LC DESIGN

Analogue Data Display Unit

preliminary

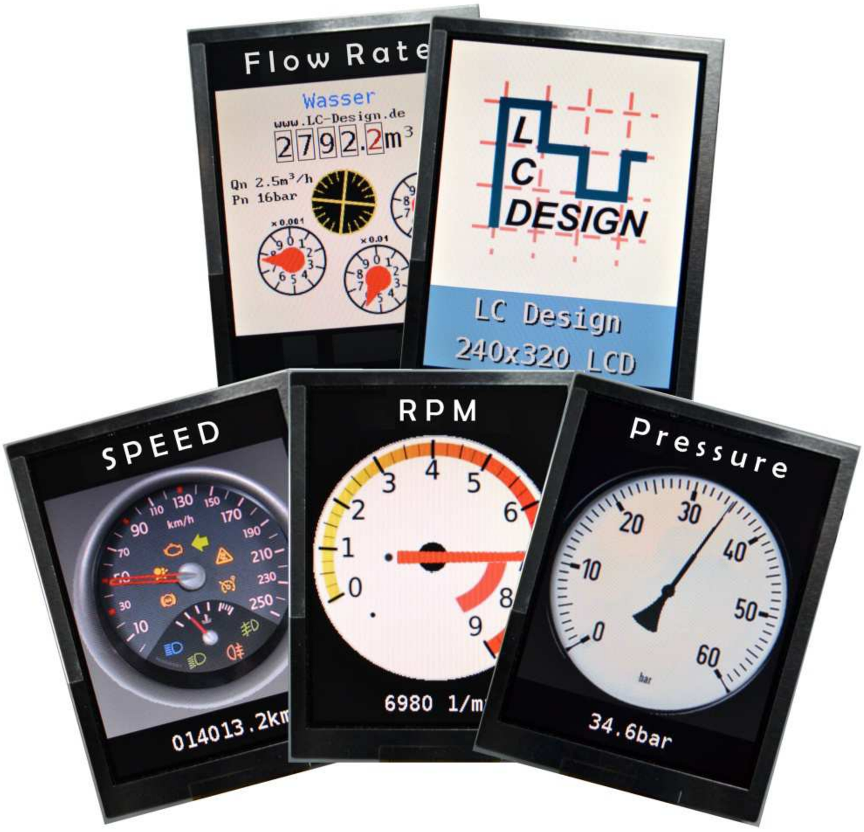
short

© LC Design
Inh. Dipl. Ing. Hartmut Putzig
Johann-Knecht-Straße 28 • 63785 Obernburg a.M.
Telefon 0 60 22/61 44 30 • Fax 0 60 22/61 44 31
Internet http://www.LC-Design.de
e-mail Support@LC-Design.de

This document and the information given in connection with, so far is based on LC Design creations, are and remain the property of LC Design. This document and all information are confidential and it is not allowed to make them accessible to third parties without having the previous written consent of LC Design. All rights especially in relation to inventions are reserved.

Directory

Description



The analogue data display unit (ADD) consists of a TFT-display with a resolution of 240xRGBx320 dots and a processing unit which can handle different interfaces. USB, RS232, SPI and I2C can be selected.

The unit is powered either via an USB-2.0 – interface or from another 5V source, depending on the used interface.

The display content can be programmed with a simple command structure to keep the data transfer in the application as low as possible and save. Especially the simulation of an analogue instrument is made easy due to the possibility to load dial and pointer as a picture, while the picture of the position of the pointer in relation to the dial is calculated in the unit without the need to transfer the final picture. Only the value for the desired pointer position and angle is needed.

Views

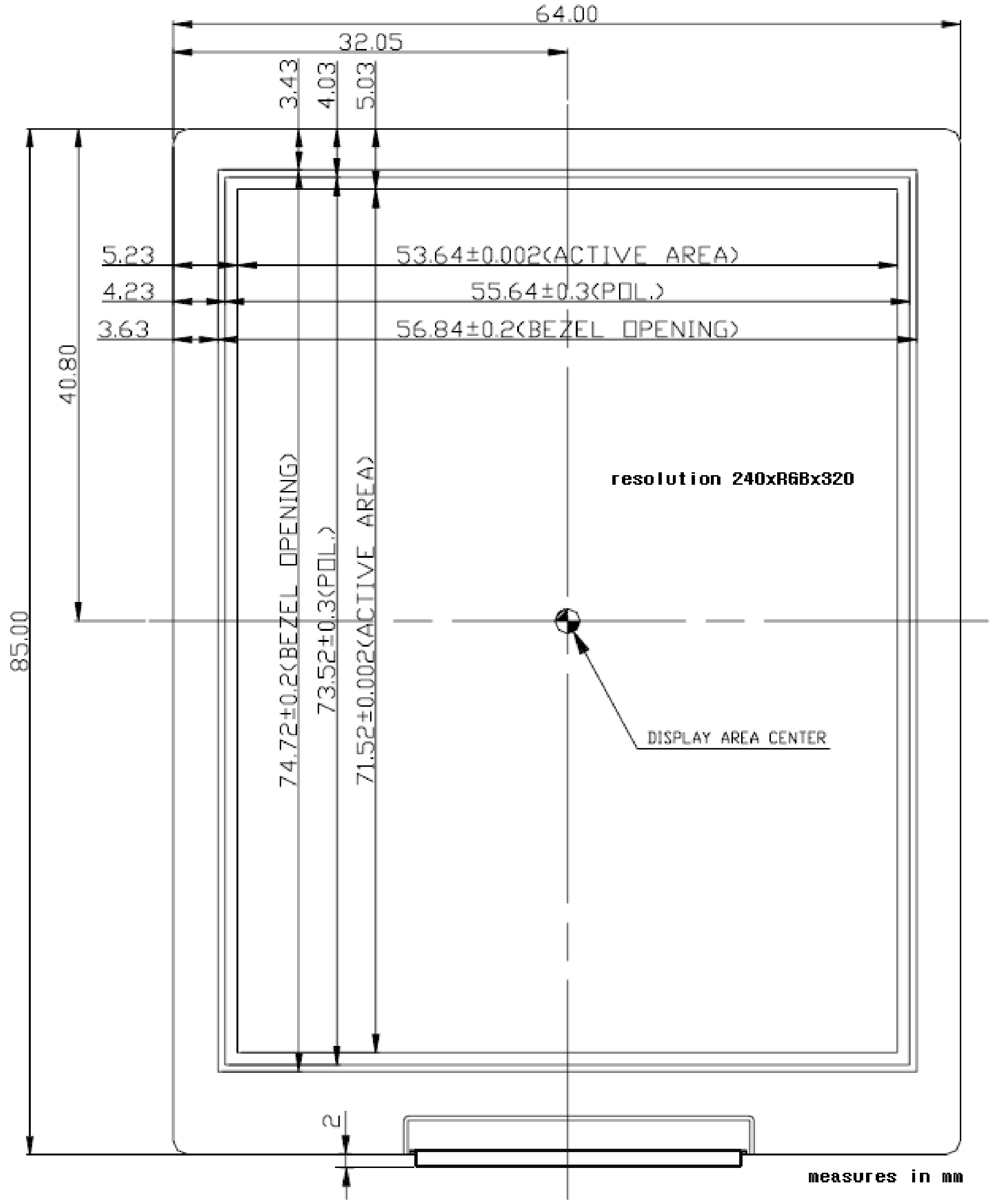
front view:

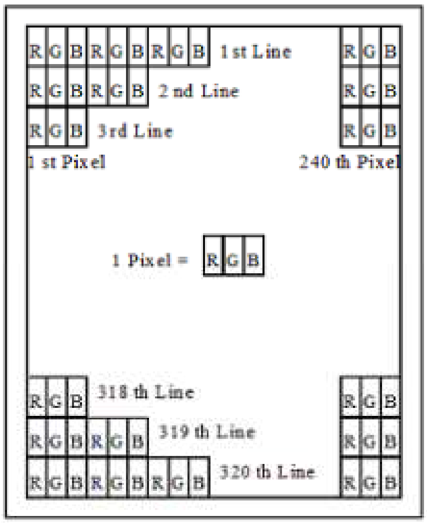
The size of the unit is mainly defined by the size of the display.

view from the back:

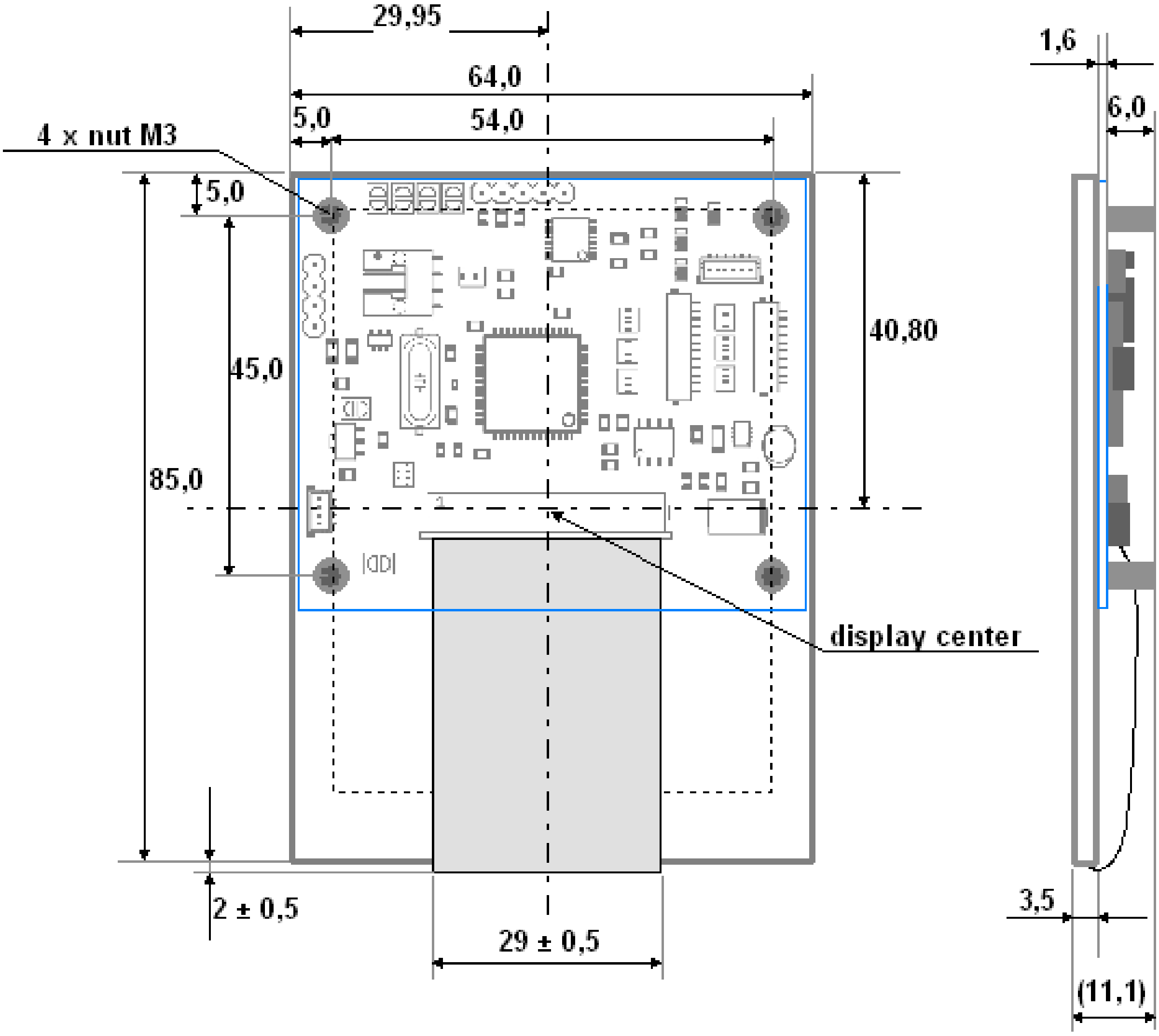
Dimensions

Display




R G B R G B R G B
1 st Line
R G B R G B
2 nd Line
R G B
3rd Line
1 st Pixel
240 th Pixel
1 Pixel = R G B
318 th Line
319 th Line
320 th Line

Module

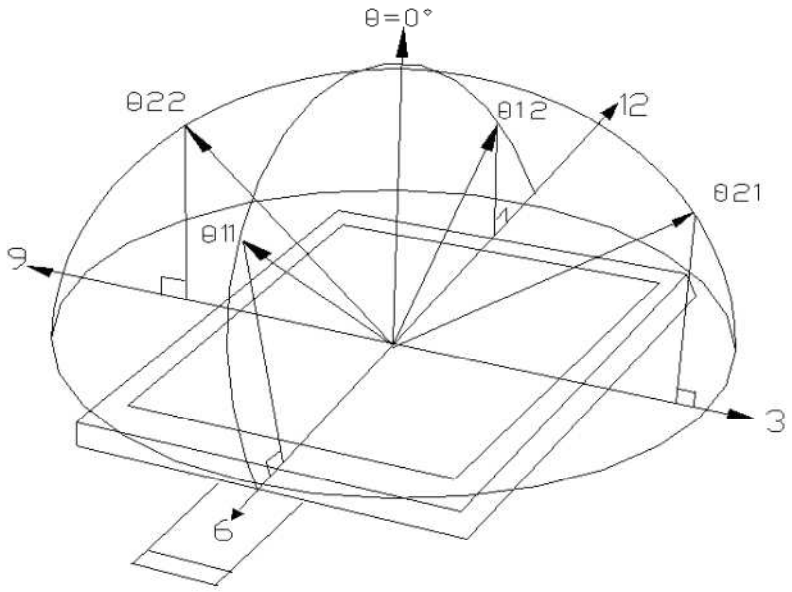

29,95
64,0
5,0
54,0
4 x nut M3
5,0
45,0
85,0
40,80
display center
2 ± 0,5
29 ± 0,5
1,6
6,0
3,5
(11,1)

Optical characteristics

Ta=25°C

Parameter		Symbol	Condition	min.	typ.	max	Unit
Viewing Angle	Horizontal	ϴ21, ϴ22	CR≥ 10	70	75	-	deg
	Vertical	ϴ11		45	50	-	deg
		ϴ12		55	60	-	deg
Contrast Ratio		CR	At optimized viewing angle	200	400	-	-
Brightness		L	θ =0°	400	450		cd/m^2
Uniformity		U%	θ =0°	75	80	-	%
Response	Rise	Tr	θ =0°	-	15	30	ms
	Fall	Tf		-	25	50	ms
White Chromaticity		x	θ =0°	0.28	0.32	0.36	-
		y		0.30	0.34	0.38	-
LED Life Time		-	+25°C	20.000	30.000	-	hrs

The definitions of viewing angles are as follows:



Power supply

The unit has to be powered with 5V, even all logic levels are on a 3.3V base.

There is no special connector for power supply. For easy use every interface connection includes pins for power supply.

Pin assignments

SPI- interface

CON 2 10 pin single row header, 1.00 mm pitch, JST BM10B-SRSS

Pin	Signal	Function
1	V+	5V supply
2	SDI	Serial data input 3.3V level (5V tolerant)
3	GND	Reference level 0V
4	SDO	Serial data output 3.3V level
5	GND	Reference level 0V
6	SCI	Serial clock input 3.3V level (5V tolerant)
7	GND	Reference level 0V
8	SS	Slave select input 3.3V level (5V tolerant)
9	GND	Reference level 0V
10	BUSY	Busy indication output 3.3V level

USB- interface

CON 5, 4 pin connector, 2.00mm pitch, FCI 95000-004LF

Pin	Signal	Function
1	V+	5V supply
2	USB D -	USB differential signal D -
3	USB D+	USB differential signal D +
4	GND	Reference level 0V

RS232- interface

CON 1, 6 pin single row header, 1.00 mm pitch, BM06B-SRSS

Pin	Signal	Function
1	V+	5V supply
2	TXD	Output RS232 or 3.3V level
3	RTS	Output RS232 or 3.3V level
4	RXD	Input RS232 or 3.3V level (5V tolerant)
5	CTS	Input, currently not used RS232 or 3.3V level (5V tolerant)
6	GND	Reference level 0V

I2C- interface

CON 4, 4 pin single row header, 1.00 mm pitch, BM04B-SRSS

Pin	Signal	Function
1	V+	5V supply
2	SCL	Serial clock no pull up, 3.3V level (5V tolerant)
3	SDA	Serial data no pull up, 3.3V level (5V tolerant)
4	GND	Reference level 0V

keyboard interface

CON 6, 12 pin single row header, 1.00 mm pitch, JST BM12B-SRSS

Pin	Signal	Function
1	1	Button top left
2	2	Button top centre
3	4	Button top right
4	8	Button middle left
5	16	Button middle centre
6	32	Button middle right
7	64	Button bottom left
8	128	Button bottom centre
9	256	Button bottom right
10	512	Underline left
11	1024	Underline right
12	GND	Reference level 0V

All buttons switch to ground if pressed

USB Controller

The connection to the host will automatically be detected by the host. Afterwards a driver file has to be installed. This will create a virtual COM port with the next available address.

Windows Server 2008
Windows Server 2008 x64
Windows Vista
Windows Vista x64
Windows 7
Windows 7 x64
Windows XP
Windows XP x64
Windows 2000
Windows Server 2003
Windows Server 2003 x64 will be supported.

Suspend mode:

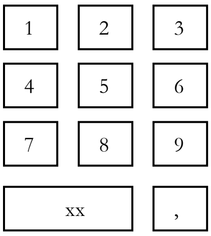
The unit will not comply to the suspend mode for USB devices. The backlight will be switched off, but the microcontroller will remain powered. Therefore the current in that mode will be around 50mA to 100mA. For most systems this will not be a problem. If the powering system (PC) allows a higher current, then, by setting a jumper, the display can also be backlit to show a logo or anything else at power up, even before USB drivers are running and grant the higher current.

Keyboard reading

The keyboard reading sends a code packet any time a key is pressed or released or on request depending on the mode, which has been set upfront.

The packet starts with start byte (211), length (4), command (50) followed by the acknowledge code 0 and low and high byte of the actual pressed keys. (Details see command set and command structure.)

0	no key pressed
1	key 1 pressed
2	key 2 pressed
4	key 3 pressed
8	key 4 pressed
$2^{(K-1)}$	key K pressed



If more than one key is pressed, the values are added.
e.g. key 1 and key 3 pressed => 1 + 4 = 5

Command set

command	short form
1	WRITECHAR
2	LINE FEED
3	CARRIAGE RETURN
4	GOTOXY
5	FONT
6	TEXTCOLOR
8	WRITECHARROT
10	SETPIXEL
11	LINE
12	CIRCLE
13	BLOCK
14	LOADPIC
20	BRIGHTNESS
21	CLEAR
22	TRANS
23	INVERT
24	INVERTAREA
30	REINIT
31	RESTART
33	LOADPICFLASH
34	STORESTARTUP
34	STORESTARTUP
35	STOREPICFLASH
36	READPIC
37	COPYPIC
38	LOADPICFLASHROT
39	STOREINDICATOR
40	LOADINDICATOR
41	SOFTINFO
42	HARDINFO
43	LCDINFO
44	LCDSTRING

	command	short form
50	50	GETKEYS
51	51	AUTOKEYS
52	52	CLEARKEYS
60	60	UPDATE
61	61	AUTOUPDATE
70	70	ANSWER
80	80	LOADPICFLASHBLEND
81	81	LOADPICFLASHBLENDBACK
82	82	BLENDAREA
90	90	MAINPOS
91	91	WRITECHARSUB
92	92	BLOCKSUB
93	93	LOADPICSUB
100	100	----

In all cases, except picture load, the colour has to be specified as 16bpp, even only less different colours are needed. See down low calculations for colour conversion

* key scan and display activities cannot be controlled at the same time.

If a command is detected (start byte received), no key stroke is send to the host. The key strokes will be monitored, stored in a buffer of max. 256 entries and

send to the host after the command has been executed. This will typically delay the key stroke by a time, depending of the actual command between a few

usec and a few msec in normal case. If a picture is transferred the delay may be longer.

Composition of command set

- start byte (211) indicator of communication
- length of the string, command and data, without CRC (max. 64 bytes)
- command see list above
- data
- CRC lowbyte
- CRC highbyte

The CRC has to be calculated including length, command and data.
The CRC-CCITT (CRC-16) with the start value of 0xFFFF and the polynomial x16 + x12 + x5 + 1 has to be used.

The response has the same structure.

- start byte (211)
- length of the string, command and data, without CRC
- command return
- failure code or data
- CRC lowbyte
- CRC highbyte

The code of the command will be returned followed by a failure code of 1 Byte:

0	command executed successfully
1	timeout
2	size invalid (that means >64 Bytes)
3	CRC failure
4	coordinates invalid
5	invalid parameters
6	no picture available
7	number of parameters incorrect
8	unknown command
128	picture data transferred

If a failure occurs which prevents the reception of a valid command code, 0 will be send back as command
code (e.g. timeout error, CRC error, packet length error).

After reception of the start byte of a command a timeout of 10 seconds will be started.
Within this 10 seconds the command or all picture data must be fully transferred.

If the transfer is not finished within this time, the transfer will be aborted and timeout error will be send.

Transparency function

If this function is active, a comparison between colour to be drawn and transparency colour will be carried out with all drawing operations, draw line, draw rectangle, etc.
At identity on colour on any position of the display, the former colour of that pixel will be kept.
This allows e.g. to write a text with a transparent fill-up colour over a picture. The rest will keep the colour of the former display content. This will also work with pictures.

The comparison is time consuming, because it is made pixel by pixel. This lengthens the drawing operations. Therefore it is recommended to switch off this function if not needed.

Loading of pictures

Different colour depth can be selected: 16bpp, 8bpp, 4bpp or 1bpp.

- 16bpp uses the standard RGB 5:6:5 bit format. 5bit for red, 6bit for green, 5 bit for blue.
  Within this format all colour values will be transferred.
  2 bytes per pixel have to be transferred.
- 8bpp reduces the amount of data to 1 byte per pixel. The data will be transferred in the 3:3:2 format.
- 4bpp uses the standard 16 colour palette, which means 1 bit per red, green and blue and 1 bit for the intensity.
- 1bpp transfers only 2 colours which are taken from text colour and background colour. With 1 byte, data for 8 pixels can be transferred.

If the amount of bits to be transferred in 4bpp or 1 bpp mode not fit exactly to the number of bytes, which will be transferred, the not used bits per row will be neglected.
e.g. to transfer the data for 60 pixel in the 1bpp format 60/8 = 7,5 byte would be needed.
8 bytes will be transferred and 4 bits neglected.
The data in 4bpp and 1bpp format have to be transferred in LSB first configuration. Bit 0 will be the first pixel set in the data string.

Storing and loading pictures into/from flash

Flash memory has not an unlimited number of write cycles. Therefore the user should avoid unnecessary write cycles and distribute the pictures over all available picture storage locations evenly. The internal flash memory of the microcontroller has at least 1000 write cycles per storage location, external flash memory even more then 100000. Whenever possible the external memory should be preferred, because it is more endurable. The load time for a picture from the external memory is slightly higher than from the internal memory (~100msec instead of ~ 50 msec for a 240x240 picture).

Recommended picture storage location usage:
- start up picture: location 0
- background images: locations 6-21
- small rotated images/needles: locations 2-5
- large rotated images/needles: locations 0-1
- icons: locations 2-21

Display needle indicators

Displaying analogue needle indicators (and other rotated objects) is simple and fast because of an extra command witch does all the necessary work.
To use this command the user must first set a background image (optional) and the needle picture and parameters and store them into the flash memory like normal pictures. In fact, needle indicators are stored like normal pictures, except that additional parameters are stored together with the picture data and they can not put in external memory. These pictures can be loaded afterwards just like normal pictures, ignoring the additional data.

To display the needle indicator, the software needs to know the following parameters:

- position of needle in display
- rotation axis of needle image
- transparent colour of needle image
- rotation angle for minimum input value
- rotation angle for maximum input value
- minimum input value
- maximum input value

The allowed input value range is 0 to 65535 for maximum resolution of rotation. The user has not to use the full range, instead he can use any part of it and the needle angle is scaled to fit the given range.
Normally the needle moves clockwise with increasing input values, even if maximum angle is lower than minimum angle. For example if the minimum angle is given is 16384 (3 o'clock) and the maximum angle is 0 (12 o'clock), it will move clockwise from 3 o'clock to 12 o'clock. If the angle values are reversed, the needle will move from 12 o'clock to 3 o'clock. To reverse the movement, the input values must be swapped, thus the minimum input value must be larger than the maximum value. The angle values must also be swapped because the maximum input value always corresponds with maximum angle. If the user sets e.g. 100 as minimum value, 16384 as minimum angle and 50 as maximum value, 49152 as maximum angle, the needle will start at 9 o'clock at an input value of 50 and move counter clockwise to 3 o'clock with the input value increasing towards 100.
If the input value is outside the given input range, the needle will stop at the minimum or maximum angle position.

The needle indicator command uses all these stored parameters and the data given in the command and first loads the background image (if given) with previous transparency settings. Then it calculates the needle angle and draws it with transparency settings stored together with the needle picture and updates the display if autoupdate is activated. Afterwards the old transparency settings are restored.

Main & sub display

The display has a resolution of 240xRGBx320. This area is divided into a main screen with 240x240 pixels and a sub screen with 240x80 pixels. All drawing functions can be used within the main display area. For the sub display area a limited set of drawing functions is available.
Only the main screen uses double buffering: All drawing functions draw into memory. The final image is then copied to the screen. The sub display area is written directly.
The software ensures that both screens are logically separated. If any coordinate is outside of one screen it will never overflow into the other. Therefore, if the sub screen is not used, there is no need to use any of the sub display commands and leave it filled with the default colour.
With default settings the main screen will start at top of display and the remaining 80 lines will be filled with black colour. To better fit the display to the application, the user can overwrite default setting with additional infos given with startup picture or change the main screen position at any time with the MAINPOS command and fill the sub display area with a new colour, text or an image.

Blend functions

The blend functions merge two pictures together by using a given percentage of information from the first and the remaining fraction of the second image. This allows to display half transparent images or fading between two images.
When blending the display contents many times in small steps it should be kept in mind that the display has a limited number of displayable colours. This will result in rounding errors and very dark colours will even not fade at all, because they will stick at black because of rounding errors. On the other hand, this can give a nice looking effect if the adequate combination of picture and fading values is used, particularly together with the transparency function.

Default settings

After power-on the following settings are valid.

- character font	2 (12x22)
- text colour	white
- background colour	black
- transparency	off
- autokey	on
- answers	on
- picture updates	auto
- brightness	50 (=20%) or depending on start screen setting

Colour conversion

All colours values have to be transmitted in 16bpp mode, except a picture is transferred.

- Conversion 3x 8bit RGB data to 16bpp: (((b&248)/8) | ((g&252)*8) | ((r&248)*256))

- Conversion 3x 8bit RGB data to 8bpp: (((b&192)/64) | ((g&224)/8) | (r&224))

- Colour table for 4bpp mode (based on 3x8 bit RGB):

(0, 0, 0)
(0, 0,170)
(0,170, 0)
(0,170,170)
(170, 0, 0)
(170, 0,170)
(170, 85, 0)
(170,170,170)
(85, 85, 85)
(85, 85,255)
(85,255, 85)
(85,255,255)
(255, 85, 85)
(255, 85,255)
(255,255, 85)
(255,255,255)

Character fonts

Eight different font sizes are available, based on the font in the table down low.

Font 0: size 6 x 8
Font 1: size 8 x 14
Font 2: size 12 x 22
Font 3: size 16 x 30
Font 4: size 24 x 42
Font 5: size 32 x 58
Font 6: size 48 x 84
Font 7: size 64 x 116

The first 32 characters are not available at all font sizes. For small sizes not all symbols are displayable. For font size 4 and larger they are removed to save memory.

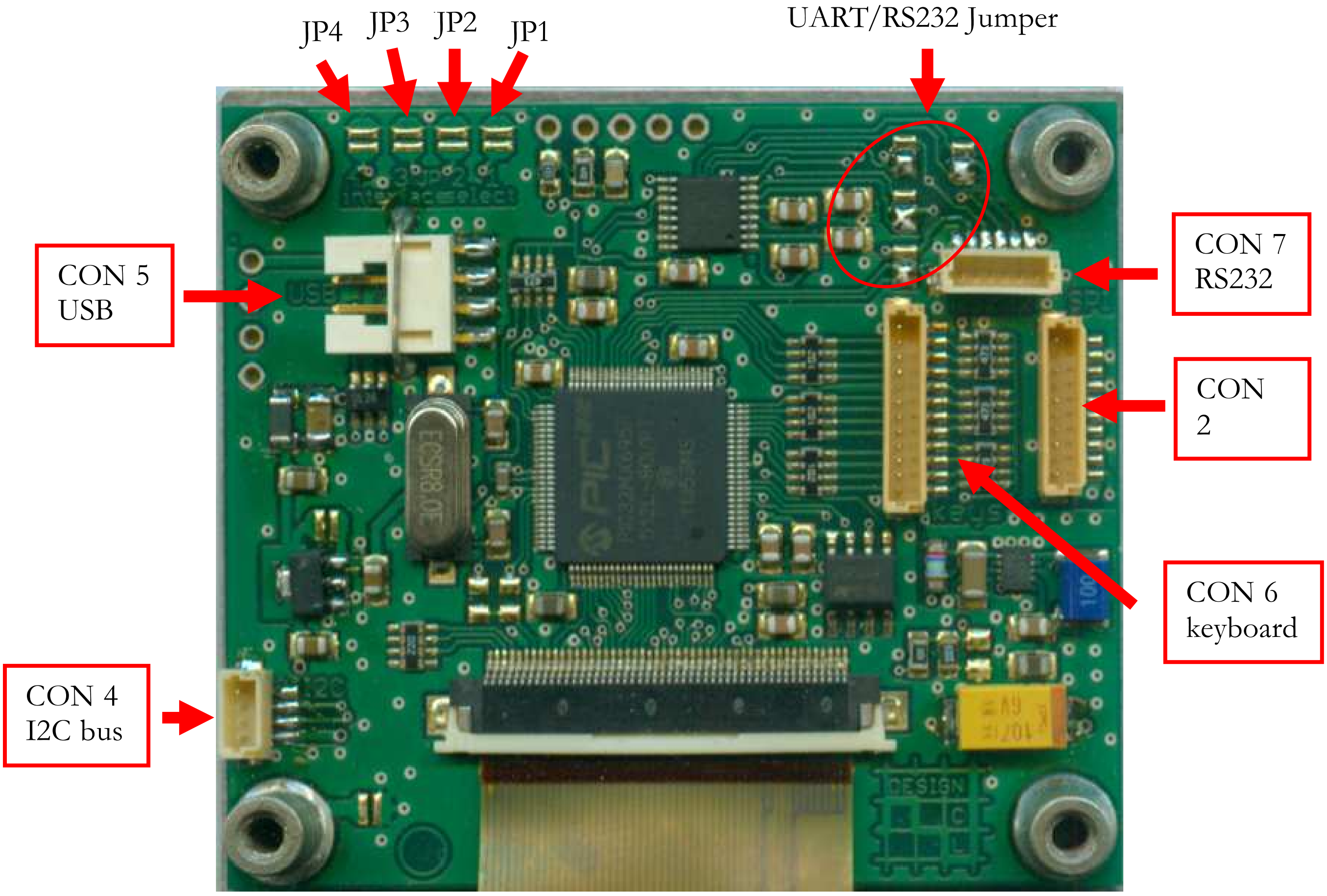
	☺	☻	♥	♦	♣	♠	○	●	◘	◙	♂	♀	♪	♫	☼
►	◄	↕	‼	¶	§	▬	↨	↑	↓	→	←	∟	↔	▲	▼
	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	⌂
Ç	ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	ì	Ä	Å
É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	ø	£	Ø	×	ƒ
á	í	ó	ú	ñ	Ñ	ª	º	¿	®	¬	½	¼	¡	«	»
░	▒	▓	│	┤	Á	Â	À	©	╣	║	╗	╝	¢	¥	┐
└	┴	┬	├	─	┼	ã	Ã	╚	╔	╩	╦	╠	═	╬	¤
ð	Ð	Ê	Ë	È	ı	Í	Î	Ï	┘	┌	█	▄	¦	Ì	▀
Ó	ß	Ô	Ò	õ	Õ	µ	þ	Þ	Ú	Û	Ù	ý	Ý	¯	´
-	±	‗	¾	√	∑	÷	¸	°	¨	▪	¹	³	²	■	Ω

Interface selection

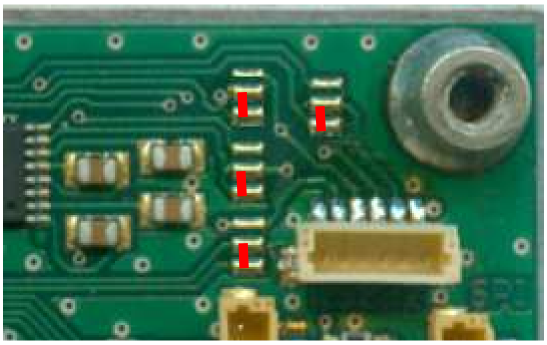
The module pcb carries four solder jumpers. The first two (JP1, JP2) selects the interface, the second two (JP3, JP4) set different options on each interface.

JP1	JP2	JP3	JP4	Interface	Option
open	open	open	open	USB	normal operation
		closed	open		backlight always on
		open	closed		reserved (do not use)
		closed	closed		reserved (do not use)
closed	open	open	open	UART/RS232	19200 Baud
		closed	open		57600 Baud
		open	closed		115200 Baud
		closed	closed		500000 Baud
open	closed	open	open	SPI	mode 0
		closed	open		mode 1
		open	closed		mode 2
		closed	closed		mode 3
closed	closed	open	open	I^2C	I^2C address 0x70
		closed	open		I^2C address 0x72
		open	closed		I^2C address 0x78
		closed	closed		I^2C address 0x7A

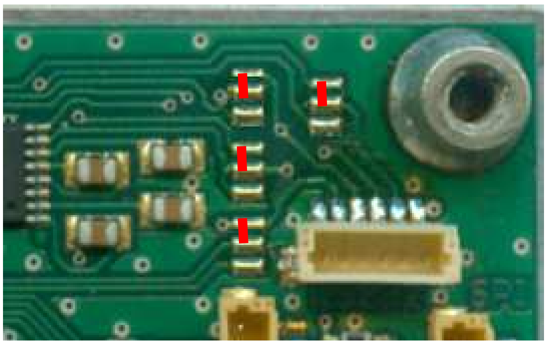
jumper and connector locations:



UART/RS232 jumper settings:



Jumper setting for RS232 levels



Jumper setting for 3,3V logic levels

Notes for UART (RS232), SPI and I²C interfaces

If the input buffer is filled by more then 75%, the module signals to stop data transfer.
The user should stop data transmission to avoid a buffer overflow but can finish currently transferred command if its size is not longer than 256 bytes. Handshaking is done via different ways dependent on selected interface:

- UART/RS232: RTS is set to high
- SPI: BUSY is set to high
- I²C: BUSY (from SPI) is set to high and module address is not acknowledged. Note: During a transmission it is without checking BUSY from SPI not possible to determine buffer state, because address acknowledge only affects start of transmission. This should be kept in mind when using I²C.

Loading pictures into flash (STOREPICFLASH, STORESTARTUP and STOREINDICATOR commands) is possible via all interfaces, but should only be done using USB. During programming of flash no reception is possible, even if there is room in the input buffer. Therefore the module sets the BUSY line and RTS to high and stops acknowledging of I²C data 1ms prior of starting programming. If data transmission continues, data loss can occur.

When transferring picture data to the module, a special case occurs with SPI interface selected:
Because data is read by shifting in data, it is not possible to read the answer without sending (dummy) data to the module. In normal command mode, characters other than the starting character are ignored. For picture data all values are interpreted as data. This would distort picture when reading the command answer. To avoid this problem, in SPI mode an additional start byte is necessary to start picture transfer:
After acknowledge of picture loading command is send, the module ignores all received characters until 0x55 is received. After that all data are interpreted as picture data until picture transfer is finished.
When sending dummy data to read an answer or to finish an unfinished command, send 0 as data.

In SPI and I²C Mode the user initiates a read operation. Therefore the module always has to send some data even when no data is available. Then the module sends 0 as dummy data. When waiting for an answer the user should read data and discard all data until the start byte (211) is received. After that the remaining data of answer should be read.
The start of an answer can occur any time, so it is not necessary to deselect the module in SPI mode or restart a read operation in I²C.

Answer data are collected in a transmit buffer, therefore the user can send several commands before reading answer data. About 100 answers can be stored until the buffer overflows. If this occurs all further answers (or parts of them) will get lost. This should be avoided.

SPI

SPI frequency can be as high as 25MHz but the average data rate (over 8 or more bytes) must not be higher than 250kByte/s. Otherwise data loss can occure.

Technical data

Logic supply voltage:	5V ± 5%
Supply current:	typical 220 mA max. 600 mA (mainly depending on backlight setting)
Operating temperature:	-25 °C to +75°C
Storage temperature:	-30°C to +80°C

Precautions

- The display unit is an electrostatic sensitive component. To prevent destruction of the elements by static electricity, be careful to maintain an optimum work environment.
    a. Be sure to ground the body when handling the display units.
    b. Tools required for assembly, such as soldering irons, must be properly ground.
    c. To reduce the amount of static electricity generated, do not carry out assembly and other work under dry conditions.
    d. The LCD is coated with a film to protect the display surface during assembly and transportation. Be careful when peeling off this protective film since static electricity may be generated. Pull it off slowly.

- The display panel is made of glass. Do not subject it to a mechanical shock by dropping it from a high place, etc.

- The surface of the display is covered with a polarizer, which easily can be scratched with spiky and sharp materials. Handle the polarizer carefully.
  if the display surface is contaminated, breathe on the surface and gently wipe it with a soft dry cloth.
  If still not completely clear, moisten cloth with one of the following solvents:
    - Isopropyl alcohol or Ethyl alcohol
  Solvents other than those may damage the polarizer. Especially, do not use the following fluids:
    - Water, Ketone, Aromatic solvents
  Remove any fluids immediately.

- Condensation on the display surface is not allowed. This will damage the polarizer in short term.

- The display is pressure sensitive. Uneven or high pressure cause change in optical appearance or may destroy the display. Be careful while mounting the unit. The mounting must not cause any mechanical stress in the display or its interconnection.

- Do not expose the display to direct sunlight or intensive ultraviolet rays for many hours. The polarizer and the liquid crystal are deteriorated by ultraviolet rays.
- Don´t activate the same area of the display with the same text or picture for longer than 5 minutes. This may cause an image sticking on the display.
- The display is backlit with white LEDs. The lifetime of these LEDs is not as long as of green or red LEDs. Therefore it is recommend to reduce the brightness to a level which is just sufficient for the application and to reduce the brightness to a low level or switch the backlight off, if the display is not used.
- The write cycle count of flash memory is limited. Use store picture command as rarely as possible.

Revisions

22/06/2012	short description