	01	Textzeiger setzen
kwait_c	66	42	Autorepeat setzen
laut_c	73	49	Lautstärke

mov_cur_c	23	17	Grafikcursor relativ
no_tx_ctr	18	12	Emulation aus
plt_code	7	07	Pixel setzen x y
rbox_c	80	50	Box gerundet
scal_c	77	4D	Touch Kalibrierdaten
set_grf_c	21	15	Grafikzeiger auf x y
sound_c	61	3D	Sound Frequenz
stp_kbd	64	40	Tastaturscan AUS
str_kbd	63	3F	Tastaturscan AN
tcal_c	76	4C	Touch kalibrieren
text_ctr	19	13	Emulation an
tmode_c	75	4B	Touch Betriebsart
touch_c	74	4A	Touch Definition
versio_c	48	30	Ausgabe Version
wgr_mod_c	30	1E	Zeichen als Grafik
wtx_mod_c	29	1D	Zeichen als Text

6.3 Befehlsbeschreibung

Mnemo	cls_code
Dezimal	0
Hex	0x00
Parameter	-

Der Text-Bildschirm wird zu Leerzeichen gelöscht, der Grafikbildschirm bleibt erhalten

Mnemo	gsz_code
Dezimal	1
Hex	0x01
Parameter 1	spalte
Parameter 2	zeile

Der Textzeiger (Cursor) wird auf die Position (spalte,zeile) gesetzt (Grafikzeiger s. set_grf_c).

(0,0) entspricht der linken, oberen Ecke.

Mnemo	atr_code
Dezimal	4
Hex	0x04
Parameter	attribut

Das Darstellungsattribut für Zeichen im Textmode wird gesetzt. Folgende Attribute werden unterstützt:

- 0 TXT_ATTR_NORMAL normale Textdarstellung
- 14 TXT_ATTR_NEGATE invertierte Textdarstellung

Mnemo	grf_code
Dezimal	5
Hex	0x05
Parameter	-

Das Display wird auf kombinierte Text- und Grafikdarstellung geschaltet. Text und Grafik werden ODER-verknüpft.

Mnemo	plt_code
Dezimal	7
Hex	0x07
Parameter 1	x
Parameter 2	y

Das Pixel (x,y) wird gesetzt (Grafikdarstellung muss aktiv sein).

Der Grafikzeiger wird auf (x,y) gesetzt.

(0,0) ist die linke, obere Ecke, die Maximalkoordinaten sind (239,127).

Mnemo	drw_code
Dezimal	8
Hex	0x08
Parameter 1	x
Parameter 2	y

Es wird eine Linie von der aktuellen Position des Grafikzeiger (s. plt_code) zu (x,y) gezeichnet (Grafikdarstellung muss aktiv sein).

Der Grafikzeiger wird auf (x,y) gesetzt.

(0,0) ist die linke, obere Ecke, die Maximalkoordinaten sind (239,127).

Mnemo	clrs_c
Dezimal	9
Hex	0x09
Parameter	-

Der Grafikbildschirm wird gelöscht.

Mnemo	no_txt_ctr
Dezimal	18
Hex	0x12
Parameter	-

Alle zur Darstellung vorgesehene Zeichen werden dargestellt (keine Interpretation von Sonderzeichen).

Mnemo	text_ctr
Dezimal	19
Hex	0x13
Parameter	-

Sonderzeichen werden interpretiert:

- CR (0x0D) positioniert an Zeilenanfang
- DEL (0x7F) löscht das Zeichen an der aktuellen Position
- BS (0x08) positioniert den Cursor 1 Zeichen nach links
- LF (0x0A) positioniert in die nächste Zeile

Mnemo	esc_code_c
Dezimal	20
Hex	0x14
Parameter	ESC-Symbol

Das Befehls-Einleitungszeichen (Default ESC, 0x1B) wird geändert.

Mnemo	set_grf_c
Dezimal	21
Hex	0x15
Parameter 1	x
Parameter 2	y

Der Grafikzeiger (Cursor) wird auf die Position (x,y) gesetzt (Textzeiger s. gsz_code).

(0,0) ist die linke, obere Ecke, die Maximalkoordinaten sind (239,127).

Mnemo	g_sig_opt
Dezimal	22
Hex	0x16
Parameter	Optionen

Optionen für die Textdarstellung im Grafikmode setzen.

Optionen ist eine Bitmaske mit folgender Bedeutung:

- 0x40: GRF_OPT_FETT Fetter Zeichensatz
- 0x80: GRF_OPT_CURON Cursor Ein

Ein gesetztes Bit schaltet die Option ein, ein gelöscht Bit schaltet die Option aus

Mnemo	mov_cur_c
Dezimal	23
Hex	0x17
Parameter	rrnn nnnn

Relative Positionierung des Cursors auf dem Grafikschild.
Im Parameter werden sowohl die Bewegungsrichtung (r) als auch die Anzahl der Positionierungsschritte (n) angegeben:

- 00nn nnnn: GRF_CURMOV_RIGHT: Cursor rechts
- 01nn nnnn: GRF_CURMOV_LEFT: Cursor links
- 10nn nnnn: GRF_CURMOV_DOWN Cursor abwärts
- 11nn nnnn: GRF_CURMOV_UP Cursor aufwärts

Die Positionierung erfolgt in n-fachen der aktuellen Zeichensatzgröße.

Mnemo	wtx_mod_c
Dezimal	29
Hex	0x1D
Parameter	-

Textmode einschalten. Die Textausgabe erfolgt auf dem Textbildschirm (s. wgr_mod_c), die Positionierung kann nur in Zeichenpositionen erfolgen.

Mnemo	wgr_mod_c
Dezimal	30
Hex	0x1E
Parameter	-

Grafikmode einschalten. Die Textausgabe erfolgt auf dem Grafikschild (s. wtx_mod_c), die Positionierung kann pixelgenau erfolgen.

Mnemo	grf_xor_c
Dezimal	36
Hex	0x24
Parameter	-

Grafik- und Textschirm werden EXOR-verknüpft dargestellt.

Dieser Befehl ist z.Zt. noch nicht implementiert.

Mnemo	chr_siz_c
Dezimal	44
Hex	0x2C
Parameter	hhhh 0www

Die Zeichengröße bei der Textausgabe auf dem Grafikschild wird eingestellt.

- h+1: Multiplikator für Zeichenhöhe
- w+1: Multiplikator für Zeichenbreite

Z.Zt. sind nur die Multiplikatorwert 0 und 1 zulässig, d.h. es können nur Zeichen in 1- oder 2-facher Höhe oder Breite dargestellt werden.

Mnemo	cur_chr_c
Dezimal	47
Hex	0x2F
Parameter	0xDB

Der Cursor wird als blinkender Block dargestellt.

Der Parameter wird z. Zt. ignoriert, aus Kompatibilitätsgründen sollte der Wert 0xDB (dez. 219) gesendet werden.

Mnemo	def_key_c
Dezimal	55
Hex	0x37
Parameter 1	Tastaturwert
Parameter 2	Scancode

Festlegung des Tastaturwerts für einen Scancode der Matrixtastatur (s. auch.cap. 6.5, Scancodes). Die CAN-DISP sendet diesen Tastaturwert, wenn der Tastaturscan den entsprechenden Scancode liefert.

Der Scancode besteht aus einer Zeilen- und Spaltenkodierung gemäß zzzz ssss (z= 1..8, s= 1..8). Für jeden gültigen Scancode kann ein Tastenwert angegeben werden.

Wird der ungültige Wert 0 als Scancode angegeben, so wird eine bestehende Tastenwert-Tabelle gelöscht, die CAN-DISP überträgt dann die Scancodes als Tastenwert.

Der Tastenwert 0 wird von der CAN-DISP nicht gesendet. Über den Tastenwert 0 können somit einzelnen Tasten oder auch komplette Tastenbereiche deaktiviert werden.

Die Default-Tastentwerttabelle ist in cap. 6.5, Scancodes, angegeben.

Mnemo	versio_c
Dezimal	48
Hex	0x30
Antwort 1	version
Antwort 2	subversion
optional	weitere Byte

Die CAN-DISP sendet die Versionsnummer der Firmware sowie ggf. weitere Zeichen zur Signalisierung ihrer Konfiguration. Die minimale Antwortlänge beträgt 2 Byte, diese können auch den Wert 0 enthalten.

Als weitere Byte können z. Zt. auftreten:

- 0x54 ('T'): CAN-DISP mit Touchscreen

Die weiteren Bytes können den Wert 0 nicht enthalten.

Das Ende der Ausgabe kann z.B. durch stp_kbd als Folgebefehl erkannt werden, da stp_kbd mit 0 antwortet

Mnemo	sound_c
Dezimal	61
Hex	0x3D
Parameter 1	LSB f [Hz]
Parameter 2	MSB f [Hz]

Die Frequenz für die Soundausgabe wird gesetzt. Die Frequenz kann im Bereich 114 ... 15625 Hz liegen.

Bei einer Frequenz < 114 Hz wird die Soundausgabe ausgeschaltet.

Mnemo	str_kbd
Dezimal	63
Hex	0x3F
Parameter	-

Der Scan der Tastatur wird eingeschaltet (sowohl Matrixtastatur als auch Touchscreen im Tastatur-Betrieb). Bei Betätigung einer Taste sendet die CAN-DISP den Tastencode (s. Scancodes).

Mnemo	stp_kbd
Dezimal	64
Hex	0x40
Antwort	0x00

Der Scan der Tastatur wird ausgeschaltet (sowohl Matrixtastatur als auch Touchscreen im Tastatur-Betrieb).

Die CAN-DISP bestätigt die Abarbeitung dieses Befehls durch Senden von 0x00 als Antwort.

Da der Wert 0x00 nicht im regulären Antwort-Datenstrom der CAN-DISP enthalten ist, kann dieser Befehl auch zur Synchronisation zwischen dem ausgebenden Programm und der CAN-DISP eingesetzt werden.

Mnemo	kwait_c
Dezimal	66
Hex	0x42
Parameter 1	delay
Parameter 2	period

Einstellen des Tastenrepeats (Auto-Repeat):

- delay: Verzögerung bis zum ersten Tastenrepeat
- period: Abstand zwischen folgenden Tastenrepeats

Die Zeitangabe erfolgt in ms, eine Angabe von 0 schaltet die jeweilige Funktion ab.

Als Default sind 100 ms eingestellt.

Mnemo	contr_c
Dezimal	67
Hex	0x43
Parameter	contrast

Der Kontrast des Displays wird eingestellt.

Es sind Werte von 0 ... 255 zulässig.

Als Default ist 200 voreingestellt, der Kontrast hierzu kann mit TR1 manuell abgeglichen werden.

Die Kontrastregelung beinhaltet einen NTC zur Temperaturkompensation des Kontrastverhaltens des LCDs.

Mnemo	brght_c
Dezimal	68
Hex	0x44
Parameter	brightness

Die Helligkeit der Hintergrundbeleuchtung wird eingestellt (nur bei LED-Hintergrundbeleuchtung, eine CCFL-Hintergrundbeleuchtung ist nicht regelbar).

Es sind Werte von 0 ... 255 zulässig.

Z. Zt. wird dieser Befehl ignoriert.

Mnemo	laut_c
Dezimal	73
Hex	0x49
Parameter	Lautstärke

Die Lautstärke der Soundausgabe wird eingestellt.

Es sind Werte von 0 ... 100 zulässig.

Als Default sind 100 voreingestellt.

Mnemo	touch_c
Dezimal	74
Hex	0x4A
Parameter 1	Tastaturcode
Parameter 2	x1
Parameter 3	y1
Parameter 4	x2
Parameter 5	y2

Der Touchscreenbereich (x1,y1) ... (x2,y2) in Pixelkoordinaten wird als rechteckiger Tastenbereich für den Tastaturcode definiert.

Durch Angabe des Tastaturcodes 0 wird die komplette Liste der definierten Touchscreen-Tasten gelöscht.

Zur Default-Vorbesetzung s. Touchscreen-Tasten

Z. Zt. ist die Definition von 10 Touchscreen-Tasten möglich. Werden mehr als zulässig Tasten definiert, so werden die überzähligen Definitionen ignoriert.

Überschneiden sich definierte Tastenbereiche, so wird der in der Scanreihenfolge als erster gefundene Bereich verwendet.

Mnemo	tmode_c
Dezimal	75
Hex	0x4B
Parameter	Mode

Die Betriebsart des Touchscreens wird festgelegt:

- 0x01: TOUCH_KBD Der Touchscreen arbeitet als Tastatur (s. touch_c)
- 0x02: TOUCH_GRF Der Touchscreen arbeitet als Grafik-Eingabe.

Im Grafikbetrieb sendet die CAN-DISP bei erkannter Betätigung des Touchscreens die Koordinaten des Betätigungspunktes in der Form

0xFE x y

Bei fortlaufender Betätigung wird nur bei Änderung der Koordinaten ein neuer Datensatz gesendet.

Mnemo	tcal_c
Dezimal	76
Hex	0x4C
Parameter	Punktezahl
Antwort	Anz. Kalib-Bytes
Antwort	Kalibrierdaten

Ein Kalibriervorgang für die xy-Kalibrierung des Touchscreens wird eingeleitet.

Die Kalibrierung kann über 3, 4 oder 5 Punkte erfolgen, Punktezahlen < 3 werden als 3 interpretiert, Punktezahlen > 5 als 5.

Die CAN-DISP führt die Kalibrierung selbstständig durch. Nach Abschluß des Kalibriervorgangs sendet die CAN-DISP zunächst die Anzahl der Bytes der ermittelten Kalibrierdaten (z. Zt. 24 Byte), gefolgt von den tatsächlichen Kalibrierdaten.

Die Kalibrierung läuft wie folgt ab:

- 1) Der Bildschirm wird gelöscht.
- 2) Ein Zielkreuz wird angezeigt.
- 3) Das Zielkreuz muss betätigt werden.
- 4) Nach Betätigung des Zielkreuzes wird es wieder gelöscht.
- 5) Der Vorgang 2...4 erfolgt für die angegebene Punktezahl.
- 6) Abschliessend werden zur Information alle Zielkreuze angezeigt.

Die CAN-DISP wird mit einer Default-Kalibrierung ausgeliefert. Eine Neukalibrierung ist nur bei wesentlichen Abweichungen zwischen den Pixelkoordinaten bei der Darstellung und den Rückgabekoordinaten bei der Touchscreen-Betätigung erforderlich.

Die Kalibrierdaten werden in der CAN-DISP nicht permanent gespeichert, sie müssen nach dem Start der CAN-DISP mit scal_c übertragen werden.

Mnemo	scal_c
Dezimal	77
Hex	0x4D
Parameter 1	Anz. Kalib-bytes
Parameter 2..n	Kalibrierdaten

Übertragung von Touchscreen-Kalibrierdaten

Die über tcal_c bestimmten Kalibrierdaten (z. Zt. 24 Byte) werden an die CAN-DISP übertragen.

Bis zur Übertragung der Kalibrierdaten arbeitet die CAN-DISP mit einer Default-Kalibrierung.

Mnemo	circle_c
Dezimal	78
Hex	0x4E
Parameter 1	x
Parameter 2	y
Parameter 3	r
Parameter 4	c

Auf dem Grafikschirm wird ein Kreis mit dem Radius r um den Mittelpunkt (x,y) gezeichnet.

Als Farbe c sind die Werte 0 (Pixel löschen) und 1 (Pixel setzen) zulässig.

Mnemo	box_c
Dezimal	79
Hex	0x4F
Parameter 1	x
Parameter 2	y
Parameter 3	f000 000c

Auf dem Grafikschirm wird ein Rechteck vom aktuellen Zeichenpunkt (z.B. über plt_code) zum angegebenen Punkt (x,y). (x,y) wird zum aktuellen Zeichenpunkt.

Als Farbe c sind die Werte 0 (Pixel löschen) und 1 (Pixel setzen) zulässig.

Ist der Funktionscode f gesetzt, wird ein gefülltes Rechteck gezeichnet.

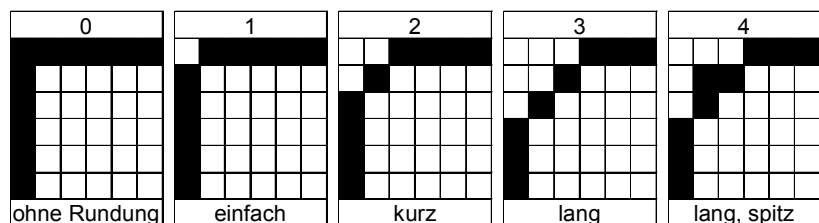
Mnemo	rbox_c
Dezimal	80
Hex	0x50
Parameter 1	x1
Parameter 2	y1
Parameter 3	x2
Parameter 4	y2
Parameter 5	fsss 000c

Auf dem Grafikschirm wird ein Rechteck (x1,y1) – (x2,y2) mit gerundeten Ecken gezeichnet. Der aktuelle Zeichenpunkt wird nicht geändert.

Als Farbe c sind die Werte 0 (Pixel löschen) und 1 (Pixel setzen) zulässig.

Ist der Funktionscode f gesetzt, wird ein gefülltes Rechteck gezeichnet.

Mit dem Stil sss können unterschiedliche Eckenformen gewünscht werden:



Ist die gewünschte Eckenform nicht darstellbar, da der Seitenabstand zu klein ist, wird die grösste darstellbare Eckenform gewählt.

6.4 Zeichensätze

Die CAN-DISP unterstützt den 128-Zeichensatz aus dem Character-Generator des Toshiba T6963. Von den auszugebenden Zeichen wird 32 (0x20) abgezogen, d.h. Steuerzeichen werden unterdrückt. Dann erfolgt der Zugriff über folgenden Zeichengenerator:

↓MSB LSB→	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	blank	!	“	#	\$	%	&	'	()	*	+	,	-	.	/	
1	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
2	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
3	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
4	\	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
5	p	q	r	s	t	u	v	w	x	y	z	{	:	}	~	blank
6	Ç	ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	ì	Ä	Å
7	É	ô	ö	ò	û	ù	ÿ	ö	ü	ç	£	¥	f			

Die Zeichendarstellung erfolgt auf der Basis von 6x8 (BreitexHöhe) Pixeln.

6.5 Scancodes

Die CAN-DISP unterstützt Matrixtastaturen bis zu 8x8 Tasten. Beim Scannen der Tastatur wird ein Scancode bestimmt. Der Scancode besteht aus einer Zeilen- und Spaltenkodierung gemäß zzzz ssss (z= 1..8, s= 1..8). Für jeden gültigen Scancode kann über eine Tastenwerttabelle ein Tastenwert angegeben werden (s. Befehl `def_key_c`).

Die CAN-DISP unterstützt den Tastenwert 0 nicht. Ist einem Scancode ein Tastenwert 0 zugeordnet, so wird das Drücken dieser Taste nicht gesendet.

In der Default-Konfiguration ist die CAN-DISP zum Scan einer 4x5-Tastatur mit folgenden Tastenwerten konfiguriert:

Zeile	Spalte	Scancode	Tastenwert
1	1	0x11	4
1	2	0x12	8
1	3	0x13	12
1	4	0x14	20
1	5	0x15	28
2	1	0x21	3
2	2	0x22	7
2	3	0x23	11
2	4	0x24	19
2	5	0x25	27
3	1	0x31	2
3	2	0x32	6

3	3	0x33	10
3	4	0x34	28
3	5	0x35	26
4	1	0x41	1
4	2	0x42	5
4	3	0x43	9
4	4	0x44	17
4	5	0x45	25

6.6 Touchscreen-Tasten

Bei Einsatz eines Touchscreens im Tastaturmode sind 8 Default-Tasten definiert:

Zeichen	x1	y1	x2	y2
'1' (0x31)	0	0	34	18
'2' (0x32)	0	36	34	54
'3' (0x33)	0	72	34	90
'4' (0x34)	0	108	34	126
'5' (0x35)	206	0	240	18
'6' (0x36)	206	36	240	54
'7' (0x37)	206	72	240	90
'8' (0x38)	206	108	240	126

Dies entspricht folgendem Bildschirmeindruck:

1		5
2		6
3		7
4		8