

LCDlogic was designed to be very simple and effective solution for using Liquid Crystal Displays based on the HD44780 controller (or equivalent controllers).

Only one UART line is necessary to configure, interface your display. You also need 2 wires for power supply and ground.

All the settings are adjustable via software commands and are saved automatically when changed. Settings are stored in internal EEPROM and loaded during power up.

Interface Specifications

Default Communication Settings :

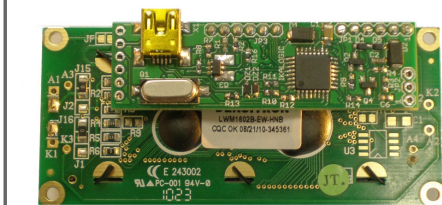
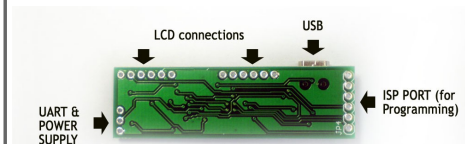
- BAUD Rate : 9600¹
- 8bits per transfer
- 1 Start bit, 1 Stop bit
- No parity bit

The LCDlogic is controlled using actual ASCII characters. That is, if you send via the UART interface the ASCII character 'i' (0x69) to LCDlogic, an 'i' will be displayed on the LCD and the cursor position will be incremented by 1.

The only exception is the character (0x11) which is used as command frame header. A command frame is composed of this character followed by a command identifier and a parameter if it is necessary.

For easy C language integration, a header file is available on our web site including definitions for command identifiers and default parameters. To download it follow the link below: www.ikalogic.com/LCDlogic/header.h

IMPORTANT: Before connecting LCD Logic to an LCD screen, It's the user responsibility to make sure it is compatible with the LCD's operating voltages and that its controller is an HD44780 or any equivalent. If you have doubts about an LCD, you may contact us with the reference of your LCD.



¹ If you change the baudrate and save it, the new one will be effective at the next power up

Table Of Content

Electrical Specifications and connections.....	3
Configuration.....	4
1. Mode Changing.....	5
2. BackLight control.....	5
3. Contrast control	6
LCD Screen size Setup.....	6
Cursor management.....	6
1. Setting Cursor Position	6
2. Moving Cursor position.....	7
3. Cursor Display.....	7
Power Save Mode.....	7
Reinitialize and Load Default Parameters	8
.....	8
BAUD Rate definition (UART).....	8
Defining your own SplashScreen.....	8
LCD Direct Command.....	9
Custom Characters.....	9
1. LCDlogic FirmWare.....	9
2. Creating Your Own Character.....	10
3. Bargraphs.....	11
Mechanical specifications.....	12
Getting more help.....	12



Electrical Specifications and connections

The minimum power supply voltage is 3.6 VDC and the absolute maximum is 5.5VDC without causing micro controller or LCD damages.



LCD Logic Pinout

LCD logic Pins descriptions

Pin group	Pin name	Function	Comments
USER INTERFACE	VCC	Power supply	3.6V min, 5.5V max
	UART_IN	UART Input	UART Input from the micro controller
	GND	GND	Connect to the GND of the controller
LCD INTERFACE	LGND	LCD GND connection	To be directly connected to your 16-PINs LCD
	LVCC	LCD VCC connection	
	VO	Output Voltage	
	RS	Register Select	
	R/W	Read/write	
	E	Enable	
	D4 to D7	Data	
	BLA	Anode for Back-light	
BLK	Cathode for back-light		
Programming Interface	GND	GND	Used for programming the micro-controller of the LCD Logic. Do not interface those pins unless you need to reprogram the LCD Logic controller. For advanced users only.
	VCC	Power supply	
	RST	SPI Interface PINs	
	SCK		
	MISO		
	MOSI		



Configuration

The LCDlogic firmware allows you to configure each parameter via UART command frames. A command frame is composed of three bytes:

- The command header: `CMD_CHAR` [0x11]
- The Command code from the table below
- A parameter corresponding to the command. Depending on the command, it may not be necessary, so a null parameter may be sent.

Command	ID in Library	HEX	Parameter
MOVE CURSOR TO LINE 0	<code>CMD_LINE_0</code>	0x00	$0 \rightarrow 20 / PAR_COLUMN_X$
MOVE CURSOR TO LINE 1	<code>CMD_LINE_1</code>	0x01	$0 \rightarrow 20 / PAR_COLUMN_X$
MOVE CURSOR TO LINE 2	<code>CMD_LINE_2</code>	0x02	$0 \rightarrow 20 / PAR_COLUMN_X$
MOVE CURSOR TO LINE 3	<code>CMD_LINE_3</code>	0x03	$0 \rightarrow 20 / PAR_COLUMN_X$
BACKLIGHT Control	<code>CMD_BACKLIGHT</code>	0x65	$0 \rightarrow 255 / PAR_BL_XX\%$
CONTRAST Control	<code>CMD_CONTRAST</code>	0x66	$0 \rightarrow 255$
LCD LINES NUMBER	<code>CMD_NB_LINE</code>	0x67	$1 \rightarrow 4$
LCD COLUMNS NUMBER	<code>CMD_NB_COLUMN</code>	0x68	$1 \rightarrow 20$
CLEAR SCREEN	<code>CMD_CLR_SCRN</code>	0x69	$0 / PAR_DEFAULT$
POWER SAVE MODE	<code>CMD_PWR_SV_MD</code>	0x6B	$0 / PAR_DEFAULT$
DIRECT MODE ON	<code>CMD_DRCT_MD_ON</code>	0x30	$0 / PAR_DEFAULT$
DIRECT MODE OFF (Standard)	<code>CMD_DRCT_MD_OFF</code>	0x31	$0 / PAR_DEFAULT$
CURSOR DISPLAY	<code>CMD_DSPL_CURSOR</code>	0x6C	$0 / PAR_DEFAULT$
SET RS	<code>CMD_RS_SET</code>	0x6D	LCD Command to send
CLEAR RS	<code>CMD_RS_CLEAR</code>	0x6E	LCD Command o send
SAVE SPLASHSCREEN	<code>CMD_SV_SPLSHSCRN</code>	0x72	$0 / PAR_DEFAULT$
RE-INIT with DEFAULT PARAMETERS	<code>CMD_LD_DEFAULT_PAR</code>	0x73	$0 / PAR_DEFAULT$
CHANGE BAUDRATE	<code>CMD_CHG_BR</code>	0x74	$1 \rightarrow 7 / PAR_BR_XXXX$
MOVE ONE RIGHT	<code>CMD_MOV_RIGHT</code>	0xC0	$0 / PAR_DEFAULT$
MOVE ONE LEFT	<code>CMD_MOV_LEFT</code>	0xC1	$0 / PAR_DEFAULT$
MOVE ONE UP	<code>CMD_MOV_UP</code>	0xC2	$0 / PAR_DEFAULT$
MOVE ONE DOWN	<code>CMD_MOV_DOWN</code>	0xC3	$0 / PAR_DEFAULT$

There are two main operating modes : The first one is the standard mode, it is used to display characters on an LCD. The second mode is the Direct Command Mode, which allows you to use the LCDlogic as a serial to parallel adaptor. In this mode one can send commands and data via the UART interface to the LCD PINs directly. The table bellow shows you all the LCDlogic capabilities and the modes where they are available:



STANDARD MODE	DIRECT COMMAND MODE
<i>SERIAL INTERFACE</i>	
<i>BACKLIGHT SOFTWARE CONTROL</i>	
<i>CONTRAST SOFTWARE CONTROL</i>	
<i>BAUDRATE CHANGE</i>	
<i>LCD SETUP</i>	
<i>CURSOR POSITIONNING</i>	<i>SET RS</i>
<i>EASY TO USE BARGRAPHS</i>	<i>CLEAR RS</i>
<i>CUSTOM CHARACTERS</i>	<i>CREATE YOUR OWN CHARACTER</i>
<i>AUTOMATIC DISPLAY</i>	

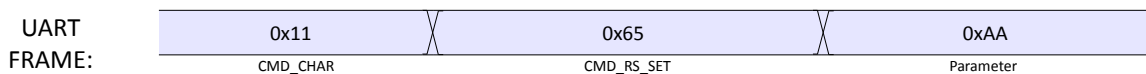
1. Mode Changing

To use Direct Command mode, first you have to send the command character CMD_CHAR [0x11] followed by the direct command identifier DIRECT_COMMAND_MODE [0x30]. At this point, the direct command mode is active, and you can use the SET_RS and CLEAR_RS commands to send data bytes to the LCD.

The SET_RS command is used to send a data byte while the RS² pin of the LCD is set.

The CLEAR_RS command is used to send a data byte while the RS pin of the LCD is cleared.

The following example UART frame sends the data byte 0xAA to the LCD while the RS pin is set:



2. BackLight control

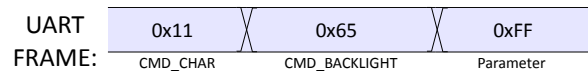
The LCDlogic controls the Backlight with a PWM signal via a MOSFET transistor.

By sending the command character CMD_CHAR (0x11) followed by the backlight identifier CMD_BACKLIGHT (0x65) and a value between 0 and 0xFF the Backlight value is set.

The brightness reduction allows a lower power consumption. The chosen backlight is stored automatically in LCDlogic (Non-Volatile Memory), so that it is re-loaded at next power up.

Example :

If you send on the UART CMD_CHAR [0x11] followed by the backlight command identifier and the parameter 0xFF you will get a full backlight.



2 RS: Register Select.



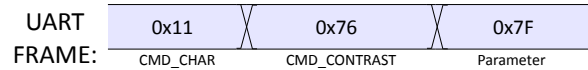
3. Contrast control

Contrast can be adjusted from 0 to 255, 255 being the higher contrast level.

The contrast level is automatically regulated, that is, if the power supply voltage changes the contrast control will adapt in consequence. Hence, very small contrast change is visually observable when power supply voltages changes (E.g. battery voltage dropping)

Example :

If you send on the UART the frame CMD_CHAR [0x11], CMD_CONTRAST [0x76], and the parameter (e.g. 0x7F), the contrast is set to an average value.



LCD Screen size Setup

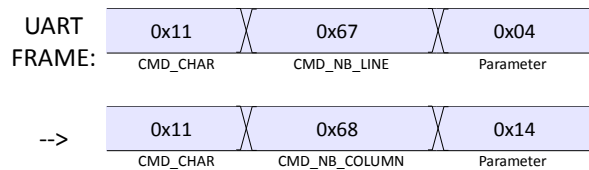
LCDlogic can be used for the following types of LCDs based on the very common HD44780:

- 1, 2 and 3 lines
- 8, 12, 16, and 20 characters wide

To setup your LCD Type, you need to specify its number of lines and its number of columns. For this, you have to use the command identifier CMD_NB_LINE [0x67] for lines and CMD_NB_COLUMN [0x68] for columns.

Example :

If you send on the UART the frame CMD_CHAR [0x11], CMD_NB_LINE [0x67], LINE_4 [0x04], followed by CMD_CHAR [0x11], CMD_NB_COLUMN [0x68], COLUMN_20 [0x14], the LCDlogic will be configured to work with a 4x20 LCD



Cursor management

1. Setting Cursor Position

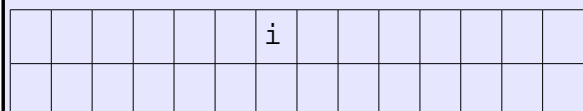
The cursor shows you where the next character sent on the UART line, will be displayed on the screen.

The LCDlogic allows you to position the cursor text (where the next character will be written).

If you want to position your cursor, you have to send the command character CMD_CHAR [0x11] followed by the line index (LINE_X) and the column (COLUMN_X) index.

Example :

If you send on the UART the frame CMD_CHAR [0x11], LINE_0 [0x00], COLUMN_6 [0x06] and the character 'i' your LCD will display :



2. Moving Cursor position

The cursor's position can be moved step by step in any of the 4 directions .

The following table describe the command corresponding to each direction. When using one of those commands, send 0x00 as a parameter.

Cursor Movement direction	>	<	^	v
Hex Code	0x70	0x71	0x72	0x73

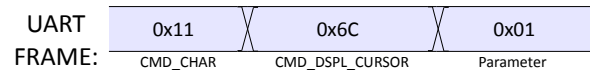
3. Cursor Display

You can display or hide the cursor position in the screen using the command identifier CMD_DSPL_CURSOR [0x6C].

If the parameter is set to 1, the cursor will be displayed. Otherwise the cursor will be hidden.

Example :

If you send on the UART the frame CMD_CHAR [0x11], CMD_DSPL_CURSOR [0x6C], and 0x01 as a parameter, the cursor will be displayed.



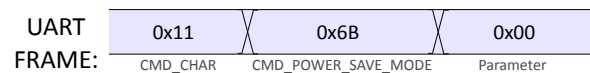
Power Save Mode

In order to reduce power consumption and if your application doesn't need the LCD you can put it in the power save mode. Your LCD will be switched off and LCDlogic will go in low power consumption mode. This is done using CMD_POWER_SAVE_MODE [0x6B]. In this mode the consumption of the LCDlogic goes down to **less than 1 μ A**.

To return to normal mode, you have to send any character or command frame on UART Line. Be advised that this character or this command will only be used for waking up LCDlogic and will not be displayed. Upon wake-up, the LCD will have the exact same configuration as before the shut down.

Example :

If you send on the UART the frame CMD_CHAR [0x11], CMD_POWER_SAVE_MODE 0x6B LCDlogic will go in low power consumption mode. If you want to go back in normal mode just send the same frame or any character



Reinitialize and Load Default Parameters

If you want to reinitialize LCDlogic with the default factory settings you just have to send on the UART line a classic 3 Bytes frame with the command `CMD_LD_DEFAULT_PAR - [0x73]`

Example :

If you sent on UART : `CMD_CHAR[0x11]`, `CMD_LD_DEFAULT_PAR [0x73]`, `[0x00]` you will load the following parameter :

*- Number Of Line : 2 Contrast : 0xDF
- Number Of Column : 16 BackLight : 0xFF
- BaudRate : 9600bd*

BAUD Rate definition (UART)

LCDLogic offers the possibility to change the serial Baud rate from 2400 bps to 38400kbps.

For that, the command character `CMD_CHAR (0x11)` should be sent on the UART line, followed by the identifier `CMD_CHANGE_BAUDRATE [0x74]` and the corresponding Baud rate identifier (`BR_ID`) from the next table

The default Baudrate is 9600 bps.

Baud rate can also be changed via the USB software.

BR/ bps	2400	4800	9600	14k4	19k2	28k8	38k4
BR_ID	0x01	0x02	0x03	0x04	0x05	0x06	0x07

Example :

Example: If you send on UART the frame `CMD_CHAR (0x11)`, `CMD_CHANGE_BAUDRATE (0x74)` and `(0x01)` as a parameter, the BaudRate will be set to 2400 bps.

Defining your own SplashScreen

The user has the possibility to define his own Splash screen and the corresponding display duration. This means that at the power up, the chosen message will be display during the chosen duration.

To adjust the splashscreen, first ensure that the LCD is displaying the desired splash screen content. Next, send the command character `CMD_CHAR [0x11]`, the command identifier `SAVE_SPLASH_SCREEN [0x75]` and the duration parameter as defined by this formula :

$$Time_{SplashScreen}(ms) = parameter \times 10ms$$

Note that when the splash screen is saved, the contrast and the backlight levels are also saved and are used each time the Splashscreen is displayed. When the splash screen is saved, the LCD content is

erased and the cursor goes on the initial position.

If you want to deactivate the splash screen just choose a null parameter time in the `SAVE_SPLASH_SCREEN` command.

Example :

After sending characters to display this screen:

			L	C	D	L	O	G	I	C		V	1	.	0	
W	W	W	.	I	K	A	L	O	G	I	C	.	C	O	M	

If you send `CMD_CHAR [0x11]`, `SAVE_SPLASH_SCREEN [0x72]`, `[0x64]` (Time/10ms), this screen will be erase and it will appear during 1s at the next power up.



LCD Direct Command

LCDlogic provides you the possibility to send some commands directly to your LCD. These commands can be sent with RS pin at logic level 0 or at 1.

If you want to send direct commands with RS set use the command identifier `CMD_SET_RS` [0x6D] and the value to send as a parameter.

If you want to send direct commands with RS clear use the command identifier `CMD_CLEAR_RS` [0x6E] and the value to send as a parameter.

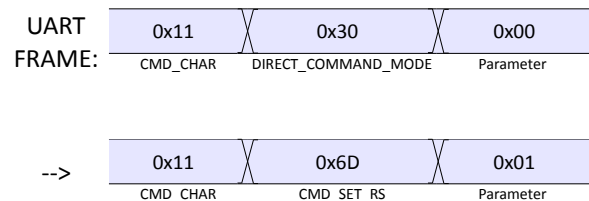
This example shows how to clear the screen in using HD44780 commands:

1- Choose the direct command mode

*Send on UART: `CMD_CHAR` [0x11],
`DIRECT_COMMAND_MODE` [0x30], [0x00]*

2 – Send LCD clear command mode with RS set

Send on UART: `CMD_CHAR` [0x11], `CMD_SET_RS` [0x6D], [0x01] (0x01 is the clear screen command as defined in the standard HD44780 commands table)



Custom Characters

1. LCDlogic FirmWare

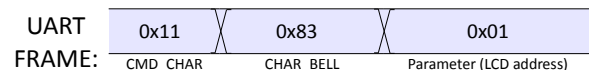
Each HD44780 based LCDs have its own character table. This table contains the standard ASCII characters from 0x20 to 0x7F. The others characters depend on the controller but you can configure the 8 first characters stored on the LCD table from 0x00 to 0x07.

The LCDlogic firmware allows you to define those 8 first characters easily and quickly using a set of pre-programmed icons. If you want to use it, you just have to send the command identifier corresponding to the character expected. Next, you have to choose the LCD address from 0x00 to 0x07 where you want to store the new character definition. **You have to wait minimum 100ms between two uses of this command.**




















Example :

If you choose to add a bell icon at the 0x01 LCD address just send `CMD_CHAR` [0x11], `CHAR_BELL` [0x83], 0x01(LCD address).

Next, when you send the 0x01 character on the UART line a bell will appear at the cursor position.



List of pre-programmed icons with their corresponding identifiers:

0x80	0x81	0x82	0x83	0x84	0x85	0x86	0x87	0x88	0x89	0x8A	0x8B	0x8C	0x90	0x91	0x92	0x93	0x94	0x95
																		

2. Creating Your Own Character

If you want to add your own character in the LCD, you have to use the Direct Command mode. Please refer to the example below :

Example: Defining a new custom character so that an 'I' is displayed when we send 0x01 on the UART.

1- Choose the direct command mode

Send **CMD_CHAR** [0x11],
DIRECT_COMMAND_MODE [0x30], [0x00]

2 – Choose the LCD memory address where you want to store your new character definition with the RS set.

For our example it's 0x48, because the start address is 0x40, and each character takes 8 bytes.

Send Send **CMD_CHAR** [0x11], **CMD_SET_RS** [0x6D], [0x48]

3 – Send the bytes which describe your custom character with the RS cleared.

In our example send :

- **CMD_CHAR** [0x11],**CMD_CLEAR_RS**[0x6D],[0x1F]

- **CMD_CHAR** [0x11],**CMD_CLEAR_RS**[0x6D],[0x04]

...

- **CMD_CHAR** [0x11],**CMD_CLEAR_RS**[0x6D],[0x04]

- **CMD_CHAR** [0x11],**CMD_CLEAR_RS**[0x6D],[0x1F]

Now, if you go in a standard mode(**CMD_CHAR** [0x11], **DIRECT_COMMAND_MODE** [0x31], follow by the character 0x01, a 'I' will be display.




3. Bargraphs








A bargraph is a visual indicator that is proportional to a given signal. There are a lot of applications where it's very interesting to use bargraphs and that's why we provide you with this solution.

When using LCDLogic bargraphs, the custom characters of your LCD (from 0x00 to 0x07) will be replaced.

To use horizontal bargraph use the command identifier CHAR_BARGRAPH_HORIZONTAL [0x96]. The characters downloaded in the LCD memory are present in the table below:

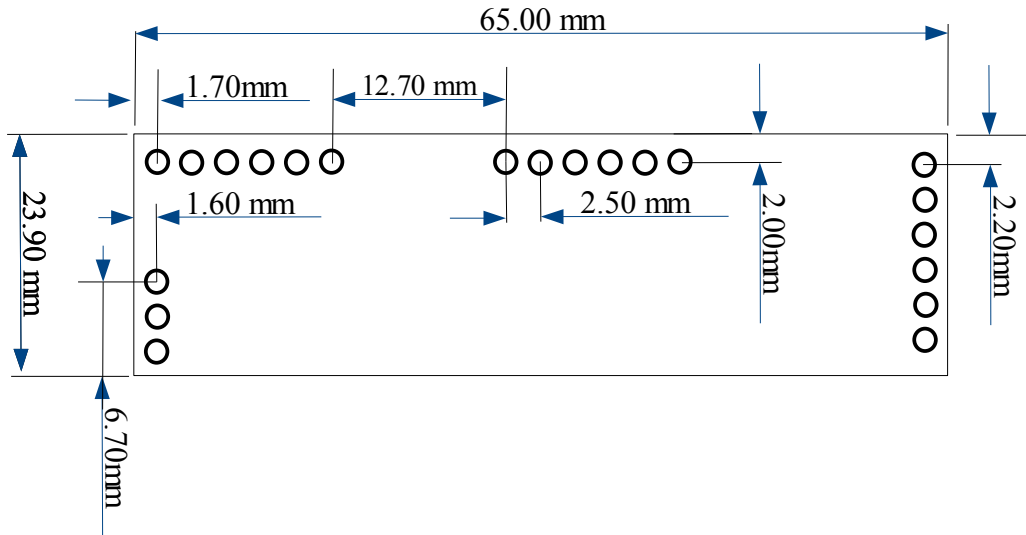
To use vertical bargraph use the command identifier CHAR_BARGRAPH_VERTICAL [0x97]. The characters downloaded in the LCD memory are present in the table below:

Char. LCDlogic	0x00	0x01	0x02	0x03	0x04	0x05
Car. Bargraph						

Char. LCDlogic	0x00	0x01	0x02	0x03	0x04	0x05	0x06	0x07
Car. Bargraph								



Mechanical specifications



Getting more help

For most up to date information and more specifications, please visit:

<http://www.ikalogic.com/lcdlogic.php>

