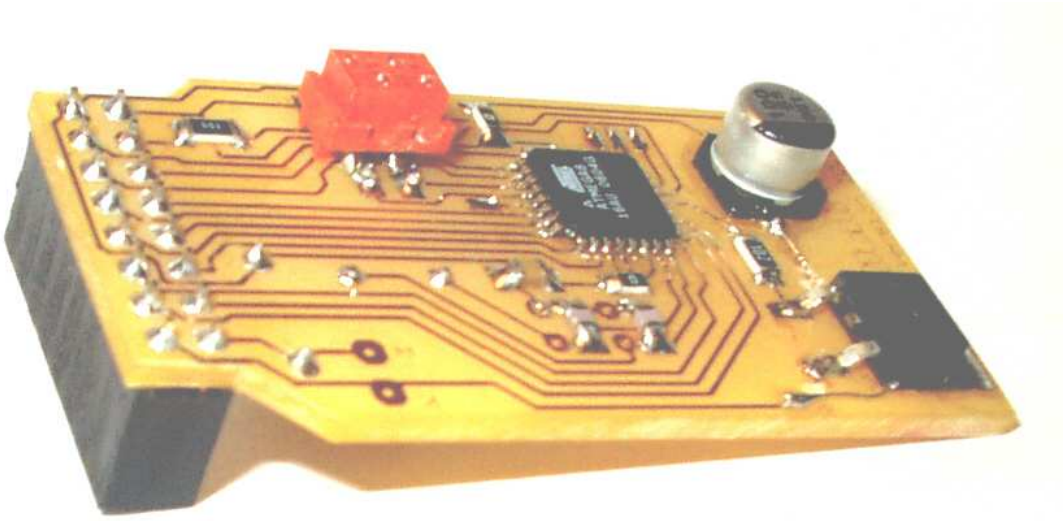
I2C LC-DISPLAY ADAPTER MANUAL

Typ V1020 V3c



Steffen Hörhold
Wolfgangstr. 13
06366 Köthen

email: steffen.hoerhold@web.de

1. General discription

The I²C LC-Display Adapter enables easy interfacing a LC-Display (HD44780) to an I²C-Bus (two wire serial interface). Communication between a host and the I²C LCD takes places via the two wire bi-directional serial interface. It is possible to connect up to 127 devices on the same bus. Slave addresses are programmable via controller.

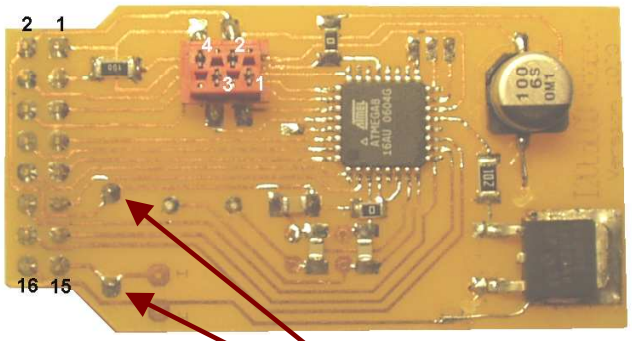
Features:

- 5 V supply voltage
- for positively LCD Bias
- LCD Bias adjustable at I2C Bus
- Backlight adjustable at I2C Bus
- Low-level input voltage (SCL,SDA only): min. –0.5V - max. 0.3V_{DD}
- High-level input voltage (SCL,SDA only): min. 0.7 V_{DD} - max. 5.5V
- Up to 400 kHz byte-wide I2C-bus communication port (it depends on LC-Display)

Note:

Only LCD using Vo more positively.

2. Pinning Information V1020



Resistor to limiting current of Backlight.
!!! Warning !!!
Default is bridged !!!

IN Connektor V1020 (AMP Micro-Match) :

Pin		Description
1	SDA Line	serial data line
2	Vdd	5V Supply Voltage
3	SCL Line	serial clock line
4	GND	0V ground

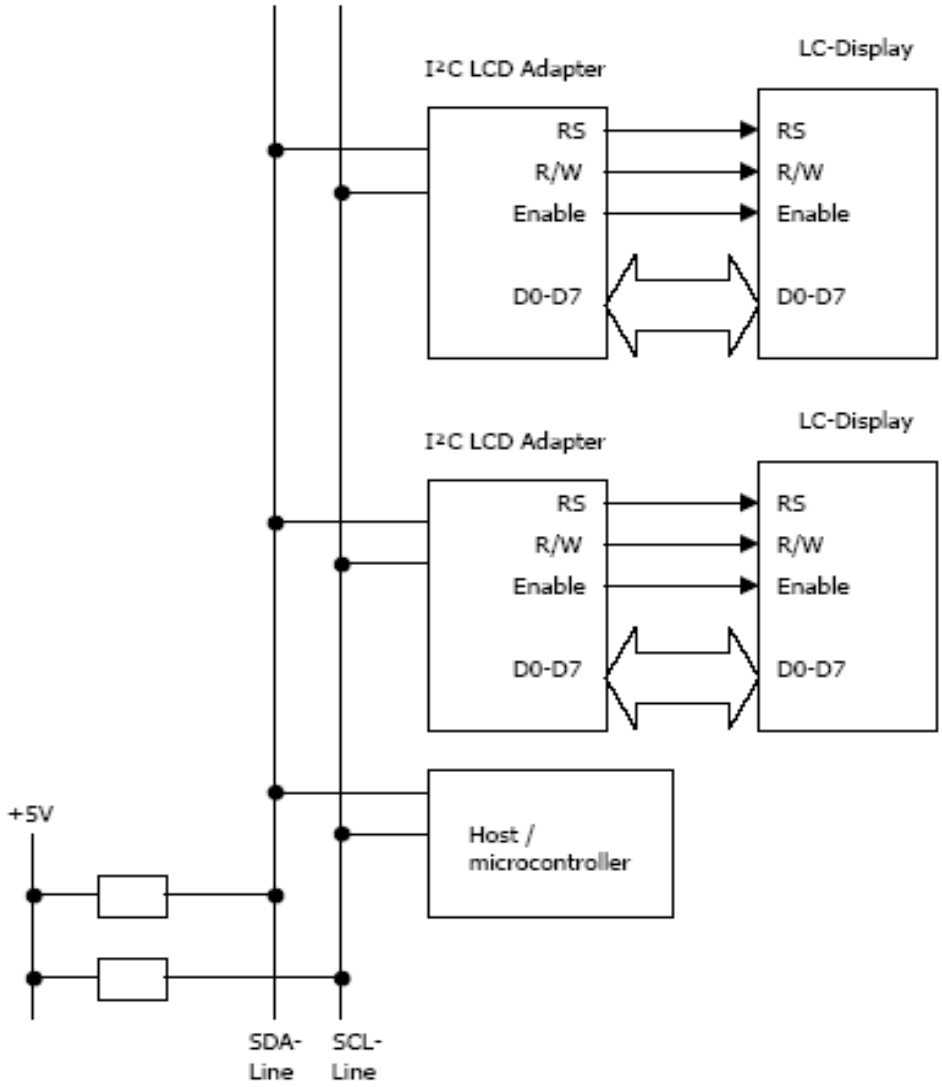
OUT Connektor V1020 (to Display):

Pin		Description
1	Vss	0V Ground
2	Vdd	5V Supply Voltage for Logic
3	Vo	Bias for LCD (positiv)
4	RS	H: Data signal, L: Instruktion signal
5	R/W	H: Read mode, L: Write mode
6	E	Chip enable signal
7	DB0	Data Bit 0
8	DB1	Data Bit 1
9	DB2	Data Bit 2
10	DB3	Data Bit 3
11	DB4	Data Bit 4
12	DB5	Data Bit 5
13	DB6	Data Bit 6
14	DB7	Data Bit 7
15	BLA	Backlight Anode
16	BLK	Backlight Kathode

3. Communication via I²C-Bus

3.1 I²C-Bus configuration

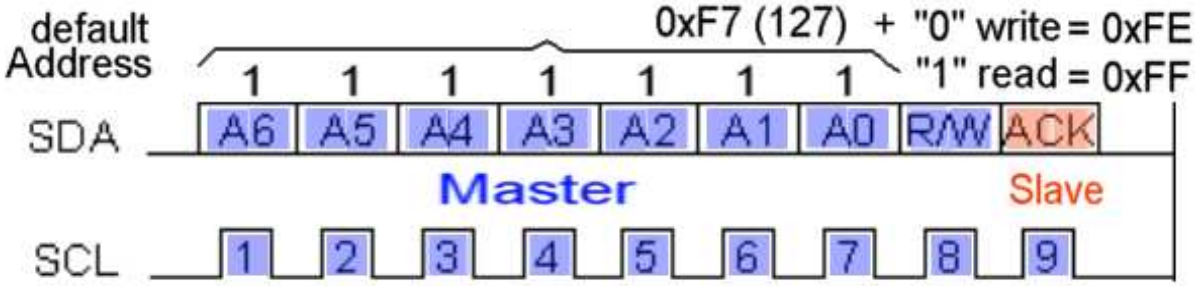
I²C-Bus uses two wires (SDA and SCL) to transfer information between devices connected to the bus. Each device connected to the bus is software addressable by a
unique addresse and simple master/slave relation ship exists at all times.



Example of an I²C-bus configuration

3.2 Address of the circuit

All I2C addresses are either 7 bits or 10 bits. The use of 10 bit addresses is rare and is not covered here. All of this modules will have 7 bit addresses. This means that you can have up to 127 devices on the I2C bus, since a 7bit number can be from 1 to 127 because, “0” is reserved for General Call. When sending out the 7 bit address, we still always send 8 bits. The extra bit is used to inform the slave if the master is writing to it or reading from it. If the bit is zero are master is writing to the slave. If the bit is 1 the master is reading from the slave. The 7 bit address is placed in the upper 7 bits of the byte and the Read/Write (R/W) bit is in the LSB (Least Significant Bit).



The placement of the 7 bit address in the upper 7 bits of the byte.

3.3 The I²C Bus specification

How fast?

The standard clock (SCL) speed for I2C up to 100KHz. Philips do define faster speeds: Fast mode, which is up to 400KHz and High Speed mode which is up to 3.4MHz. All of our module are designed to work at up to 100KHz. Module is tested up to 400kHz but this needs a small delay of a few uS between each byte transferred.

3.3.1 Start and Stop conditions

Within the procedure of the I²C bus, unique situations arise which are defined as START (S) and STOP (P) conditions.

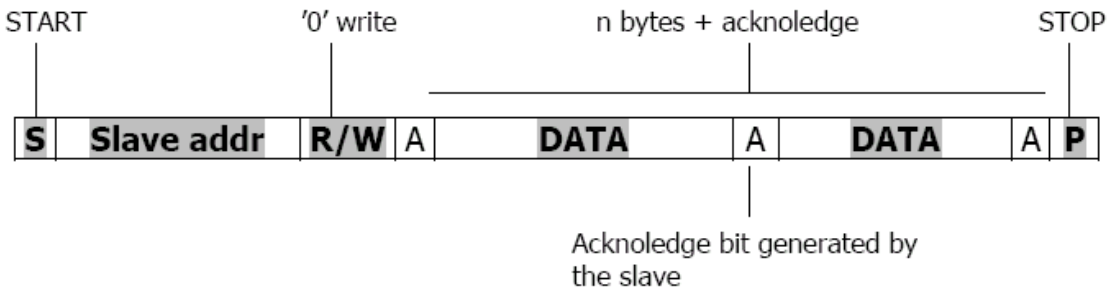
Start A HIGH to LOW transtion on the SDA line while SCL is HIGH defines a condition.

Stop A LOW to HIGH transtion on the SDA line while SCL is HIGH defines a condition.

Start and Stop conditions are always generated by the master. The bus is considered
to be busy after a START condition. The bus is considered to be free again a certain time after a STOP condition.

3.3.2 Write data to I²C LCD Controller

Writing data to a device on the I²C-Bus is agreed as follow:
After the START condition, a slave address is sent. This address is 7 bits long followed by an eight bit which is a data direction bit (‘0’=write, ‘1’=read) and an acknoledge bit. Any further data byte is followed by an acknoledge bit. A data transfer is always term inated by a STOP condition.

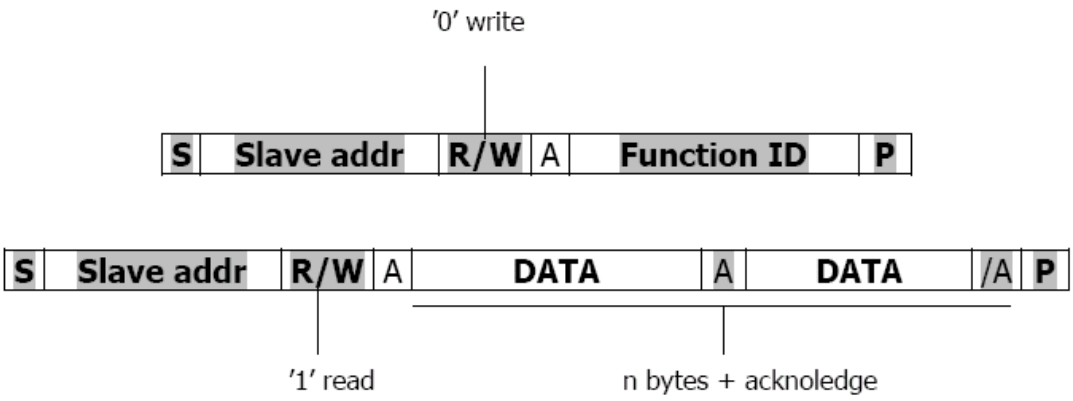


3.3.3 Read data from I²C LCD Controller

Reading data from a device on the I²C-Bus is agreed as follow and it consists of two sequences:

1. The first sequence consists of two bytes of data. The first byte is the slave address and write bit. The second consists of the function ID which reads the data from the LCD CG- or GGRAM and stores them in an internal register of the I²C LCD Controller.

2. The second sequence consists of the slave address and the read bit. Then the master can read the data. After each byte of data the master has to acknoledge correct data reception. The last byte is not to acknoledge by the master and therfore the slave knows the end of transmission.



For further information about the I²C-Bus specification please visit philips semiconductors on the internet at http://www.semiconductors.philips.com and download the datasheet.

4. Commands description

4.1 Command Overview

Command	Funktion	Command Byte
ClrDisplay	Clears LC-Display	0x61
ReturnHome	Sets cursor address / display to zero	0x62
SetCursor	Programes a new cursor address	0x63
CharToLCD	Writes a character on the LCD	0x64
CMDToLCD	Sends a command to the LCD	0x65
GetCursorAdr	Returns aktualcurso address	0x66
GetCharAtCur	Returns character at aktual cursor address	0x67
ReadRAM	Returns character at actual address of CG- or DDRAM	0x68
DisplayON_OFF	Sets display´s parameters	0x69
Shift	Sends shift command to LCD	0x6A
SetEntryMode	Sets display´s Entry Mode	0x6B
SetFunktion	Sets display´s funktion (e.g. 2 or 1 lines)	0x6C
SetCGRAMAdr	Sets CG-RAM address	0x6D
SetDDRAMAdr	Sets DD-RAM address	0x6E
WriteSlaveAdr	Programmes and stores slave address	0x6F
LCDType	Sets the Display type (e.g. 4x20 or 2x16..)	0x70
ShowLCDIni	Show a test-sequenz / Ini status after the restart	0x71
SaveLCDIni	Saves LCD parameters into EEprom	0x72
WriteUserDefChr	Writes a user defined character to LCD	0x73
SaveUserDefChr	Saves a user character into EEprom	0x74
WriteString	Writes a string on the LCD at the aktual cursor address	0x75
WriteIndexString	Writes a string on the LCD on the defined position	0x76
Delete	Deletes characters from LCD	0x77
Copy	Copies characters into a temporary storage	0x78
Paste	Inserts characters to LCD (from temporary storage) to the akt. Position of cursor	0x79
SetCursorIndex	Define the position of cursor at row (1..4) and column (1..20)	0x7A
8bitValueToLCD	Writes 8bit hex value to the LCD right-jusified	0x7B
16bitValueToLCD	Writes 16bit hex value to the LCD right-jusified	0x7C
BacklightZone	Set the min/max value for brightness of the background lights	0x7D
DimmBacklight	Dimm backlight from minValue to maxValue and back	0x7E
ContrastZone	Set the min/max value of the Contrast	0x7F
DimmContrast	Dimm contrast of LCD from minValue to maxValue and back	0x80
SaveUserPage	Saves a user page (1..10) into EEprom	0x81
WriteUserPage	Writes a user page (1..10) to the LCD	0x82
WriteUserRow	Writes a user Row (1..20) to the LCD	0x83

4.2 Standard HD44780 Commands

4.2.1 ClrDisplay

In order to clear the LC-Display issue the folowing command by the master.

ClrDisplay command (0x61)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	ClrDisplay	0	1	1	0	0	0	0	1

4.2.2 ReturnHome

This command is provided to the circuit to return cursor / display home

ReturnHome command (0x62)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	ReturnHome	0	1	1	0	0	0	1	0

4.2.3 SetCursor

In order to set the cursor address issue the following command

SetCursor command (0x63)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	SetCursor	0	1	1	0	0	0	1	1
2	Cursor address	cursor address (0..127)							

4.2.4 CharToLCD

To write a character on the LCD issue the following command

CharToLCD command (0x64)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	CharToLCD	0	1	1	0	0	1	0	0
2	Character	a byte as the standard character pattern of the LCD							

4.2.5 CMDToLCD

Some times it is necessary to a single command to the LCD. In order to send a command as agreed for the LCD issue this command.

CMDToLCD command (0x65)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	CMDToLCD	0	1	1	0	0	1	0	1
2	Command	a byte as the standard HD44780 command							

4.2.6 GetCursorAdr

This function returns two bytes: 1. the slave address of the selected device and 2. the actual cursor address.

GetCursorAdr command (0x66)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	GetCursorAdr	0	1	1	0	0	1	1	0

GetCursorAdr Response									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address (W)	AD6	AD5	AD4	AD3	AD2	AD1	AD0	1
1	Slave addess	Slave address of the selected device							
2	Cursor Address	Actual cursor address							

4.2.7 GetCharAtAdr

This function returns two bytes: 1. the slave address of the selected device and 2. the character at the actual cursor address.

GetCharAtAdr command (0x67)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	GetCharAtAdr	0	1	1	0	0	1	1	1

GetCharAtAdr Response									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address (W)	AD6	AD5	AD4	AD3	AD2	AD1	AD0	1
1	Slave addess	Slave address of the selected device							
2	Char at address	Character at actual cursor address							

4.2.8 ReadRAM

This function returns two bytes: 1. the slave address of the selected device and 2. a data byte from the selected CG- or DDRAM address of the LC-Display. In order to read a RAM content select first the address by using the functions SetCGRAMAdr or SetDDRAMAdr.

ReadRAM command (0x68)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	ReadRAM	0	1	1	0	1	0	0	0

ReadRAM Response									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address (W)	AD6	AD5	AD4	AD3	AD2	AD1	AD0	1
1	Slave addess	Slave address of the selected device							
2	CG- or DDRAM	Data byte from the selected RAM address							

4.2.9 DisplayON_OFF

In order to change display's paramters issue this command and write the paramters as a byte to the selected device.

DisplayON_OFF command (0x69)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	DisplayON_OFF	0	1	1	0	1	0	0	1
2	Config. Byte	X	X	X	X	X	**D**	**C**	**B**

Bit	L	Function	comment
D	1	Display ON	
	0	Display OFF	
C	1	Cursor ON	
	0	Cursor OFF	
B	1	Blinking ON	
	0	Blinking OFF	
X	X	n.a.	Do not care

4.2.10 Shift

For shifting display or moving the cursor use the following command. In order to configure the direction set or clear the R/L.

Shit command (0x6A)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	Shift	0	1	1	0	1	0	1	0
2	Shift Byte	X	X	X	X	X	X	**S/C**	**R/L**

Bit	L	Function	comment
S/C	1	Display shift	
	0	Cursor shift	
R/L	1	Right shift	
	0	Left shift	
X	X	n.a.	Do not care

4.2.11 SetEntryMode

In order to change the Entry Mode of the display issue this command and write the config. byte to the selected device.

SetEntryMode command (0x6B)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	SetEntryMode	0	1	1	0	1	0	1	1
2	Config. Byte	X	X	X	X	X	X	**I/D**	**S**

Bit	L	Function	comment
I/D	1	Increase (+1)	
	0	Decrease (-1)	
S	1	Display is shifted	
	0	Display is not shifted	
X	X	n.a.	Do not care

4.2.12 SetFunction

In order to change display's paramters issue this command and write the paramters as a byte to the selected device. After a new startup the LCD will be initialised automatically with the new parameters.

SetFunktion command (0x6C)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	SetFunktion	0	1	1	0	1	1	0	0
2	Config. Byte	X	X	X	X	X	X	**N**	**F**

Bit	L	Function	comment
N	1	2 line display	
	0	1 line display	
F	1	5 x 10 dots	
	0	5 x 7 dots	
X	X	n.a.	Do not care

4.2.13 SetCGRAMAdr

In order to read a CGRAM address select first the address of the CGRAM with the following command. After the CGRAM address is successfully selected, the RAM content could be read.

SetCGRAMAdr command (0x6D)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	SetCGRAMAdr	0	1	1	0	1	1	0	1
2	CGRAM address	X	X	The CGRAM address to select					

4.2.14 SetDDRAMAdr

In order to read a GGRAM address select first the address of the DDRAM with the following command. After the DDRAM address is successfully selected, the RAM content could be read.

SetDDRAMAdr command (0x6E)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	SetDDRAMAdr	0	1	1	0	1	1	1	0
2	DDRAM address	The DDRAM address to select							

4.3 LCD Control Commands

4.3.1 WriteSlvAdr

In order to change the slave address issue this command and then send the new slave address.

WriteSlvAdr command (0x6F)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	CurrentSlave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	WriteSlvAdr	0	1	1	0	1	1	1	1
2	New Slave Address	X	7 bits of Slave address						

X	Do not care

4.3.2 LCDType

In order to select the type of your LC-Display issue the folowing command by the master.

LCDTyp command (0x70)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	LCDTyp	0	1	1	1	0	0	0	0
2	Type	X	X	X	X	**DZ1**	**DZ0**	**ZL1**	**ZL0**

Bit		**Type**	**comment**
DZ1	**DZ0**		
0	0	1 Line	
0	1	2 Line	
1	0	4 Line	
1	1	reserved	

Bit		**Type**	**comment**
ZL1	**ZL0**		
0	0	16 Char by Line	
0	1	20 Char by Line	
1	0	24 Char by Line	
1	1	40 Char by Line	

4.3.3 ShowLCDIni

Use this order to have a test sequence and/or the stored users reported to LCDIni after a restart or powerup.

ShowLCDIni command (0x71)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	ShowLCDIni	0	1	1	1	0	0	0	1
2	Show	X	X	X	X	**I**	**T**	X	X

Bit	L	Function	comment
I	1	Show Ini Status ON	
	0	Show Ini Status OFF	
T	1	Show Test-Sequenz ON	
	0	Show Test-Sequenz OFF	
X	X	n.a.	Do not care

4.3.4 SaveLCDIni

This command is provided by the master to save the configuration bytes for the LCD.
The following Bytes will be saved:

1. SetFunction
2. SetEntryMode
3. DisplayON_OFF
4. LCDTyp
5. ShowLCDIni
6. Backlight
7. Contrast

On startups the LCD will be initialised with these settings.

SaveLCDIni command (0x72)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	SaveLCDIni	0	1	1	1	0	0	1	0

4.4 LCD Instruction Commands

4.4.1 WriteUserDefChr

This command is provided by the master to create a user defined characters and to load them to the CGRAM of a LCD. A LC-Display allows 8 users defined characters to
be loaded to the CGRAM. These are the first 8 characters (0x00–0x07) in the standard character pattern of a LCD.

The character number ist a number between 0x00 and 0x07.

WriteUserDefChar command (0x73)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	WriteUserDefChar	0	1	1	1	0	0	1	1
2	Charater Number	X	X	X	X	X	Char.Nr (1 - 8)		
3	Character Byte 0	X	X	X	5 dots (4:0)				
4	Character Byte 1	X	X	X	5 dots (4:0)				
5	Character Byte 2	X	X	X	5 dots (4:0)				
6	Character Byte 3	X	X	X	5 dots (4:0)				
7	Character Byte 4	X	X	X	5 dots (4:0)				
8	Character Byte 5	X	X	X	5 dots (4:0)				
9	Character Byte 6	X	X	X	5 dots (4:0)				
10	Character Byte 7	X	X	X	5 dots (4:0)				

X	Do not care

4.4.2 SaveUserDefChr

After loading a user defined character to the LCD, the character could be saved, and after a startup the 8 characters will be automatically loaded to the CGRAM. In order to save a character issue this command and send the character number (0x00 – 0x07) to the controller.

SaveUserDefChar command (0x74)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	SaveUserDefChar	0	1	1	1	0	1	0	0
2	Charater Number	X	X	X	X	X	Char.Nr (1 - 8)		

X	Do not care

4.4.3 WriteString

To write a String (1 to 40 characters) on the LCD issue the following command

WriteString command (0x75)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	WriteSring	0	1	1	1	0	1	0	1
2	Character (1)	a byte as the standard character pattern of the LCD							
...	...	...							
42	Character (42)	a byte as the standard character pattern of the LCD							

4.4.4 WriteIndexString

To write a indexed String (1 to 40* characters) on the LCD issue the following command

WriteIndexString command (0x76)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	WriteIndexString	0	1	1	1	0	1	0	0
2	Row	Number of row (1 - 4*)							
3	Column	Start-Index on the selected row (1 - 40*)							
4	Character (1)	a byte as the standard character pattern of the LCD							
...	...	...							
44	Character (42)	a byte as the standard character pattern of the LCD							

* depending on display type

4.4.5 Delete

In order to delete a character, two characters or a complete line proceed as follow:

Delete command (0x77)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	Delete	0	1	1	1	0	1	1	1
2	Line	Number of line (1 - 4*)							
3	Index	Start-Index on the selected line (1 - 40*)							
4	Length	Length, characters shold be deleted (1 - 40*)							

4.4.6 CopyToClipBoard

This function Copies a set of characters (1-40*) to a temporary storage, which could be written to the LCD later.

CopyToClipBoard command (0x78)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	CopyToClipBoard	0	1	1	1	1	0	0	0
2	Line	Number of line (1 - 4*)							
3	Index	Start-Index on the selected line (1 - 40*)							
4	Length	Length, characters shold be deleted (1 - 40*)							

* depending on display type

4.4.7 Paste

This command is provided by the master to insert the copied characters in the temporary storage on the LCD

Paste command (0x79)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	Paste	0	1	1	1	1	0	0	1

4.4.8 SetCursorIndex

In order this command to index set the cursor in Display issue the following command.

SetCursorIndex command (0x7A)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	WriteIndexString	0	1	1	1	1	0	1	0
2	Row	Number of row (1 - 4*)							
3	Column	Start-Index on the selected row (1 - 40*)							

* depending on display type

4.4.9 8bitValueToLCD

Use this order to display a 8bit value to the display. The controller change autonomous value in ASCII and clears the left-hand zeros. You can define position after decimal point with the first data byte in instruction. The value is then written in LCD **right adjusified** at the actuality cursor address and need 4digit to display. The cursor address isn't changed at this.

8bitValueToLCD command (0x7B)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	ValueToLCD2	0	1	1	1	1	0	1	1
2	pos. after dec. point	X	X	X	X	X	X	digit (1..2)	
3	8bit Value	8bit Value (0..255)							

digit	function	example
0	no decimal point	**123**
1	one digit after DP	**12.3**
2	two digit after DP	**1.23**

4.4.10 16bitValueToLCD

Use this order to display a 16bit value to the display. The controller change autonomous value in ASCII and clears the left-hand zeros. You can define position after decimal point with the first data byte in instruction. The value is then written in LCD **right adjusified** at the actuality cursor address and need 6digit to display. The cursor address isn't changed at this.

16bitValueToLCD command (0x7C)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	16bitValueToLCD	0	1	1	1	1	1	0	0
2	pos. after dec. point	X	X	X	X	X	digit (1..4)		
3	16bit Value	16bit Value (low Byte)							
4	16bit Value	16bit Value (high Byte)							

digit	**function**	**example**
0	no decimal point	**12345**
1	one digit after DP	**1234.5**
2	two digit after DP	**123.45**
3	tree digit after DP	**12.345**
4	four digit after DP	**1.2345**

4.4.11 BacklightZone

Use this order to adjust the utilizable area of the background lights. For the start the maximum value of the background lights is used. The user defined attitude of the background lights also is stored with the SaveLCDIni command.

BacklightZone command (0x7D)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	BacklightZone	0	1	1	1	1	1	0	1
2	MIN Brightness	min Value (0 - 255)							
3	MAX Brightness	max Value (0 - 255)							

4.4.12 DimmBacklight

One can use this order to the automatic dim of the brightness. The speed of the diming Backlight is fixed with the DB1..0 bits. With the IO bit the automatic dimmer is the manner fixed.

DimmBacklight command (0x7E)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	DimmBacklight	0	1	1	1	1	1	1	0
2	Config	X	X	X	X	X	**DB1**	**DB2**	**IO**

Bit		**Type**	**comment**
DB1	**DB0**		
0	0	no dimming	
0	1	dimming fast	
1	0	dimming midle	
1	1	dimming slow	

Bit	**L**	**Function**	**comment**
IO	1	Backlight On / dimming IN	
	0	Backlight Off / dimming OUT	
X	X	n.a.	Do not care

4.4.13 ContrastZone

Use this order to adjust the utilizable area of the contrast. For the start the maximum value of the contrast from LCD is used. The user defined attitude of the contrast also is stored with the SaveLCDIni command

ContrastZone command (0x7F)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	ContrastZone	0	1	1	1	1	1	1	1
2	MIN Contrast	min Value (0 - 255)							
3	MAX Contrast	max Value (0 - 255)							

4.4.14 DimmContrast

One can use this order to the automatic dim of the contrast. The speed of the diming contrast is fixed with the DB1..0 bits. With the IO bit the automatic dimmer is the manner fixed.

DimmBacklight command (0x80)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	DimmBacklight	1	0	0	0	0	0	0	0
2	Config	X	X	X	X	X	**DB1**	**DB2**	**IO**

Bit		**Type**	**comment**
DB1	**DB0**		
0	0	no dimming	
0	1	dimming fast	
1	0	dimming midle	
1	1	dimming slow	

Bit	**L**	**Function**	**comment**
IO	1	Contrast On / dimming IN	
	0	Contrast Off / dimming OUT	
X	X	n.a.	Do not care

4.4.15 SaveUserPage

Use this order around 40 signs or one display page (2x20) to save into the EEPROM of the driver.

StoreUserPage command (0x81)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	StoreUserPage	1	0	0	0	0	0	0	1
2	Page	number of page (1 - 10)							
3	Character (1)	a byte as the standard character pattern of the LCD							
...	...	...							
23	Character (20)	a byte as the standard character pattern of the LCD							

4.4.16 WriteUserPage

Use this order to send one stored display page on the display.

WriteUserPage command (0x82)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	WriteUserPage	1	0	0	0	0	0	1	0
2	Page	number of page to write to Display (1 - 10)							

4.4.17 WriteUserLine

Use this order to send one stored display Line on first to fourth row of the display.

WriteUserLine command (0x83)									
Byte	**content**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	Slave Address	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0
1	WriteUserLine	1	0	0	0	0	0	1	1
2	LCD-Row	number of LCD-row to display the stored Line (1 - 4*)							
3	stored Line	number of stored Line to write in Display (1 - 20)							

* depending on display type

Construction of the UserWrite / UserSave commands specific to 2x20 LCDs

Nr	Number of Line / Page to write to Display	Display Position WritePage	Display Position WriteLine
1	20 character UserText Line 1 / Page 1	Row 1	**independed of selected Line of LCD-Row**
2	20 character UserText Line 2 / Page 1	Row 2	
3	20 character UserText Line 3 / Page 2	Row 1	
4	20 character UserText Line 4 / Page 2	Row 2	
5	20 character UserText Line 5 / Page 3	Row 1	
6	20 character UserText Line 6 / Page 3	Row 2	
7	20 character UserText Line 7 / Page 4	Row 1	
8	20 character UserText Line 8 / Page 4	Row 2	
9	20 character UserText Line 9 / Page 5	Row 1	
10	20 character UserText Line 10 / Page 5	Row 2	
11	20 character UserText Line 11 / Page 6	Row 1	
12	20 character UserText Line 12 / Page 6	Row 2	
13	20 character UserText Line 13 / Page 7	Row 1	
14	20 character UserText Line 14 / Page 7	Row 2	
15	20 character UserText Line 15 / Page 8	Row 1	
16	20 character UserText Line 16 / Page 8	Row 2	
17	20 character UserText Line 17 / Page 9	Row 1	
18	20 character UserText Line 18 / Page 9	Row 2	
19	20 character UserText Line 19 / Page 10	Row 1	
20	20 character UserText Line 20 / Page 10	Row 2	

Notize:

Notize: