LCD Module with I2C / Serial Interface and Keypad Control
«LCD I2C/Serial»

User’s Guide

CONTENTS

1 INTRODUCTION. 3

2 MODULE CONNECTION. 3

2.1 I2C/Serial interface connector. 4

2.2 Keypad connector. 4

2.3 Communication interface. 4

2.4 Module connection to I2C / Serial bus. 5

3 COMMANDS. 6

3.1 Configuration Commands. 7

3.2 LCD Display Commands. 9

3.3 Bar Graphs and Custom Characters. 11

3.4 Keypad Commands. 14

Copyrights

Disclaimer of Liability

IMS, LLC is not responsible for special, incidental, or consequential damages resulting from any breach of warranty, or under any legal theory, including lost profits, downtime, goodwill, damage to or replacement of equipment or property, or any costs of recovering, reprogramming, or reproducing any data stored in or used with IMS products. IMS is also not responsible for any personal damage, including that to life and health, resulting from use of any of our products.

1 Introduction.

The I2C and serial display module provides easy operation of a standard character LCD display. The simple command structure allows text and bar graphs to be displayed on the screen. Provision is made for up to 8 user-defined characters. The module includes also the 4x4 matrix keypad control.
The LDC backlight and contrast may be adjusted under program control to compensate for differing lighting conditions and viewing angles.

The module supports 2 interfaces:

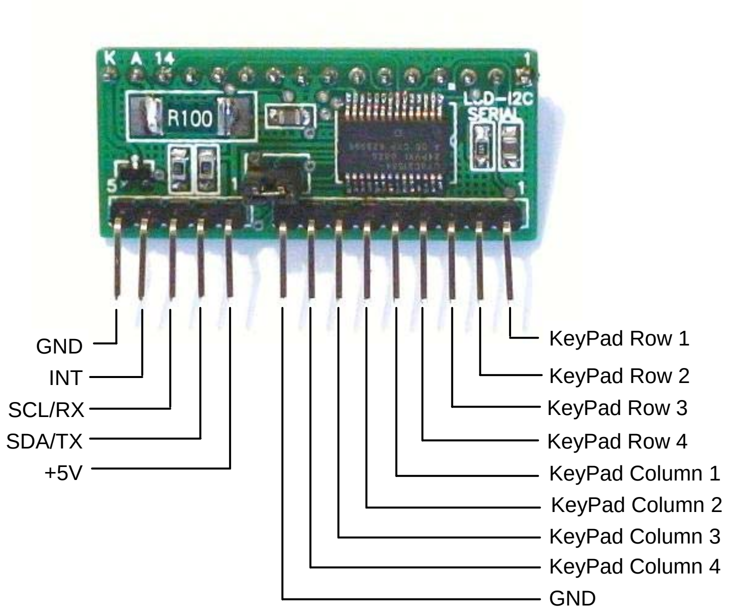
- I2C slave interface
- Serial TTL interface

The module has the following features:

- Communicate over I2C or Serial TTL interface with software controlled speed
- 80 Byte buffer for messages received via communication interface
- Up to 8 custom characters can be defined
- Built in commands for drawing Bar Graphs
- User configurable start up screen
- LCD Backlighting controlled via software, has 254 brightness levels
- LCD Contrast controlled via software, has 254 contrast levels
- Keypad encoder for a keypad up to 16 keys (4 rows by 4 columns)
- Save the basic settings in module internal EEPROM

2 Module connection.

The module has 3 connectors: LCD, I2C/Serial interface and keypad connector.



2.1 I2C/Serial interface connector.

Table 2.1 shows the connector pin assignments.

Table 2.1

Pin No.	Pin Name	Description
1	VDD	Supply voltage
2	SDA/TX	I2C SDA signal (TX – for serial interface)
3	SCL/RX	I2C SCL signal (RX – for serial interface)
4	INT	Key pressed signal
5	GND	Ground connection

2.2 Keypad connector.

Table 2.2 shows the connector pin assignments.

Table 2.2

Pin No.	Pin Name	Description
1	Row1	Keypad Row 1
2	Row2	Keypad Row 2
3	Row3	Keypad Row 3
4	Row4	Keypad Row 4
5	Col1	Keypad Col 1
6	Col2	Keypad Col 2
7	Col3	Keypad Col 3
8	Col4	Keypad Col 4
9	GND	Ground connection

2.3 Communication interface.

The device supports 2 interfaces:

- I2C slave interface
- Serial TTL interface

I2C Slave Interface
Industry standard Philips I^{2}C bus compatible interface.
Data rate 100 kbps.

Serial Interface
Baud Rates 2400, 4800, 9600 and 19200 bits per second (default 9600).
8 Bits per character
None Parity
1 Stop Bit
None Flow Control

Do NOT connect RS232 directly to the module. Use a MAX232 chip or equivalent to convert the RS232 levels to 5v.

The communication interface is determined by the states of the single jumper. When the jumper is present (factory default) the module is in serial mode. If the jumper is removed the module is in I2C mode. The mode jumper is only checked as part of the power-up sequence.

Once the interface has been changed, the module parameters will restored in DEFAULT values.

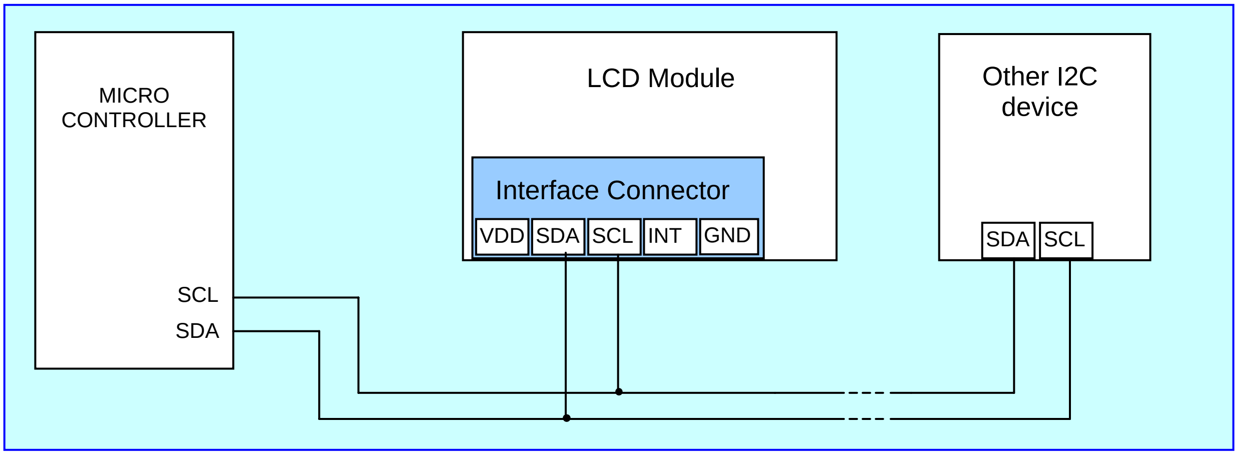
To restore the module default parameters (module address, baud rate, contrast, brightness):

1. Change the jumper state.
2. Power up.
3. Power down.
4. Change back the jumper state.
5. Power up.

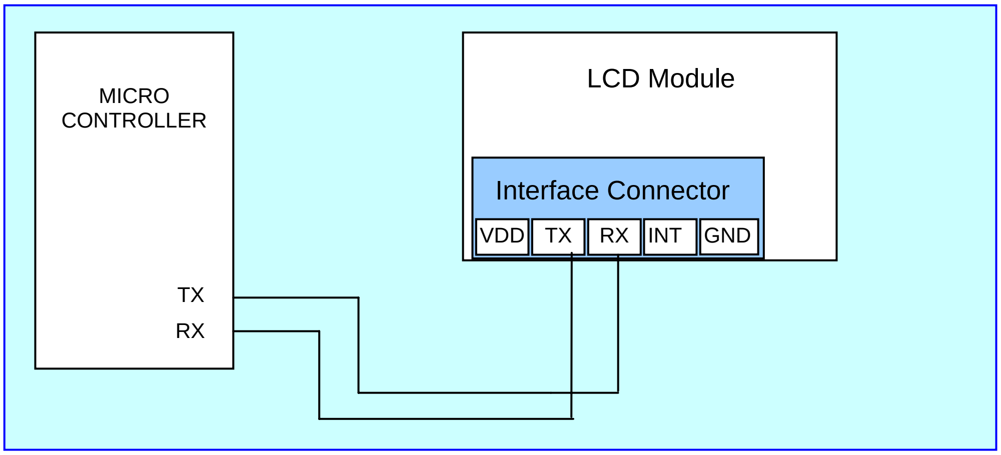
2.4 Module connection to I2C / Serial bus.

Each device must have its own unique address (ID). The address range is from ‘A’ to ‘Z’ (HEX from 0x41 to 0x5A). Default address shipped from the manufacture is ‘L’ (0x4C).
The address can be easily changed by send the command “Set the new device address”.

Module connecting to I2C bus.

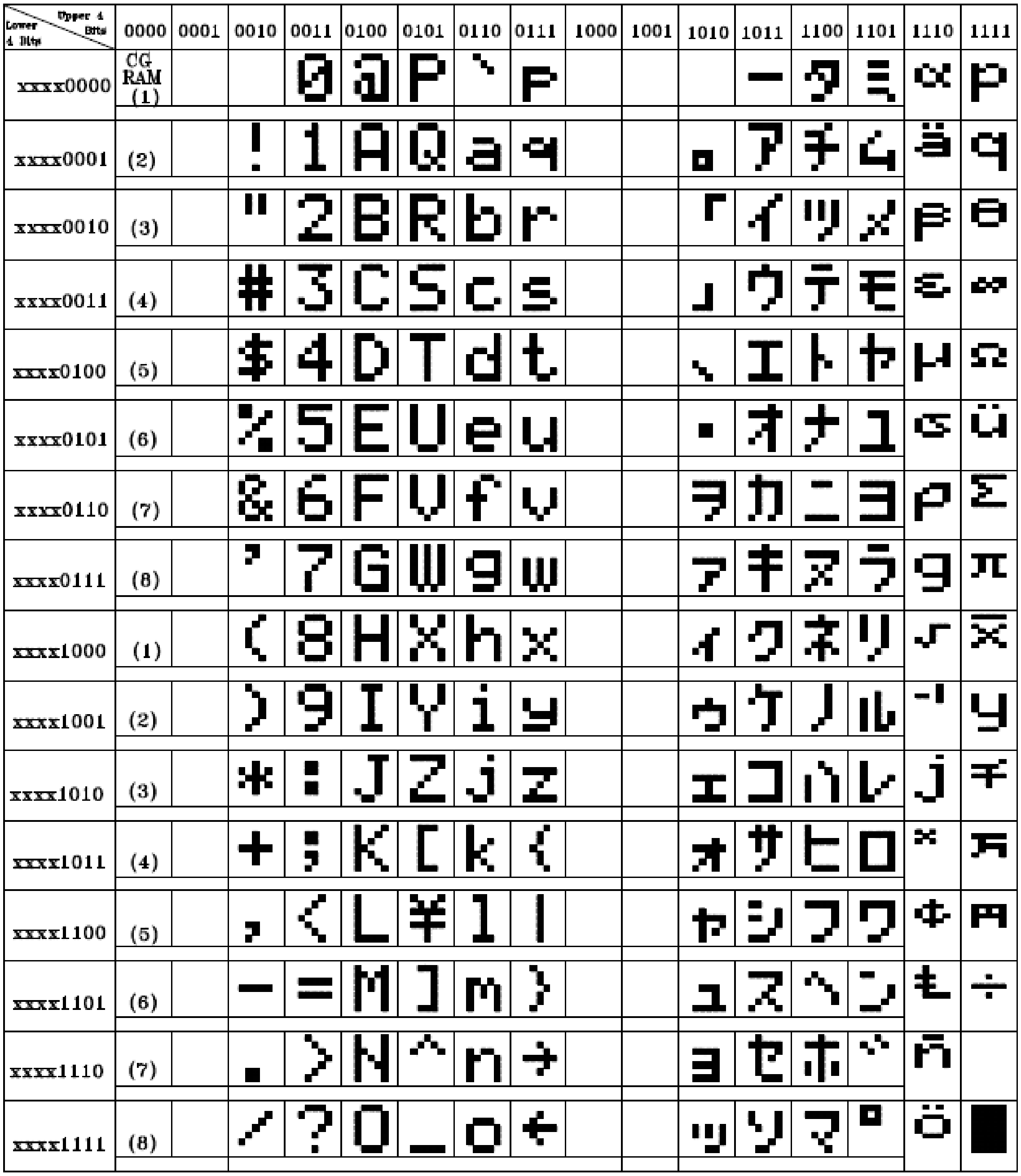


Module connecting to Serial TTL interface.



3 Commands.

The module is controlled using ASCII characters. The character decimal 254 (0xFE) is a command prefix. Any data sent to the LCD that is not prefixed by the command prefix (0xFE) will be displays on the LCD.
LCD includes a built-in 5 x 7 dot matrix font with the full range of ASCII characters plus a variety of extended characters, as shown in Figure 3-1.



Upper 4 Bits Lower 4 Bits	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
xxxx0000	CG RAM (1)			0	@	P	`	p				ｰ	ﾀ	ﾐ	α	p
xxxx0001	(2)		!	1	A	Q	a	q			｡	ｱ	ﾁ	ﾑ	ä	q
xxxx0010	(3)		"	2	B	R	b	r			｢	ｲ	ﾂ	ﾒ	β	θ
xxxx0011	(4)		#	3	C	S	c	s			｣	ｳ	ﾃ	ﾓ	ε	∞
xxxx0100	(5)		$	4	D	T	d	t			､	ｴ	ﾄ	ﾔ	μ	Ω
xxxx0101	(6)		%	5	E	U	e	u			･	ｵ	ﾅ	ﾕ	σ	ü
xxxx0110	(7)		&	6	F	V	f	v			ｦ	ｶ	ﾆ	ﾖ	ρ	Σ
xxxx0111	(8)		'	7	G	W	g	w			ｧ	ｷ	ﾇ	ﾗ	g	π
xxxx1000	(1)		(	8	H	X	h	x			ｨ	ｸ	ﾈ	ﾘ	√	x̄
xxxx1001	(2)		)	9	I	Y	i	y			ｩ	ｹ	ﾉ	ﾙ	⁻¹	y
xxxx1010	(3)		*	:	J	Z	j	z			ｪ	ｺ	ﾊ	ﾚ	j	千
xxxx1011	(4)		+	;	K	[	k	{			ｫ	ｻ	ﾋ	ﾛ	ˣ	万
xxxx1100	(5)		,	<	L	¥	l	|			ｬ	ｼ	ﾌ	ﾜ	¢	円
xxxx1101	(6)		-	=	M	]	m	}			ｭ	ｽ	ﾍ	ﾝ	£	÷
xxxx1110	(7)		.	>	N	^	n	→			ｮ	ｾ	ﾎ	ﾞ	ñ	
xxxx1111	(8)		/	?	O	_	o	←			ｯ	ｿ	ﾏ	ﾟ	ö	█

To display normal text, just enter its ASCII number, a number from 0x00 to 0x07 displays the user defined custom character, 0x20 to 0x7F displays the stand set of characters. And numbers from

0xA0 to 0xFD display characters and symbols those are factory-masked on the LCD controller and 0xFE is reserved for function command.

Command Summary

Prefix	Command	Parameter	Description
0xFE	0x00	null	No operation
0xFE	0x01	1 byte	Changing the I2C Slave Address
0xFE	0x02	1 byte	Changing BAUD Rate
0xFE	0x03	1 byte	Set Backlight Brightness
0xFE	0x04	1 byte	Set Contrast
0xFE	0x05	None	Save Splash/Startup Screen
0xFE	0x06	None	Display Firmware Version Number
0xFE	0x07	None	Display Serial Baud Rate
0xFE	0x08	None	Display I2C Address
0xFE	0x09	Null	No operation
0xFE	0x0A	None	Turn On Display
0xFE	0x0B	None	Turn Off Display
0xFE	0x0C	2 bytes	Set Cursor Position
0xFE	0x0D	None	Home Cursor
0xFE	0x0E	None	Turn On Underline Cursor
0xFE	0x0F	None	Turn Off Underline Cursor
0xFE	0x10	None	Move Cursor Left One Space
0xFE	0x11	None	Move Cursor Right One Space
0xFE	0x12	None	Turn On Blinking Cursor
0xFE	0x13	None	Turn Off Blinking Cursor
0xFE	0x14	None	Clear Screen
0xFE	0x15	Variable	Print String
0xFE	0x16	1 byte	Init Horizontal Bar Graph
0xFE	0x17	4 bytes	Draw Horizontal Bar Graph
0xFE	0x18	None	Init Vertical Bar Graph
0xFE	0x19	4 bytes	Draw Vertical Bar Graph
0xFE	0x1A	9 bytes	Load Custom Characters
0xFE	0x1B	None	Read Keypad

3.1 Configuration Commands.

All settings are stored in internal module EEPROM and loaded during power up.

Changing the I2C Slave Address

Syntax hexadecimal 0xFE 0x01 [adr]

Parameter	Length	Description
[adr]	1 byte	New I2C address

Description: This command sets the I2C address, the address must be 0x42 – 0x5A (‘A’–‘Z’). The address change requires 50 ms to take effect; therefore, the subsequent input must have an appropriate delay.
Default 0x4C

Changing BAUD Rate

Syntax hexadecimal 0xFE 0x02 [baud]

Parameter	Length	Description
[baud]	1 byte	New Serial communication rate (1 – 4)

Description: This command sets the serial communication rate, the single byte parameter select the desired BAUD rate as in the table below.
The rate change requires 50 ms to take effect; therefore, the subsequent input must have an appropriate delay.
Default 9600 BAUD

Parameter	BAUD
1	2400
2	4800
3	9600
4	19200

Set Backlight Brightness

Syntax hexadecimal 0xFE 0x03 [brightness]

Parameter	Length	Description
[brightness]	1 byte	Set backlight brightness (0 – 250)

Description: This command sets the backlight value, the single byte parameter select the desired brightness. The module modulates the backlight via a transistor. This allows to set different brightness settings.
The command requires 50 ms to take effect; therefore, the subsequent input must have an appropriate delay.
Default 150

Set Contrast

Syntax hexadecimal 0xFE 0x04 [contrast]

Parameter	Length	Description
[contrast]	1 byte	Set LCD contrast (0 – 100)

Description: This command sets the LCD contrast value, the single byte parameter select the desired contrast. The module set the contrast voltage.
The command requires 50 ms to take effect; therefore, the subsequent input must have an appropriate delay.
Default 15

Save Splash/Startup Screen

Syntax hexadecimal 0x05 0x61

Parameter	Length	Description
None	None	Save Splash screen in EEPROM

Description: This command save the Splash screen in internal EEPROM.
The splash screen is displayed for 500ms during power up. It verifies that the unit is powered and working correctly. To set the new splash screen, output the desired text to the top 2 lines, then send this command to save splash screen in EEPROM.

The command requires 50 ms to take effect; therefore, the subsequent input must have an appropriate delay.

Display Firmware Version Number
Syntax hexadecimal 0xFE 0x06

Parameter	Length	Description
None	None	Display the firmware version number

Description: This command displays the firmware version number.
Default: None.

Display Serial Baud Rate
Syntax hexadecimal 0xFE 0x07

Parameter	Length	Description
None	None	Display the baud rate

Description: This command displays the current baud rate.
Default: None.

Display I2C Address
Syntax hexadecimal 0xFE 0x08

Parameter	Length	Description
None	None	Display I2C address

Description: This command displays I2C address.
Default: None.

3.2 LCD Display Commands.

Turn On Display
Syntax hexadecimal 0xFE 0x0A

Parameter	Length	Description
None	None	Turn on LCD screen

Description: This command turn on the LCD display sceen.
Default LCD screen is on

Turn Off Display
Syntax hexadecimal 0xFE 0x0B

Parameter	Length	Description
None	None	Turn off LCD screen

Description: This command turn off the LCD display sceen.
Default LCD screen is on

Set Cursor Position
Syntax hexadecimal 0xFE 0x0C [col] [row]

Parameter	Length	Description
[row] [col]	2 bytes	Put cursor at location specified row and col

Description: This command moves the cursor to a specified location where the next character will be displayed. The row value - from 1 to 4, the column value - from 1 to 20.

Home Cursor

Syntax hexadecimal 0xFE 0x0D

Parameter	Length	Description
None	None	Position cursor at line 1 column 1

Description: This command moves the cursor move the cursor to line 1, column 1 of the LCD screen.

Turn On Underline Cursor

Syntax hexadecimal 0xFE 0x0E

Parameter	Length	Description
None	None	Turn on underline cursor

Description: This command turn on the underline cursor, the cursor position is where the next character will appear.
Default: The underline cursor is off.

Turn Off Underline Cursor

Syntax hexadecimal 0xFE 0x0F

Parameter	Length	Description
None	None	Turn off underline cursor

Description: This command turns off the underline cursor.
Default: The underline cursor is off.

Move Cursor Left One Space

Syntax hexadecimal 0xFE 0x10

Parameter	Length	Description
None	None	Move cursor left one space

Description: This command moves the cursor position left 1 space.
Default: None.

Move Cursor Right One Space

Syntax hexadecimal 0xFE 0x11

Parameter	Length	Description
None	None	Move cursor right one space

Description: This command moves the cursor position right 1 space.
Default: None.

Turn On Blinking Cursor

Syntax hexadecimal 0xFE 0x12

Parameter	Length	Description
None	None	Turn on the blinking cursor

Description: This command turns on the blinking cursor, both the cursor and the character on the cursor will blink.
Default: The blinking cursor is off.

Turn Off Blinking Cursor

Syntax hexadecimal 0xFE 0x13

Parameter	Length	Description
None	None	Turn off the blinking cursor

Description: This command turns off the blinking cursor.
Default: The blinking cursor is off.

Clear Screen

Syntax hexadecimal 0xFE 0x14

Parameter	Length	Description
None	None	Clear LCD and move cursor to line 1 column 1

Description: This command clears the display and place the cursor at line 1 column 1.
Default: None.

Print String

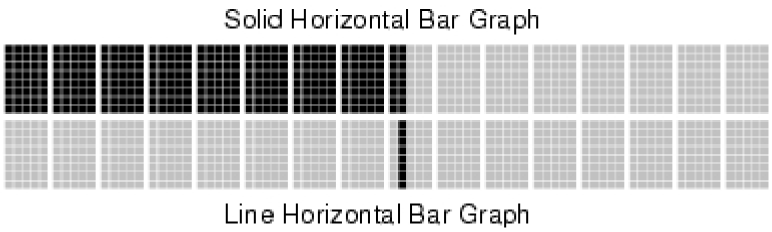
Syntax hexadecimal 0xFE 0x15 [count][string]

Parameter	Length	Description
[count][string]	variable	Print string

Description: This command prints the string with length [count] to the LCD at the present cursor position.
Default 15

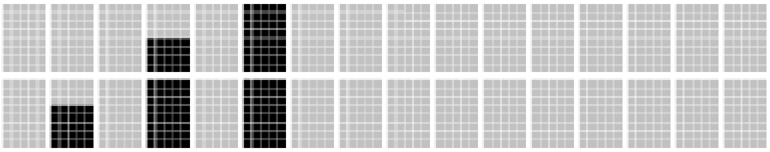
3.3 Bar Graphs and Custom Characters.

The module supports the horizontal and vertical Bar Graph functions.
Each display character consists of five horizontal pixels by eight vertical pixels.
Horizontal bar graphs display a set of vertical lines that are each composed of one horizontal pixel by eight vertical pixels, within a single character - a pixel column. Each character can display one to five vertical pixel columns, where five pixel columns display the entire character.
Starting on the left side of the display, the first pixel column is numbered 1 and the last pixel column is numbered N * 5, where N is the number of characters. A 16-character display has 80 possible pixel columns numbered 1 to 80.
Solid bar graphs display 1 to N pixel columns, within a set of specified continuous characters. Line bar graphs display only the specified pixel column. Below is an example of both types of horizontal bar graphs.



Vertical bar graphs display a set of horizontal lines that are each composed of one vertical pixel by five horizontal pixels, within a single character - a pixel row. Each character can display one to eight horizontal pixel rows, where eight pixel rows display the entire character.

Starting on the bottom of a character, the first pixel row is numbered 1 and the last pixel row is numbered 8. Combining two rows can generate a vertical bar graph of 16 pixel rows.



Only one type of bargraph (horizontal or vertical) may be used for current screen at a time since they each require their own set of characters and load those characters to LCD custom character RAM.

Init Horizontal Bar Graph

Syntax hexadecimal 0xFE 0x16 [type]

Parameter	Length	Description
[type]	1 byte	Init horizontal bar graph (type = 0 for solid graph, 1 – for line graph)

Description: Initializes the LCD to display the specified type of horizontal bar graph. This function should be called prior to calling Draw Horizontal Bar Graph. The type of bar graph must be specified. This function does not draw a bar graph, but loads the custom character RAM with the data required to display the specified type of bar graph. This routine must be called to change between horizontal bar-graph types.

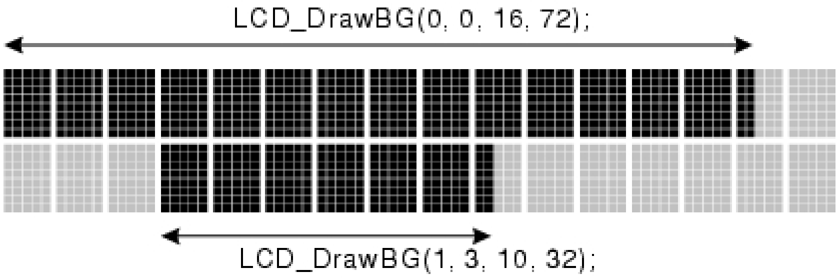
Draw Horizontal Bar Graph

Syntax hexadecimal 0xFE 0x17 [Row][Col][Len][PixelColEnd]

Parameter	Length	Description
[Row][Col][Len][PixelColEnd]	4 bytes	Draws the horizontal bar graph

Description: Draws the horizontal bar graph starting at character location (Row, Col) with a character length of "Len" to column position of "PixelColEnd".

Examples of Horizontal Bar Graphs



Note Solid bar graphs draw all the pixel columns from the first pixel column of the character defined by bRow and bCol to the pixel column specified by bPixelColEnd. Line bar graphs draw the specific pixel column specified in the define character.
For line bargraphs bLen=1 and bPixelColEnd is in the range of 1 to 5.

Init Vertical Bar Graph

Syntax hexadecimal 0xFE 0x18

Parameter	Length	Description
None	None	Init vertical bar graph

Description: Initializes the LCD to display vertical bar graphs. This should be called prior to calling Draw Vertical Bar Graph. This function initializes the custom character RAM with the data required to draw vertical bar graphs.

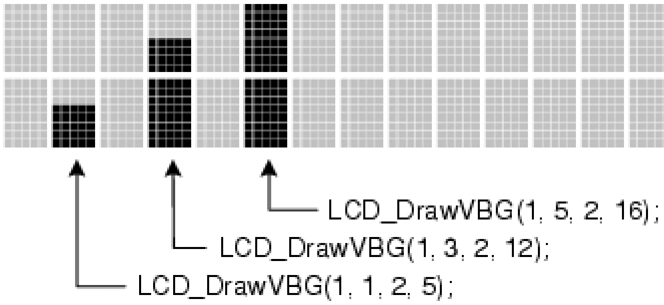
Draw Vertical Bar Graph

Syntax hexadecimal 0xFE 0x19 [Row][Height][Len][PixelRowEnd]

Parameter	Length	Description
[Row][Col][Height][PixelRowEnd]	4 bytes	Draws the horizontal bar graph

Description: Draws a vertical bar graph starting from the first pixel row at character location (Row, Col), with a character height of Height, up to the specified vertical pixel row PixelRowEnd.

Examples of Vertical Bar Graphs



Load Custom Characters

Syntax hexadecimal 0xFE 0x1A [addr][d0 … d7]

Parameter	Length	Description
[addr][d0 … d7]	9 bytes	Load custom characters, [addr] 1 byte – custom character address from 0 to 7, [d0 … d7] 8 bytes – custom character pattern bit map

Description: LCD module has space for 8 custom characters. Each custom character is 5 pixels wide by 8 pixels high.
The [addr] parameter indicates which custom character is defining, and must have a value from 0 to 7.
Following the [addr] parameter are 8 bytes that define the custom character. Bits 0 to 4 each byte byte will each define a pixel character.
Example: The bit map for character 'X'.

Bit	7	6	5	4	3	2	1	0	Hex
Byte 1	0	0	0	**1**	0	0	0	**1**	0x11
Byte 2	0	0	0	0	**1**	0	**1**	0	0x0A
Byte 3	0	0	0	0	0	**1**	0	0	0x04
Byte 4	0	0	0	0	**1**	0	**1**	0	0x0A
Byte 5	0	0	0	**1**	0	0	0	**1**	0x11
Byte 6	0	0	0	0	0	0	0	0	0x00
Byte 7	0	0	0	0	0	0	0	0	0x00
Byte 8	0	0	0	0	0	0	0	0	0x00

3.4 Keypad Commands.

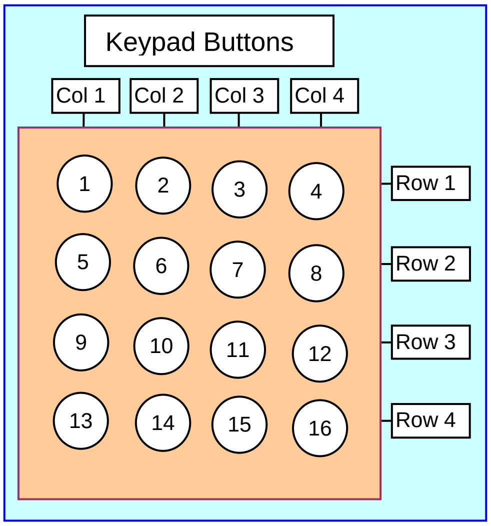
A keypad of up to 16 keys (4 rows by 4 columns) can be connected to the module. Each time a key is pressed, a key code from 1 (0x01) to 16 (0x10) is added to module keypad buffer and interface connector pin INT is pulled low. The keypad buffer can be read via Read Keypad Command. If INT signal is not used, the LCD module has to be polled to see if it has any keypad data available.

Read Keypad

Syntax hexadecimal 0xFE 0x1B

Parameter	Length	Description
None	None	Read keypad data

Description: This command sends a request for keypad data. The LCD module will return 1 byte keypad data. If there is no key data in the keypad buffer, 0 is returned. If there is key data, a key code from 0x01 to 0x10 is returned, depending on what key was pressed.


Keypad Buttons
Col 1
Col 2
Col 3
Col 4
1
2
3
4
Row 1
5
6
7
8
Row 2
9
10
11
12
Row 3
13
14
15
16
Row 4