MATRIX ORBITAL

X-Board S-Series Technical Manual

Revision: 1.0

Contents

1 Introduction

The X-Board was designed as a daughterboard to receive serial communication and display this data on an LCD, VFD or PLED. The X-Board will support an alphanumeric LCD, VFD or PLED display up to 80 characters. When you receive the X-Board it will contain the following parts:

- 1 - X-Board
- 2 - 1x8 strip headers
- 1 - Plastic Bumper

1.1 What to Expect From the X-Board

The X-Board comes equipped with the following features:

- Built in font with provision for up to 8 user defined characters for applicable displays
- Configured for 8bit, no parity and 1 stop bit communication (8N1)
- Utilizes a 4-bit bus interface
- Jumper selectable 9600bps or a lighting fast 19.2 Kbps serial communication speed
- Jumper selectable communication over RS-232 or TTL
- Fully buffered so that no delays in transmission should be necessary
- Ability to add a customized splash / startup screen
- 32 levels of software controlled brightness for an PLED display
- 4 levels of software controlled brightness for an VFD display
- 32 levels of software controlled contrast for an LCD display
- Three general purpose outputs for a variety of applications
- Horizontal or vertical bar graphs
- Medium digit capability
- Large digit capability on applicable displays (a display that consists of 4 rows)

1.2 Compatible Displays

The X-Board is compatible with any of Matrix Orbital's MOP-A Series displays excluding the following; MOP-Ax404X-XXX-XXXX-XXX. Included in this list is most 4 bit interface displays. For a complete list of the MOP Series, please see our website at www.matrixorbital.com

1.3 What Not to Expect From the X-Board

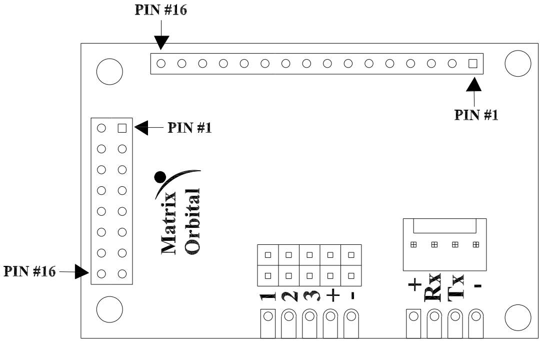
The X-Board does not include bitmap graphics capability, except that permitted by defining special characters. In addition it can't generate negative voltage. Negative voltage is required for extended temperature displays.

1.4 Installation with a typical LCD, VFD and PLED

Please refer to the figures below for installation procedures on how an X-Board will connect to a typical LCD, VFD, and PLED.

1. Once the type of display has been selected, solder the strip header pins accordingly. For example: if the display is an LCD 2x16, you would solder the 2 x 8 pin strip headers in the single 16 pin row of the display. Please see Figure1for further description.
2. Before soldering the display to the X-Board, place the plastic bumper in a convenient mounting location. It should be mounted in such a way to avoid contact of pins between the X-Board and the display. Please see Figure2 or Figure3 for further description.

NOTE Please be aware that these connections are typical and in some instances you may have more or less connection pins on the display, depending on the type of display you are connecting to. It is essential that pin 1 is recognized and connected correctly. The X-Board utilizes a 4-bit bus interface, therefore pins 7-10 must be connected accordingly from the data bus to the display. Pins 15 and 16 are used for the backlight in an LCD application. Pins 15 and 16 are not applicable for VFD and PLED applications.



PIN	SYMBOL	DESCRIPTION
1	Vss	Power Supply 0V
2	Vdd	Power Supply +5V
3	Vo	Power Supply for Display
4	RS	Register Select
5	R/W	Read/Write
6	E	Enable
7-14	DB0-DB7	Data Bus
15	A	Anode for Backlight
16	K	Cathode for Backlight

Figure 1: Pin Connections

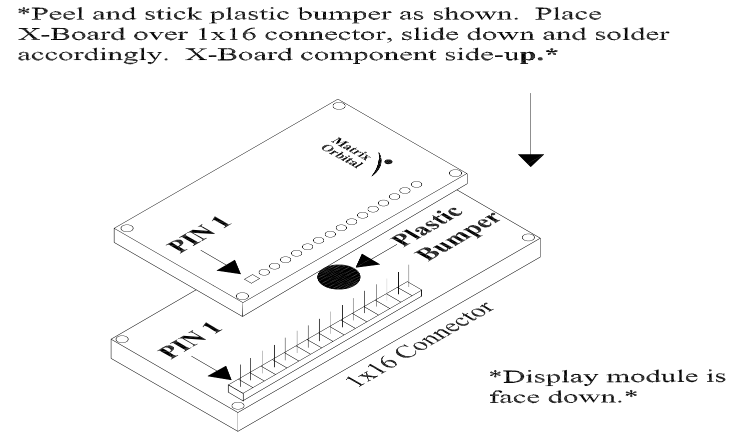

*Peel and stick plastic bumper as shown. Place
X-Board over 1x16 connector, slide down and solder
accordingly. X-Board component side-up.*
Matrix
Orbital
PIN 1
Plastic
Bumper
PIN 1
1x16 Connector
*Display module is
face down.*

Figure 2: 1x16 Connector

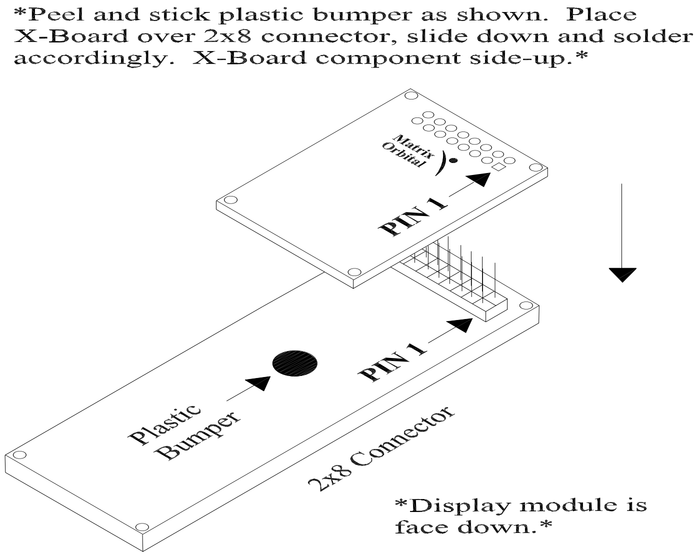

*Peel and stick plastic bumper as shown. Place
X-Board over 2x8 connector, slide down and solder
accordingly. X-Board component side-up.*
Matrix
Orbital
PIN 1
PIN 1
Plastic
Bumper
2x8 Connector
*Display module is
face down.*

Figure 3: 2x8 Connector

WARNING When an X-Board is purchased you will receive two plastic bumpers for mounting purposes. It is essential that you mount the X-Board properly to avoid contact with the display. Improper installation may cause a short circuit and void your warranty.

1.5 Configuring Your X-Board

1. The X-Board modules come with a pre-programmed module type. This means that the type and size of the display is pre-programmed to ???. This may very well be not the display type and size you will be using the X-Board with, therefore there is a need to configure (program) the X-Board to the type and size of your display.
2. Familiarize yourself with the 2x5 header connector, also known as the GPO header labeled in Figure 1. You will need 3 jumpers that will short a combination of the pins as described later. The connector has been labeled with “1,2,3,+,-”. You will only be shorting a combination of the pins labeled “1,2,3” to the other row of the header.

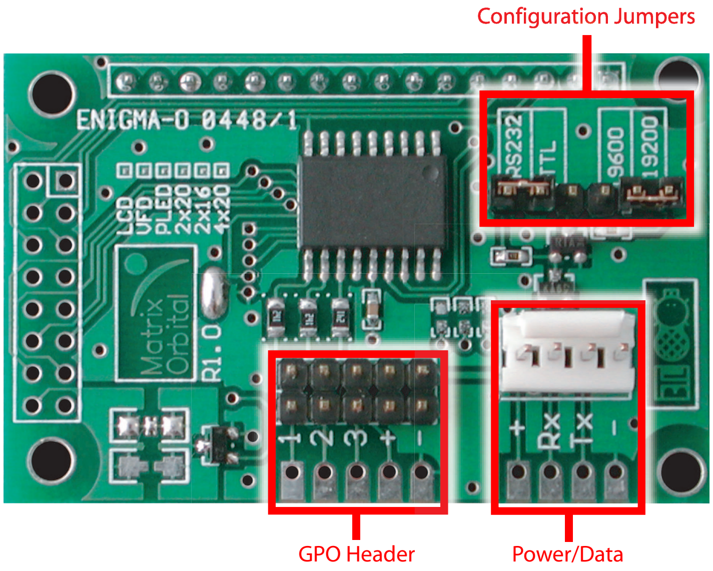
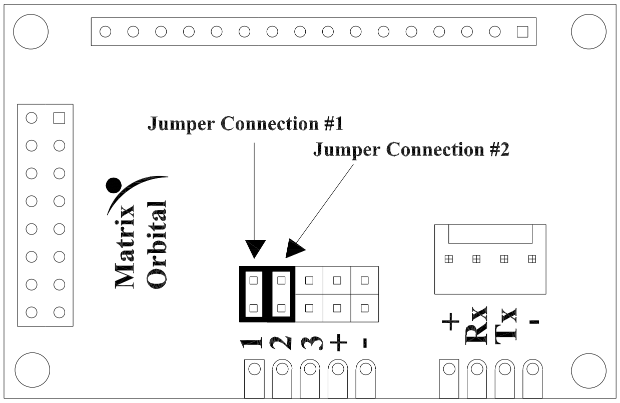


Figure 4: Connections for Testing

1. Connect the display that you intend to use with the X-Board. Be mindful of pin 1 for the display.
2. Before turning the power on, jumper connections 1, 2 and 3. This puts the unit to ‘re-configuring‘ mode. If at any point you want to change the display on your X-Board, this is one of the 2 ways to re-configure the X-Board: with power off, place the 3 jumpers, then turn the unit on.

3. After power up, “Restart” and “Set Type” is displayed on the screen. Your screen might not be displaying the best possible contrast/brightness but this is due to the fact that the X-Board does not know yet what type of display it is actually connected to.

4. Turn the power off. Place the jumpers necessary to configure the type of the display. Please refer to Table 5.

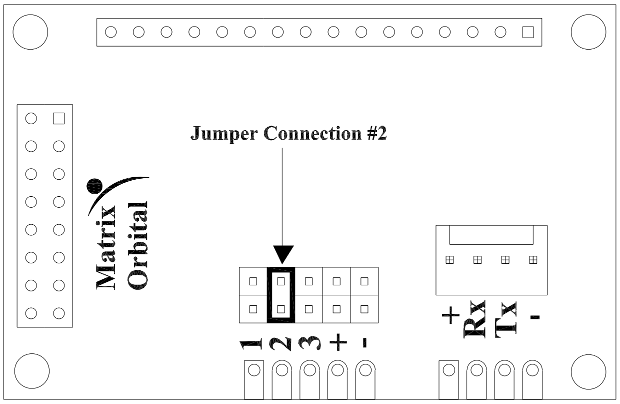


Display Type	Jumper PIN#
LCD (any color)	1
VFD (Samsung Type)	2
VFD (Noritake Type)	3
PLED	1,2

Figure 5: Configuring Display Type

5. Turn the power back on. If the correct pin(s) has been jumpered, the display type should now be displayed on the screen. “Set Size” should also be displayed on the screen now.

6. Turn the power off. Place the jumpers necessary to configure the size of the display. Please refer to Table **??**.



Display Size (row x col)	Jumper PIN#
1x16	0
2x8	1
2x16	2
2x20	1,2
2x24	3
2x40	1,3
4x16	2,3
4x20	1,2,3

Figure 6: Configuring Display Size

7. Turn the power back on. If the correct pin(s) has been jumpered, the display type and size should now be displayed on the screen. Turn the power off, take all the jumpers off and then turn the power back on. The unit is ready to talk to the type and size of the display you have configured it with.
8. At any point there has been a configuration set that is not within the tables given, the unit will display “Invalid” and the step is to be started from where it left off. For example, if you were configuring the type, then it will go back to configuring the type. If you get lost to where you are at in these steps, try jumpering ALL the pins 1,2,3 and start all over.

1.6 Setup for Testing With a PC

WARNING DO NOT apply power directly to the 4 pin connector of the X-Board. This will void your manufactures warranty and may cause serious damage.

NOTE To test the module with a PC, the appropriate cable will need to be ordered or created accordingly.

Before setting up the application, the user may want to try out the display. When connected to a PC, the following will be required:

- A 4-pin power connector of the type used to connect 3.5" floppy drive. Please see Figure8
- A PC with a spare RS-232 port (COM1 or COM2).
- A straight through serial cable (DB9F to DB9M). This part is available directly from Matrix Orbital or from one of our distributors.

- A custom cable is required for connection from the PC to the X-Board. This custom cable can be ordered directly from Matrix Orbital or from one of our distributors. The display connection is a four pin header. This header consists of power, Rx, Tx and ground as shown in Figure 7.

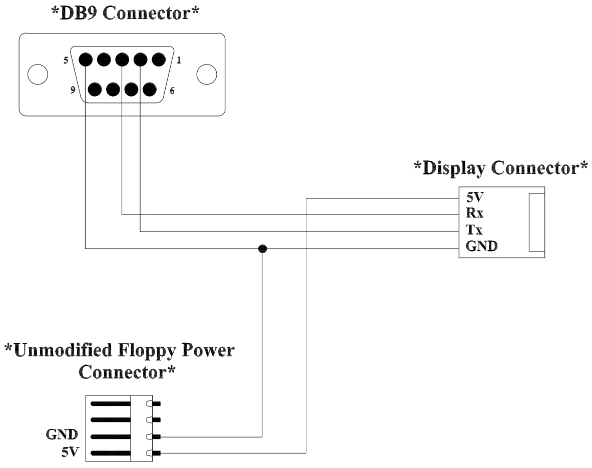


Figure 7: Connections for Testing

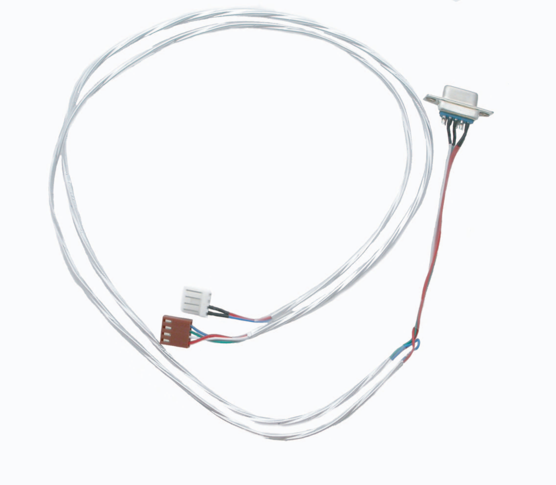


Figure 8: Communication and Power Cable

Please refer to Figure8 for communicating with the X-Board via a PC:

1. Connect the female end of your straight through serial cable to a spare com port on your PC.
2. Connect the male end of your straight through serial cable to the DB9 connector of the custom cable.
3. Using a spare unmodified floppy power cable (female), connect to the “Unmodified Floppy Power Connector (male)” of the custom cable. Since this is a locking connector, it will only fit one way.
4. The final step is to connect the “Display Connector” of the custom cable to the header on the X-Board labeled “Power/Data”

WARNING The Manufacturer’s Warranty becomes void if the unit is subjected to over-voltage or reversed polarity.

1.7 Trying Out the X-Board Using a PC

The unit should be connected to power as in Section 1.6 . The PC and display should be on. To experiment with typing text, run a PC terminal program, such as Hyperterm. Make sure it’s configured to use the correct com port. Set the baud rate to 19,200 as described in Section 2.3. If characters are typed on the keyboard, they should now appear on the display screen. Text will wrap around to the next line when the end of a line has been reached.

To exercise some of the other features of the display, a program (in any convenient language such as Basic or C) will need to be written in order to issue the required command strings. Most terminal programs are unable to issue the 0xFE character needed as a command prefix. Included with your purchase will be an e-CD and on this e-CD we have test programs that may be used at your convenience. These test programs have the ability to test such things as the backlight, contrast, GPO’s and etc.

1.8 Trying Out the X-Board Using a Basic Stamp

When connecting the module to the Basic Stamp development board, you will need to purchase the appropriate cable or modify a cable accordingly. Connection must be made via pins #2, #3 and #5 of the serial cable.

1. Connect pin #2 to P0
2. Connect pin #3 to P1
3. Connect pin #5 to Vss, which is ground

To connect the module with a development board, please see Figure 9. For code examples with the Basic Stamp, please see Section 1.9.

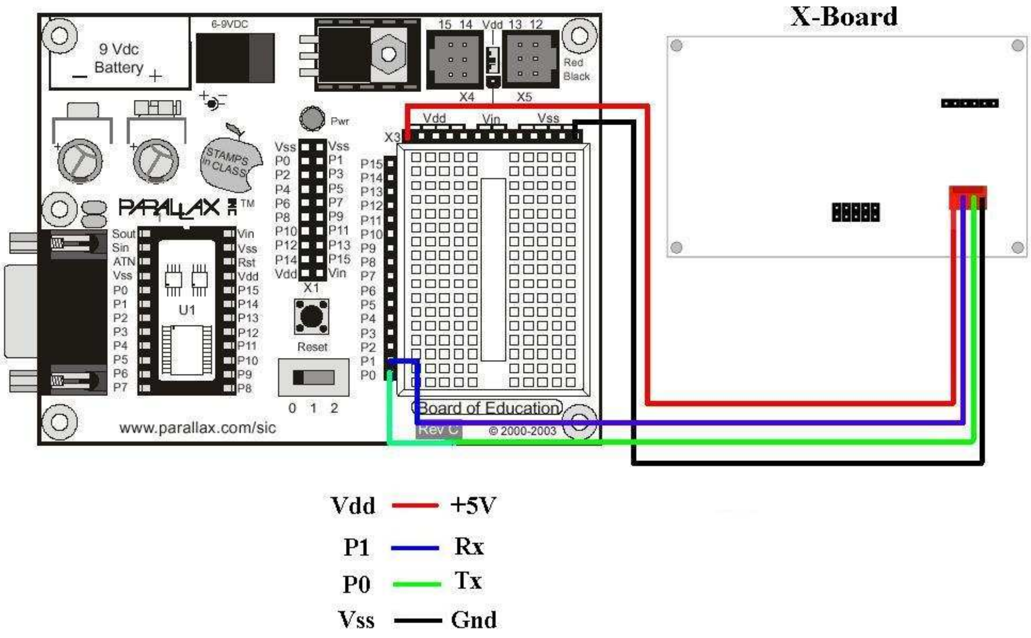


Figure 9: Interfacing to a Basic Stamp Development Board

1.9 Issuing Commands From a Basic Stamp

Before issuing commands from a Basic Stamp, the module and the Basic Stamp must be set up to communicate in TTL at 9600 baud. Please see Section 2.3 and Section 2.2.2 for setup configurations.

- Sending text to the display:

```
’{$STAMP BS2}
    SEROUT 1, 84, ["HELLO WORLD"]
```

This will display HELLO WORLD on your display.
SEROUT : is the command to send serial information out
1 : is Serial Port #1
84 : is the speed at which the information is sent at, in this case 9600bps.

- Sending a command to the display:

```
’{$STAMP BS2}
    SEROUT 1, 84, [254]    ’Command Prefix
    SEROUT 1, 84, [88]    ’Clear screen command
```

In this example, the command clear screen is sent to the display. Every command requires a "Command Prefix". With Matrix Orbital displays, that’s Hex: FE, Decimal: 254. Properly sent, any format can be used. Decimal format was used in this example.

- Setting the Backlight to go off in 2 minutes:

```
’{$STAMP BS2}
    SEROUT 1, 84, [254]    ’Command Prefix
    SEROUT 1, 84, [66]    ’Backlight ON command
    SEROUT 1, 84, [2]    ’Setting the number of minutes to be on
```

To permanently turn the backlight on, 0 would be sent as the third byte. When the display receives the command, it will know how many more bytes of information it should get. In this ‘backlight on’ case, the display knows to expect one more byte of information.

- Creating a medium digit:

```
’{$STAMP BS2}
    Digit VAR Byte    ’Create the variables Reps VAR NIB
    Digit = 0    ’Make sure the variables are 0 Reps = 0
    SEROUT 1, 84, [254]    ’Command Prefix
    SEROUT 1, 84, [88]    ’Clear screen command
    SEROUT 1, 84, [254]    ’Command Prefix
    SEROUT 1, 84, [109]    ’Initilize Medium Digits command
    FOR Reps = 1 TO 10    ’A loop to repeat it self 10 times
    SEROUT 1, 84, [254]    ’Command Prefix
    SEROUT 1, 84, [111]    ’Display medium digit command
    SEROUT 1, 84, [1]    ’Display medium digit in row 1
    SEROUT 1, 84, [1]      'Display medium digit in column 1
    SEROUT 1, 84, [Digit]      'Display the medium digit
    Digit = Digit + 1      'Incriment by 1
    PAUSE 1000      '1 second pause to see the text
    NEXT
    STOP
```

These 8 custom characters are user defined and can be used however the user sees fit.

NOTE If Medium Digits are used, Bar Graps or user defined Custom Characters cannot be used at the same time. It has to be one or the other.

1.10 Technical Support

For technical support regarding this module, please see the following links provided:

http://www.matrixorbital.ca

http://www.lcdforums.com/forums/

2 Connections

2.1 Connector Pinout

Refer to Figure 10 for this section.

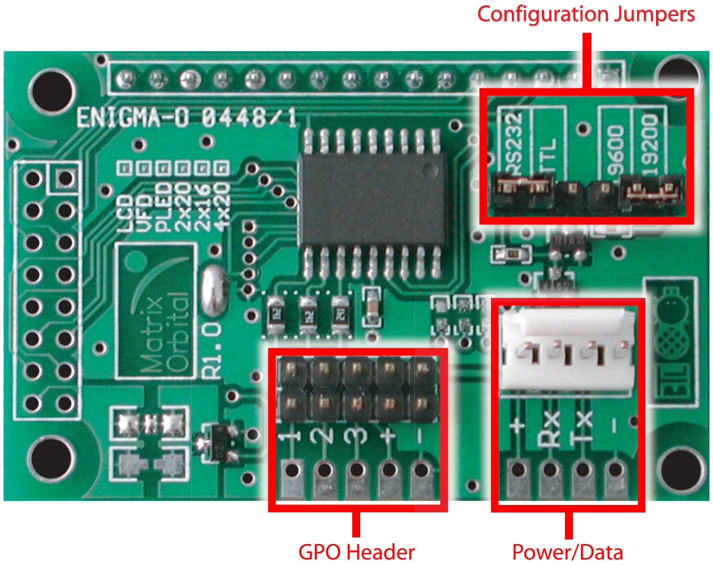


Figure 10: Electrical Connections

The display has four connectors as shown in Table 3.

Table 3: Connectors & Functions

Connector	Function
10 pin dual header	General purpose outputs (3)
4 pin	Power (5.0 Vdc), Data (Rx, Tx) and Ground
3 pin header	9600 or 19200 baud selection
3 pin header	RS232 or TTL communication selection

Table 5: Mating Connectors

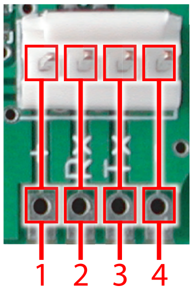
Connector	Part #	Mate Part #
10 Pin Dual Header	MOLEX 42375	Many, ex. MOLEX 70058
3 Pin Header	MOLEX 42375	Many, ex. MOLEX 70058
4 Pin White Header	AMP 171825-4	AMP 170205-1

2.1.1 Power Connections

Power is applied via pin1 and ground via pin 4 as shown in Figure 11. Power requirement is +5 VDC ±0.25V. As an alternate power connection, power may also be supplied via the “fingers” located below the 4 pin power connector. The “fingers” are provided as an additional feature allowing direct solder connection with through hole capability.

WARNINGS

- Do not apply any power with reversed polarization.
- Do not apply any voltage other than the specified voltage.
- Do not use any cables other than the cables supplied by Matrix Orbital, unless aware of the modifications required.
- When soldering to the fingers be aware of the heat being applied.
- When soldering to the fingers, check for bridging and shorts befor external power is applied.
- When soldering to the fingers, check for connectivity.



Pin 1	+5.0 Vdc
Pin 2	Rx
Pin 3	Tx
Pin 4	Ground

Figure 11: Power Connector

2.2 Communication Connection

The display communicates at 9600 or 19200 baud on RS-232 or TTL levels. Data from the module can only be received in TTL mode. The communication protocol is set up as follows: 8 data bits, no parity and 1 stop bit (8N1).

2.2.1 RS-232 Communications

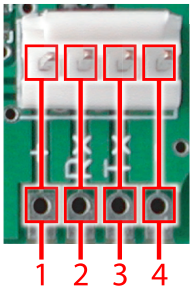
Using the custom cable available for this module, the PC’s Rx, Tx and ground connections will be connected to the 4 pin header located on the module. To make a custom cable for RS-232 communications, refer to Table 6 and Figure 11.

NOTES

- This device complies with the EIA232 standard in that it uses signal levels from +/- 3V to +/- 12V. It will not operate correctly at TTL (0 to +5V) levels without modification when used directly with a PC. A null modem cable will not work.
- With RS-232 communications, the display can only receive data, not transmit.
- The communication protocol is set up as 8N1.

Table 6: DB9 Pinout

Pin Number	Direction	Description	LCD	Host
2	Data from LCD	Data out (LCD)	Tx	Rx
3	Data to LCD	Data in (LCD)	Rx	Tx
5	-	Ground	gnd	gnd



Pin 1	+5.0 VDC
Pin 2	Rx
Pin 3	Tx
Pin 4	Ground

Figure 12: Power/Communication Connector

Please see Figure 13 for default RS-232 communication jumper settings.

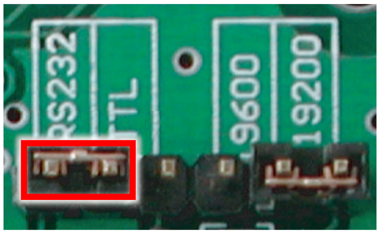


Figure 13: RS-232 Level Selected

2.2.2 TTL Communications

A PC is not to be used when TTL communication is established via a direct com port. Note that this device uses signal levels from 0V to + 5V on the TTL setting. One modification is required for TTL. The communication jumper has to be set to the TTL position. Please see Figure 14 for TTL configuration.

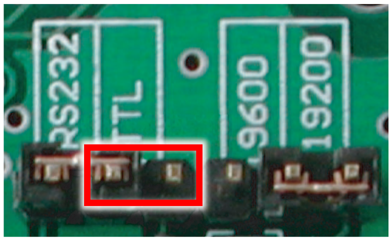
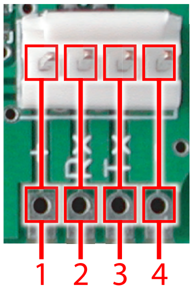


Figure 14: TTL Selected

For TTL communication via the 4 pin connector, please see Figure 11.



Pin 1	+5.0 VDC
Pin 2	Rx
Pin 3	Tx
Pin 4	Ground

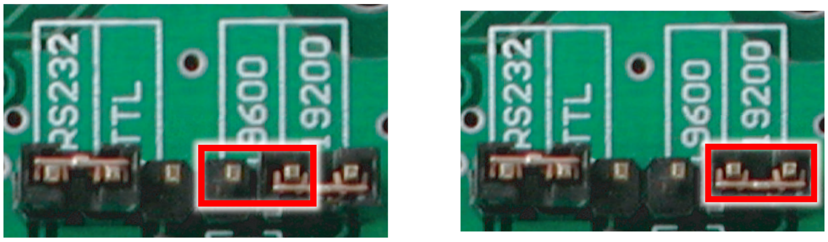
Figure 15: Power/Communication Connector

NOTE Data can be received from the display at TTL levels, such as the module type and firmware version.

2.3 Set Baud Rates

With this module you can only select between two baud rates. The baud rates available are 9600 or 19200 baud in 8N1 format. Please see Figure 16 on how to configure for each baud rate.

NOTE The baud selection is only effective at power up. Jumpers may be changed from 9600 to 19200 or vice versa, but the only effective change is the setup at power up.



(a) 9600 Baud

(b) 19200 Baud

Figure 16: Baud Rate Jumper

2.4 General Purpose Outputs

The display has three general purpose outputs. These are provided to control relays or other electronic devices. This allows control of external devices using the PC or host controller and software commands. Please see Figure 18 for GPO pinout.

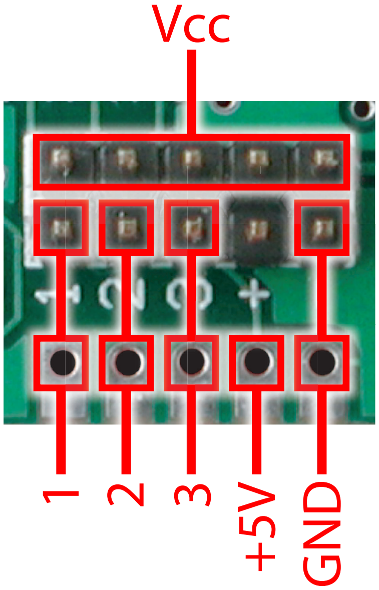


Figure 17: General Purpose Pinout

Each output is wired as shown in Figure 18 . The + terminal is connected directly to the module positive supply, the - terminal is connected through a 240 ohm current limiting resistor and the electronic switch to ground.

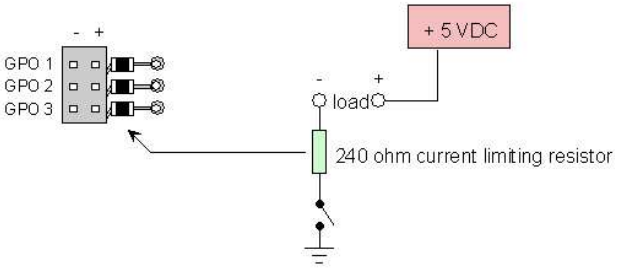


Figure 18: General Purpose Outputs

Maximum allowable current is 20 mA, which is enforced by the current limiting resistor. If the device being switched has a resistance of 240 ohms or more the corresponding resistor may be shorted. To short the corresponding resistor, solder a small jumper wire (wirewrap wire is good) accross the resistor(s), or take the resistor(s) out and short the resistor pads.

NOTE The GPOs do not have any over current or over / under voltage protection so care must be taken when using them. For instance if the external device is a relay, it must be fully clamped (using a diode and capacitor) to absorb any generated back electro-motive force (EMF). Please refer to Figure 19 on clamping a relay.35

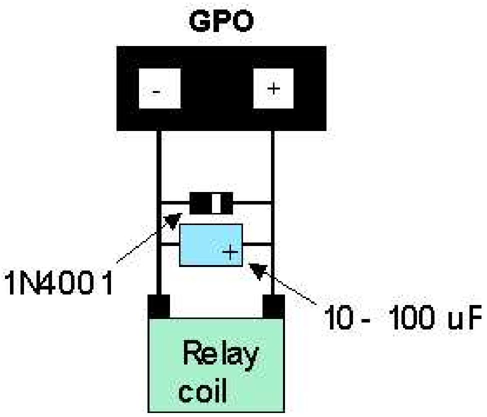


Figure 19: Clamping a Relay

3 Issuing Commands

Commands are issued to the display by the controller. In a test setup, commands can be issued to the display by means of a Basic program, using the chr$( ) function. In the tables below, we've shown commands in hex, ASCII and decimal form. All commands begin with the prefix character 0xFE (254 decimal). These commands are issued on the serial communications link (RS-232 or TTL) at the currently defined baud rate.

For example (using BASIC in a test setup), the user could issue the command clear screen on the display by including the line:

```
PRINT#1,chr$(254);chr$(88).
```

Or, with C the user could (using Zcomm serial library)

```
ZComm1->WriteCommByte(0xfe); // command prefix hex value
ZComm1->WriteCommByte(0x58); // command hex value for “clear screen”
ZComm1->WriteCommByte(0xfe); // command prefix hex value
ZComm1->WriteCommByte('X');  // command ASCII value for "clear screen"
ZComm1->WriteCommByte(254);  // command prefix decimal value
ZComm1->WriteCommByte(88);   // command decimal value for "clear screen"
```

3.1 Example Command

Syntax	Hexadecimal	0xFE 0x6F [row] [column] [digit]
	Decimal	254 111 [row] [column] [digit]
	ASCII	254 “o” [row] [column] [digit]

Parameters /home/samba/engineering/AutocadDrawings/XBoardrev1.0.eps

Description Draws a medium [digit] in the specified [row] and [column]. Medium digits occupy two rows and a single column. When [row] is specified as ‘1’, the medium digit will be displayed across the first and second rows of the display. When [row] is specified as ‘2’, only the top part of the medium digit will be visible on the second line of the display. When [row] is specified as ‘0’, only the bottom part of the medium digit will be shown on the first line of the display.
Before placing a medium digit, it is advised that you should initialize the custom characters with the "Initialize Medium Digits" command.

Remembered No

Examples Placing a medium digit on the screen:

```
txUArt.sendByte(0xFE); // command prefix
txUArt.sendByte(‘m’); // initialize medium digits
txUArt.sendByte(0xFE); // command prefix
txUArt.sendByte(‘o’); // place medium digit command
txUArt.sendByte(1); // place the digit on the first and second row
txUArt.sendByte(2); // place the digit in the second column
txUArt.sendByte(3); // place a ’3’ on the screen
```

Syntax This is a quick summary of the format of a command including the placement of parameters in hexadecimal, decimal and if appropriate ASCII representations.

Parameters Each parameter will be outlined along with the length, description and the valid values.

Description The description will outline the usage of the command in detail.

Remembered The remember command can be a convenient method to set up the start up state of the display. While the remember function is active, changes made by certain commands are written to non-volatile memory. Remembered could be one of three values:

- Yes - any changes made with this command will be remembered as long as the remember function is active.
- No - the state of the remember function does not affect any changes made with this command and they are not committed to non-volatile memory.
- Always - the state of the remember function does not affect any changes made with this command and they are always committed to non-volatile memory.

⚠ **WARNING** It is not recommended to leave the remember function active. With it active, it is very easy to reach the maximum write limit of the non-volatile memory which will cause the unit to malfunction.

Examples If the examples section is present, it will demonstrate the basic usage of the command.

4 Text Commands

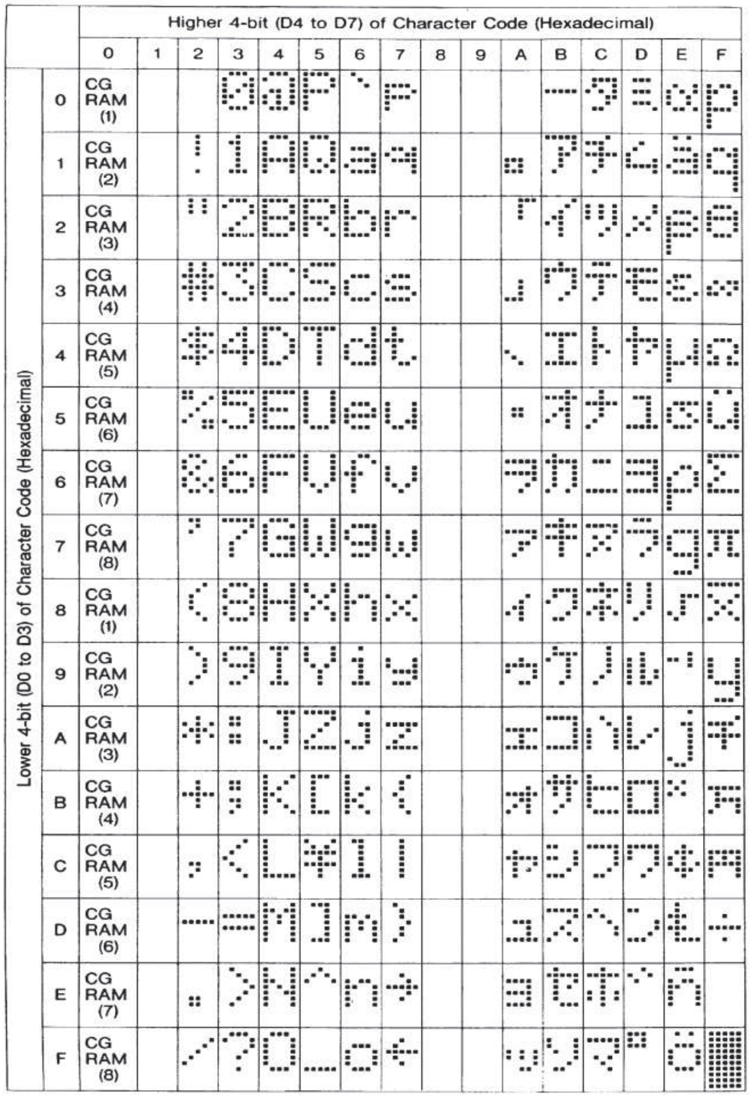
4.1 Introduction

When the display receives a character, it displays that character at the position currently defined. The next character sent to the module then advances to the following position on the display. Characters are drawn using the built in font of the display it is attached to, and only characters defined in the font are actually displayed. Characters that are not defined by the built in font print as a space (i.e., the cursor is advanced for the next character). The position where text is to be inserted is a character location stored in the display's volatile memory and maintained internally by the display's firmware. The commands in this section perform various functions that involve how text is presented on the display.

To display straight text, send equivalent ASCII, hex or decimal value of the appropriate character. Text is displayed on the X-Board using the built in 5x8 dot matrix font of the display that it is connected to. In addition, there are up to 8 user defined characters.

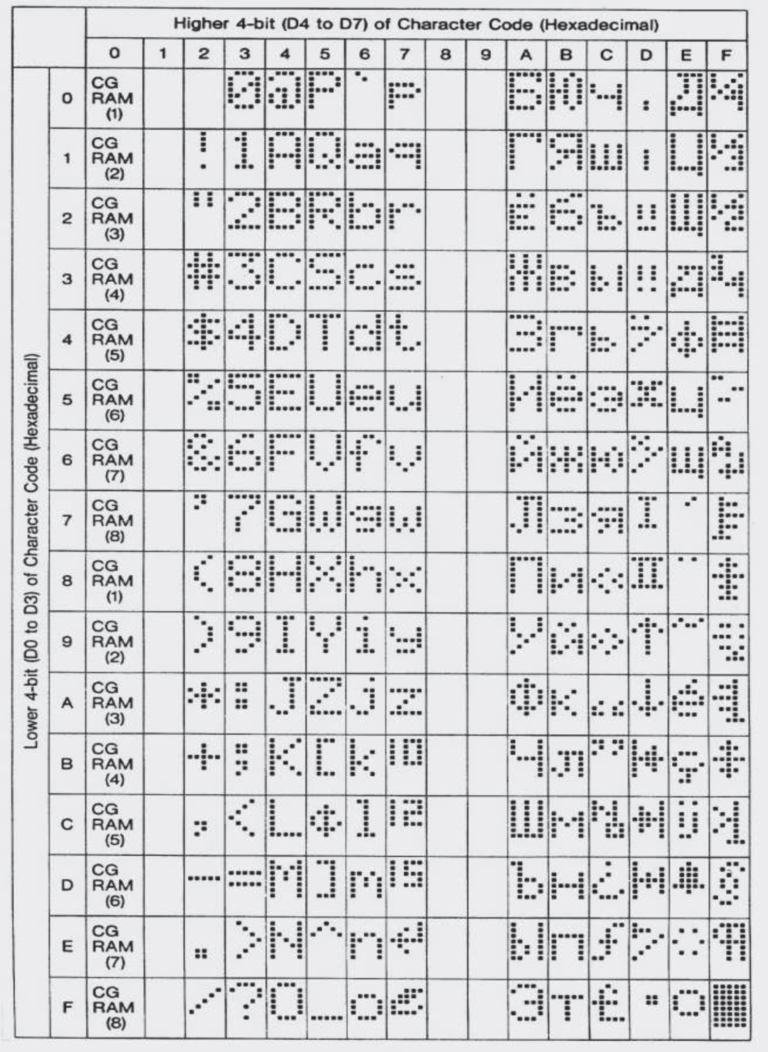
4.2 The Built In Character Font

When you purchase a display, it will include a built in character set that includes a full range of ASCII characters, plus a variety of extended characters. Depending on the type of display you purchase, the character set will be identified accordingly. Please see the following for character set examples.



		Higher 4-bit (D4 to D7) of Character Code (Hexadecimal)															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Lower 4-bit (D0 to D3) of Character Code (Hexadecimal)	0	CG RAM (1)			0	@	P	`	p				ー	タ	ミ	α	p
	1	CG RAM (2)		!	1	A	Q	a	q			。	ア	チ	ム	ä	q
	2	CG RAM (3)		"	2	B	R	b	r			「	イ	ツ	メ	β	θ
	3	CG RAM (4)		#	3	C	S	c	s			」	ウ	テ	モ	ε	∞
	4	CG RAM (5)		$	4	D	T	d	t			、	エ	ト	ヤ	μ	Ω
	5	CG RAM (6)		%	5	E	U	e	u			・	オ	ナ	ユ	σ	ü
	6	CG RAM (7)		&	6	F	V	f	v			ヲ	カ	ニ	ヨ	ρ	Σ
	7	CG RAM (8)		'	7	G	W	g	w			ァ	キ	ヌ	ラ	g	π
	8	CG RAM (1)		(	8	H	X	h	x			ィ	ク	ネ	リ	√	x̄
	9	CG RAM (2)		)	9	I	Y	i	y			ゥ	ケ	ノ	ル	⁻¹	y
	A	CG RAM (3)		*	:	J	Z	j	z			ェ	コ	ハ	レ	j	千
	B	CG RAM (4)		+	;	K	[	k	{			ォ	サ	ヒ	ロ	ˣ	万
	C	CG RAM (5)		,	<	L	¥	l	|			ャ	シ	フ	ワ	¢	円
	D	CG RAM (6)		-	=	M	]	m	}			ュ	ス	ヘ	ン	£	÷
	E	CG RAM (7)		.	>	N	^	n	→			ョ	セ	ホ	゛	ñ	
	F	CG RAM (8)		/	?	O	_	o	←			ッ	ソ	マ	゜	ö	█

Figure 20: Example of a English/Japanese Character Set



Lower 4-bit (D0 to D3) of Character Code (Hexadecimal)	Higher 4-bit (D4 to D7) of Character Code (Hexadecimal)															
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	CG RAM (1)			0	@	P	`	p			Б	Ю	Ч	·	Д	¼
1	CG RAM (2)		!	1	A	Q	a	q			Г	Я	ш	:	Ц	⅓
2	CG RAM (3)		"	2	B	R	b	r			Ё	б	ъ	[illegible]	Щ	½
3	CG RAM (4)		#	3	C	S	c	s			Ж	в	ы	‼	д	¾
4	CG RAM (5)		$	4	D	T	d	t			З	г	ь	[illegible]	ф	▤
5	CG RAM (6)		%	5	E	U	e	u			И	ё	э	ж	ц	[illegible]
6	CG RAM (7)		&	6	F	V	f	v			Й	ж	ю	[illegible]	щ	[illegible]
7	CG RAM (8)		'	7	G	W	g	w			Л	з	я	I	´	[illegible]
8	CG RAM (1)		(	8	H	X	h	x			П	и	«	Ⅱ	¨	‡
9	CG RAM (2)		)	9	I	Y	i	y			У	й	»	↑	˜	[illegible]
A	CG RAM (3)		*	:	J	Z	j	z			Ф	к	„	↓	é	[illegible]
B	CG RAM (4)		+	;	K	[	k	10			Ч	л	"	[illegible]	ç	‡
C	CG RAM (5)		,	<	L	ф	l	12			Ш	м	№	[illegible]	ü	[illegible]
D	CG RAM (6)		-	=	M	]	m	15			Ъ	н	¿	[illegible]	[illegible]	§
E	CG RAM (7)		.	>	N	^	n	↵			Ы	п	ƒ	▷	⁘	¶
F	CG RAM (8)		/	?	O	_	o	[illegible]			Э	т	£	▪	○	█

Figure 21: Example of a Cyrillic Character Set

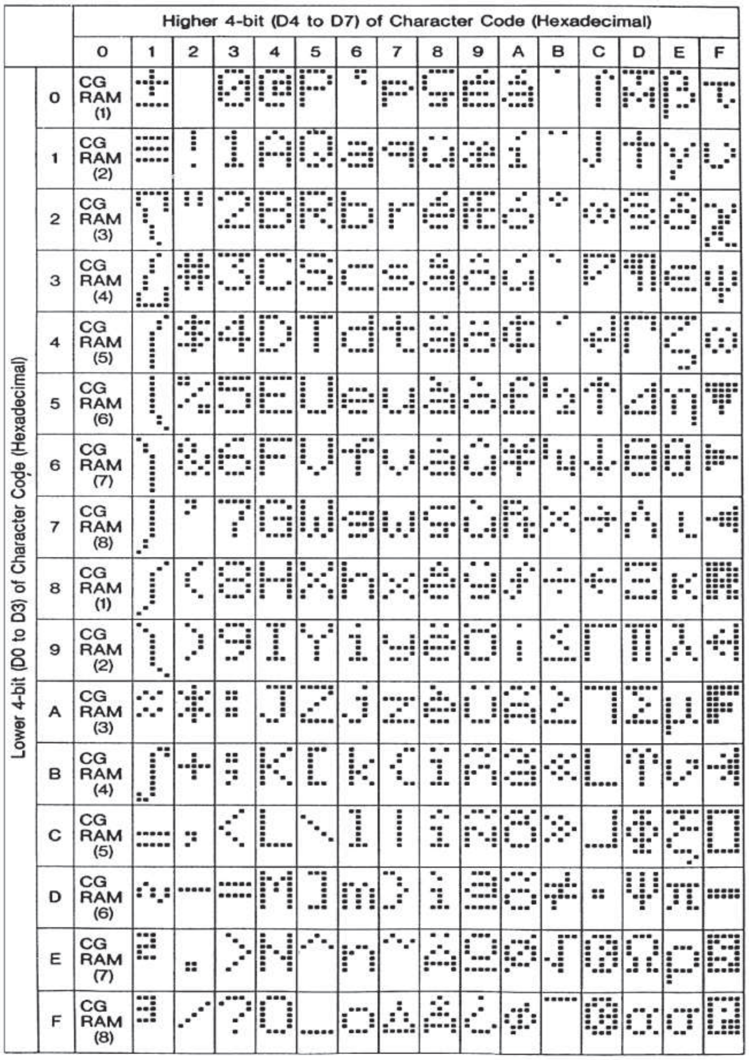

Higher 4-bit (D4 to D7) of Character Code (Hexadecimal)
0
1
2
3
4
5
6
7
8
9
A
B
C
D
E
F
Lower 4-bit (D0 to D3) of Character Code (Hexadecimal)
CG RAM (1)
CG RAM (2)
CG RAM (3)
CG RAM (4)
CG RAM (5)
CG RAM (6)
CG RAM (7)
CG RAM (8)
CG RAM (1)
CG RAM (2)
CG RAM (3)
CG RAM (4)
CG RAM (5)
CG RAM (6)
CG RAM (7)
CG RAM (8)

Figure 22: Example of a European Character Set

4.3 Command List

4.4 Introduction

The X-Board has a predefined command list that is available to the user. Depending on the type of display that is used, certain commands are only compatible with certain displays. Please see the legend below for compatibility:

Compatibility

LCD	VFD	PLED	ALL
L	V	P	A

4.4.1 Auto Line Wrap On

Syntax

Hexadecimal	0xFE 0x43
Decimal	254 67
ASCII	254 “C”

Description Enables automatic line wrapping. Note that this is not “word wrapping” and wraps may occur in the middle of a word.

Remembered Yes

Compatibility A

4.4.2 Auto Line Wrap Off

Syntax

Hexadecimal	0xFE 0x44
Decimal	254 68
ASCII	254 “D”

Description Disables automatic line wrapping. Characters beyond the end of a line will be lost.

Remembered Yes

Compatibility A

4.4.3 Set Cursor Position

Syntax

Hexadecimal	0xFE 0x47 [column] [row]
Decimal	254 71 [column] [row]
ASCII	254 “G” [column] [row]

Parameters

Parameter	Size	Description
column	1	The column number (1 - 16)
row	1	The row number (1 - 2)

Description This command sets the cursor position (text insertion point) to the [column] and [row] specified. Columns have values from 1 to 16 and rows have values from 1 to 2.

Remembered No

Compatibility A

4.4.4 Send Cursor Home

Syntax

Hexadecimal	0xFE 0x48
Decimal	254 72
ASCII	254 “H”

Description This command moves the cursor position (text insertion point) to the top left of the display area.

Remembered No

Compatibility A

4.4.5 Turn On Underline Cursor

Syntax

Hexadecimal	0xFE 0x4A
Decimal	254 74
ASCII	254 “J”

Description Turns on the underline cursor. The cursor shows the current text insertion point. Both blinking block and underline cursors may be turned on or off independently. The cursor is off by default.

Remembered Yes

Compatibility A *Except for a 4x20 VFD as this display does not have this capability*

4.4.6 Turn Off Underline Cursor

Syntax	Hexadecimal	0xFE 0x4B
	Decimal	254 75
	ASCII	254 “K”

Description Turns off the underline cursor. The cursor is off by default.

Remembered Yes

Compatibility A *Except for a 4x20 VFD as this display does not have this capability*

4.4.7 Cursor Left

Syntax	Hexadecimal	0xFE 0x4C
	Decimal	254 76
	ASCII	254 “L”

Description Moves the cursor one position to the left but does not erase any character that may be in that position. Note that this command moves the text insertion point even if the cursor is turned off.

NOTE A “destructive backspace”, which erases the character to the left of the original position, may be done by issuing the following sequence: cursor left, space, cursor left.

Remembered No

Compatibility A

4.4.8 Cursor Right

Syntax	Hexadecimal	0xFE 0x4D
	Decimal	254 77
	ASCII	254 “M”

Description Moves the cursor one position to the right but does not erase any character that may be in that position. Note that this command moves the text insertion point even if the cursor is turned off.

Remembered No

Compatibility A

4.4.9 Auto Scroll On

Syntax	Hexadecimal	0xFE 0x51
	Decimal	254 81
	ASCII	254 “Q”

Description When auto scrolling is on, it causes the display to shift the entire display’s contents up to make room for a new line of text when the text reaches the scroll position (the bottom right character position).

Remembered Yes

Compatibility A

4.4.10 Auto Scroll Off

Syntax	Hexadecimal	0xFE 0x52
	Decimal	254 82
	ASCII	254 “R”

Description When auto scrolling is disabled, text will wrap to the top left corner of the display area. Existing text in the display area is not erased before new text is placed. A series of spaces followed by a “Cursor Home” command may be used to erase the top line of text.

Remembered Yes

Compatibility A

4.4.11 Turn On Blinking Block Cursor

Syntax

Hexadecimal	0xFE 0x53
Decimal	254 83
ASCII	254 “S”

Description Turns on the blinking block cursor. The cursor shows the current text insertion point. Both blinking block and underline cursors may be turned on or off independently. The cursor is off by default. For PLED displays,the blinking block cursor is only available when a character is present. If a character is not present, the blinking block cursor may be turned "on", however not visible.

Remembered Yes

Compatibility A

4.4.12 Turn Off Blinking Block Cursor

Syntax

Hexadecimal	0xFE 0x54
Decimal	254 84
ASCII	254 “T”

Description Turns off the blinking block cursor. Does not affect the underline cursor. The cursor is off by default. For PLED displays,the blinking block cursor is only available when a character is present. If a character is not present, the blinking block cursor may be turned "on", however not visible.

Remembered Yes

Compatibility A

4.4.13 Clear Display

Syntax	Hexadecimal	0xFE 0x58
	Decimal	254 88
	ASCII	254 “X”

Description: This command clears the display and resets the text insertion point to the top left of the screen.

Remembered: No

Compatibility: A

5 Bar Graphs and Special Characters

5.1 Introduction

The display includes the ability to draw bar graphs (either horizontal or vertical) and allows users to define up to eight special characters.

Eight characters (ASCII values 0x00 to 0x07) are set aside for use with bar graphs, user defined characters, and large and medium sized numbers. Since the same 8 characters are used for each function, the functions may not be used simultaneously. The characters may be defined or redefined at any time by issuing commands shown in this section. Once defined, they may be used either by means of the bar graph commands, or by simply issuing one of the ASCII values 0x00 to 0x07, which are not prefixed by the command byte, 254.

5.2 Command List

5.2.1 Draw Vertical Bar Graph

Syntax	Hexadecimal	0xFE 0x3D [column] [height]
	Decimal	254 61 [column] [height]
	ASCII	254 “=” [column] [height]

Parameters

Parameter	Size	Description
column	1	The column number (1 - 16)
height	1	The height value (0 - 16)

Description

Draws a vertical bar graph in [column] having a height of [height] pixels. The height may range from 0 to 16 pixels. The necessary characters must be first initialized by either of the commands Initialize Wide Vertical Bar Graph (254 118), Initialize Narrow Vertical Bar Graph (254 115) or Define Custom Characters (254 78). The said commands will determine the width of the vertical graph drawn. Graphs may be erased by drawing a bar graph of height = 0 in the same column.

Remembered

No

Compatibility

A

5.2.2 Load Startup Screen

Syntax

Hexadecimal	0xFE 0x40 [characters]
Decimal	254 64 [characters]
ASCII	254 “@” [characters]

Parameters

Parameter	Size	Description
characters	Rows x Columns	Any character to be defined by the user.

Description

This command sets and memorizes the startup screen that will appear each time the display is powered on.
By default the screen shows:

Matrix Orbital
X-Board

The characters are defined by the type of display you have. For example: If you have a display that is 4 rows by 20 columns, then the maximum amount of characters you can display is 80. 20 characters per row.
If sending more than 10 characters to be stored, add in about 10ms per character delay. Predefined custom characters can also be used in the startup screen, by using 0x00 through 0x07 characters. A 2 row by 16 column display for example:

Character 1 16
Cahracter 17 32

Remembered

Always

Compatibility

A

5.2.3 Define Custom Character

Syntax	Hexadecimal	0xFE 0x4E [address] [definiton]
	Decimal	254 78 [address] [definiton]
	ASCII	254 “N” [address] [definiton]

Parameters

Parameter	Size	Description
address	1	The address (0x00 to 0x07)
definiton	8	The character definition

Description

The display allows up to 8 user defined (custom) characters. These characters occupy the first 8 (0x00 to 0x07) places in the character set.

Built-in and custom characters occupy a 5x8 pixel matrix for an LCD and PLED display. For a VFD display the built in and custom characters occupy a 5 x7 pixel matrix as the last row is reserved for an underline, except in the case for a 4x20 VFD display. It does not reserve the the last row for a an underline. A character is defined by issuing the command 254 78 [c] followed by 8 bytes to define each pixel row for the cahracter. [c] is the character number (0x00 to 0x07). The 8 bytes are mapped as shown for an LCD and PLED display:

MSB							**LSB**	
*	*	*	1	2	3	4	5	Data Byte 1
*	*	*	6	7	8	9	10	Data Byte 2
*	*	*	11	12	13	14	15	Data Byte 3
*	*	*	16	17	18	19	20	Data Byte 4
*	*	*	21	22	23	24	25	Data Byte 5
*	*	*	26	27	28	29	30	Data Byte 6
*	*	*	31	32	33	34	35	Data Byte 7
*	*	*	36	37	38	39	40	Data Byte 8

A ‘1’ bit indicates an on (black) pixel, a ‘0’ bit indicates an off (clear) pixel.

Once defined, a character is displayed simply by issuing a value (0x00 to 0x07) corresponding to the character number. The character will be laid out as follows:

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25
26	27	28	29	30
31	32	33	34	35
36	37	38	39	40
Cursor Line				

NOTE Custom characters will be erased if any of the “Initialize” commands in this section are issued after defining the custom characters, e.g. Initialize Horizontal Bar Graph, Initialize Medium Digits.

Remembered No

Examples

Defining a Degree Symbol:

```
txUart.sendByte(0xFE); // command prefix
txUart.sendByte('N'); // custom character command
txUart.sendByte(0x00); // custom character value 0-7 and in this case it
is custom character 0
txUart.sendByte(12); // The following 8 bytes create the degree synbol
txUart.sendByte(18);
txUart.sendByte(18);
txUart.sendByte(12);
txUart.sendByte(0);
txUart.sendByte(0);
txUart.sendByte(0);
txUart.sendByte(0);
```

To display the character defined:

```
txUart.sendByte(0x00); // display custom character 0
```

Compatibility A

5.2.4 Initialize Horizontal Bar Graph

Syntax

Hexadecimal	0xFE 0x68
Decimal	254 104
ASCII	254 “h”

Description

This command defines the 8 special / user characters to be blocks suitable for use in drawing horizontal bar graphs. Any previously existing definitions will be lost. Once this command has been issued, any number of horizontal bar graphs may be drawn unless the characters are re-defined by another command.

Remembered No

Compatibility A

5.2.5 Initialize Medium Digits

Syntax

Hexadecimal	0xFE 0x6D
Decimal	254 109
ASCII	254 “m”

Description — This command defines the 8 special / user characters to be lines suitable for use in drawing medium digits. Any previously existing definitions will be lost. Once this command has been issued, any number of medium digits may be drawn unless the characters are re-defined by another command.

Remembered — No

Compatibility — A *Best used with displays that have 2 rows*

5.2.6 Draw Medium Digits

Syntax

Hexadecimal	0xFE 0x6F [row] [column] [digit]
Decimal	254 111 [row] [column] [digit]
ASCII	254 “o” [row] [column] [digit]

Parameters

Parameter	Size	Description
row	1	The row number (0 - 2)
column	1	The column number (1 - 16)
digit	1	Digit (0 - 9)

Description — Draws a medium [digit] in the specified row and column. Medium digits occupy two rows and a single column. When [row] is specified as ‘1’, the medium digit will be displayed across the first and second rows of the display. When [row] is specified as ‘2’, only the top part of the medium digit will be visible on the second line of the display. When [row] is specified as ‘0’, only the bottom part of the medium digit will be shown on the first line of the display.
Before placing a medium digit, it is advised that you should initialize the custom characters with the "Initialize Medium Digits" command.

Remembered — No

Examples

Placing a medium digit on the screen:

```
txUart.sendByte(0xFE); // command prefix
txUart.sendByte('m'); // initialize medium digits
txUart.sendByte(0xFE); // command prefix
txUart.sendByte('o'); // place medium digit
txUart.sendByte(1); // place the digit on the first and second rows
txUart.SendByte(2); // place digit in the second column
txUart.sendByte(3); // place a '3' on the screen
```

Compatibility

A *Best used with displays that have 2 rows*

5.2.7 Initialize Large Digits

Syntax

Hexadecimal	0xFE 0x6E
Decimal	254 110
ASCII	254 “n”

Description

This command defines the 8 special / user characters to be blocks suitable for use in drawing large digits. Any previously existing definitions will be lost. Once this command has been issued, any number of large characters may be placed until the characters are redefined by another command.

Remembered

No

Compatibility

A *Best used with displays that have 4 rows*

5.2.8 Draw Large Digits

Syntax

Hexadecimal	0xFE 0x23 [col] [digit]
Decimal	254 35 [col] [digit]

Parameters

Parameter	Size	Description
column	1	The column number (1 - 18)
digit	1	The digit number (0 - 9)

Description This command allows the large digits to be drawn on the display screen. Numbers of almost full display height may be placed along side regular text on four row displays. The column number has a maximum value which is less than the display width because the digits are all three columns wide. Before using this command the initialize large digits command must be issued to define the blocks necessary to make up the digits. If regular text and large digits are mixed on one screen, the user should always set the display cursor position before placing regular text because the creation of a large digit will leave the cursor position to the bottom right of the large digit and not at the last regular text write position. [col] can have values from 0x01 to 0x12 (1 to 18). [digit] has values from 0x00 to 0x09 (0 to 9).

Remembered No

Compatibility A *Best used with displays that have 4 rows*

5.2.9 Initialize Narrow Vertical Bar Graph

Syntax	Hexadecimal	0xFE 0x73
	Decimal	254 115
	ASCII	254 “s”

Description This command defines the 8 special / user characters to be blocks suitable for use in drawing narrow (2 pixel) vertical bar graphs. Any previously existing definitions will be lost. Once this command has been issued, any number of vertical bar graphs may be drawn unless the characters are re-defined by another command.

Remembered No

Compatibility A

5.2.10 Initialize Wide Vertical Bar Graph

Syntax	Hexadecimal	0xFE 0x76
	Decimal	254 118
	ASCII	254 “v”

Description

This command defines the 8 special / user characters to be blocks suitable for use in drawing wide (5 pixel) vertical bar graphs. Any previously existing definitions will be lost. Once this command has been issued, any number of vertical bar graphs may be drawn unless the characters are re-defined by another command.

Remembered No

Compatibility A

5.2.11 Draw Horizontal Bar Graph

Syntax

Hexadecimal 0xFE 0x7C [column] [row] [dir] [length]
Decimal 254 124 [column] [row] [dir] [length]
ASCII 254 “|” [column] [row] [dir] [length]

Parameters

Parameter	Size	Description
column	1	The column number (1 - 16)
row	1	The row number (1 - 2)
dir	1	The direction value (0 or 1)
length	1	The length (0 - 80)

Description

Draws a horizontal bar graph in [row] starting at [column] with a length of [length] pixels. [row] may have a value of 1 or 2, [column] may range from 1 to 16 and length may be from 0 to 80 if the graph can extend the full width of the screen. Each column is 5 pixels wide (spaces between the columns do not count). [dir] specifies the direction: 0 goes from left to right, 1 or any other values go from right to left.

Remembered No

Compatibility A

5.2.12 Remember Custom Character

Syntax

Hexadecimal 0xFE 0xC2 [address] [definiton]
Decimal 254 194 [address] [definiton]

Parameters

Parameter	Size	Description
address	1	The address (0x00 to 0x07)
definiton	8	The character definition

Description — This command will store a custom character. Most common use of this command is to define characters for the startup screen. This command does not affect or alter the current custom characters that have been initialized or defined. The syntax is identical to the command Define Custom Character 254 78.

Remembered — Always

Compatibility — A

6 Display Functions

6.1 Introduction

The commands listed in this chapter are functions of the display such as contrast, brightness and start up screens.

6.2 Command List

6.2.1 Backlight On

Syntax

Hexadecimal	0xFE 0x42 [minutes]
Decimal	254 66 [minutes]
ASCII	254 “B” [minutes]

Parameters

Parameter	Size	Description
minutes	1	Backlight value (0 to 90)

Description This command turns the backlight on for [minutes], with [minutes] maximum value of 90. If [minutes] is set to zero (0), the backlight will never turn off. When this command is sent while the Remember function is on, it will be remembered. E.g. This command is sent with a value of 1 for [minutes] and Remember is on. When the power is cycled, the module will remember to turn off backlight after 1 minute. In the case for a VFD and PLED display where a backlight does not exist, this is referenced to the display being turned on.

NOTE The factory default for backlight is on

Remembered Yes

Compatibility A

6.2.2 Backlight Off

Syntax

Hexadecimal	0xFE 0x46
Decimal	254 70
ASCII	254 “F”

Description This command turns the backlight off. When this command is sent while the Remember function is on, it will be remembered. When power is cycled, the module will remember to turn off backlight. In the case for a VFD and PLED display where a backlight does not exist, this is referenced to the display being turned off.

Remembered Yes

Compatibility A

6.2.3 Set Contrast

Syntax

Hexadecimal	0xFE 0x50 [contrast]
Decimal	254 80 [contrast]
ASCII	254 “P” [contrast]

Parameters

Parameter	Size	Description
contrast	1	Contrast value (0 to 255)

Description

This command sets the display's contrast to [contrast], where [contrast] is a value between 0x00 and 0xFF (between 0 to 255). Lower values cause 'on' elements in the display area to appear lighter, while higher values cause 'on' elements to appear darker. Lighting conditions will affect the actual value used for optimal viewing. Individual display modules will also differ slightly from each other in appearance. In addition, values for optimal viewing while the display backlight is on may differ from values used when backlight is off.

This command does not save the [contrast] value, and is lost after power down; but this command has the option of remembering the settings when issued with the Remember function 'on' . When this is the case, this command acts like the Set and Save Contrast command 254 145.

Remembered Yes

Compatibility L

6.2.4 Set and Save Contrast

Syntax

Hexadecimal	0xFE 0x91 [contrast]
Decimal	254 145 [contrast]

Parameters

Parameter	Size	Description
contrast	1	The contrast setting (0 to 255)

Description

This command works in exactly the same way as the "Set Contrast" command when sent with the Remember function on. When this command is sent, not only does it set the contrast with [contrast] value, but also saves this value in the non-volatile memory so that at power cycle, this setting is restored.

Remembered Always

Compatibility L

6.2.5 Set and Save Backlight Brightness

Syntax	Hexadecimal	0xFE 0x98 [backlight]
	Decimal	254 152 [backlight]

Parameters

Parameter	Size	Description
backlight	1	Backlight setting (0 to 255)

Description This commands sets and saves [brightness] as default. In the case of a PLED where a backlight does not exist, this is referenced to the brightness of the display.

Remembered Always

Compatibility L,P

6.2.6 Set Backlight Brightness

Syntax	Hexadecimal	0xFE 0x99 [backlight]
	Decimal	254 153 [backlight]

Parameters

Parameter	Size	Description
backlight	1	Backlight setting (0 to 255)

Description This command sets the backlight settings according to [backlight]. In the case of a PLED where a backlight does not exist, this is referenced to the brightness of the display.

Remembered Yes

Compatibility L,P

6.2.7 Set VFD Brightness

Syntax	Hexadecimal	0xFE 0x59 [brightness]
	Decimal	254 89 [brightness]
	ASCII	254 “Y” [brightness]

Parameters

Parameter	Size	Description
brightness	1	VFD brightness value (0 to 3)

Description

This commands sets the VFD brightness to [brightness], where [brightness] is a value between 0x00 to 0x03 (between 0 and 3), acording to the table below:

HexValues	**Brightness**
0x00	25%
0x01	50%
0x02	75%
0x03	100%

Remembered Yes

Compatibility V

6.2.8 Set and Save VFD Brightness

Syntax	Hexadecimal	0xFE 0x91 [VFD Brightness]
	Decimal	254 145 [VFD Brightness]

Parameters

Parameter	Size	Description
VFD Brightness	1	The VFD brightness setting (0 to 3)

Description

This command works in exactly the same way as the “Set VFD Brightness” command when sent with the Remember function on. When this command is sent, not only does it set the VFD brightness with [VFD brightness] value, but also saves this value in the non-volatile memory so that at power cycle, this setting is restored.

Remembered Always

Compatibility V

7 GPO Functions

7.1 Introduction

The commands listed in this chapter describe the functionality and control of the general purpose outputs.

7.2 Command List

7.2.1 General Purpose Output Off

Syntax	Hexadecimal	0xFE 0x56 [gpo#]
	Decimal	254 86 [gpo#]
	ASCII	254 “V” [gpo#]

Parameters

Parameter	Size	Description
gpo#	1	GPO number is from (1 - 3)

Description This command turns OFF any of the general purpose outputs. [gpo#] is from 1 to 3. Note that OFF means that the output floats.

Remembered Yes

Compatibility A

7.2.2 General Purpose Output On

Syntax	Hexadecimal	0xFE 0x57 [gpo#]
	Decimal	254 87 [gpo#]
	ASCII	254 “W” [gpo#]

Parameters

Parameter	Size	Description
gpo#	1	GPO number is from (1 - 3)

Description This command turns ON any of the general purpose outputs. [gpo#] is from 1 to 3. Note that ON means that the output is pulled low (ground via 240 Ohms).

Remembered Yes

Compatibility	A

7.2.3 Remember GPO

Syntax	Hexadecimal	0xFE 0xC3 [gpo#] [state]
	Decimal	254 195 [gpo#] [state]

Parameters

Parameter	Size	Description
gpo#	1	GPO number (1 - 3)
state	1	state (0 or 1)

Description: This command will set the startup state for individual GPOs . When the device is powered up the next time, the GPOs will be set to the values from this command.
This command does not affect the current state of the GPOs , only at power up.

Remembered	Always
Compatibility	A

8 Communications Section

8.1 Introduction

The commands listed in this chapter describe how to configure data flow on the RS232 and TTL port. The display has built in flow control which may be useful when long strings of text are downloaded to the display. Flow control is enabled or disabled by two commands. If flow control is enabled, the display will return an "almost full" message (0xFE) to the controller when its internal buffer fills to a defined level, and an "almost empty" message (0xFF) when the buffer contents drop to a defined level.

8.2 Command List

8.2.1 Enter Flow Control Mode

Syntax	Hexadecimal	0xFE 0x3A [full] [empty]
	Decimal	254 58 [full] [empty]
	ASCII	254 “:” [full] [empty]

Parameters

Parameter	Size	Description
full	1	The full byte number (0 to 80)
empty	1	The empty byte number (0 to 80)

Description

This command enables flow control. When the buffer fills so that only [full] bytes are available, the display will return an “almost full” message (0xFE) to the host controller. When the buffer empties so that only [empty] bytes remain, the display will return an “almost empty” message (0xFF) to the host controller.

The display will return the “almost full” message for every byte sent to the display until the used buffer space once more drops below the [full] level.

Whether the user is in ‘flow control mode’ or not, the module will ignore display or command bytes which would overrun the buffer. While in ‘flow control mode’ the unit will return 0xFE when buffer is almost full even though it may have already thrown rejected data away. The buffer size for the display is 80 bytes.

When using this command in an application, selection of the value for the buffer [full] should be considered very carefully. This is a critical aspect to be able to use this feature to it’s full potential. When using a host system or PC which contains a FIFO, the user should set the value of [full] equal to or greater than the size of the FIFO. The reason for this is that the FIFO may be full when the host system receives 0xFE. In the case of 16550 UART the size at its maximum is 16, therefore the value of [full] should be set to 16 or greater.

Remembered No

Compatibility A

8.2.2 Exit Flow Control Mode

Syntax	Hexadecimal	0xFE 0x3B
	Decimal	254 59
	ASCII	254 “;”

Description

This command turns off flow control. Bytes may overflow the buffer without warning.

Remembered	No
Compatibility	A

9 Miscellaneous Commands

9.1 Introduction

The commands listed in this chapter don’t readily fit in any of the other categories.

9.2 Command List

9.2.1 Read Version Number

Syntax	Hexadecimal	0xFE 0x36
	Decimal	254 54
	ASCII	254 “6”

Description	This command will return the firmware version number of the module. This command is only available in TTL mode.
Remembered	No

9.2.2 Read Module Type

Syntax	Hexadecimal	0xFE 0x37
	Decimal	254 55
	ASCII	254 “7”

Description

This command is only available in TTL mode. The unit will return 2 bytes. The value of the 2 bytes returned will determine what type and size your X-Board is configured for. Values for various X-Board modules at the time of this publication are as follows:

Value Returned	Display Type	Display Size
0xC8	LCD	1x16
0xC9	LCD	2x8
0xCA	LCD	2x16
0xCB	LCD	2x20
0xCC	LCD	2x24
0xCD	LCD	2x40
0xCE	LCD	4x16
0xCF	LCD	4x20
0xD0	SVFD	1x16
0xD1	SVFD	2x8
0xD2	SVFD	2x16
0xD3	SVFD	2x20
0xD4	SVFD	2x24
0xD5	SVFD	2x40
0xD6	SVFD	4x16
0xD7	SVFD	4x20
0xD8	PLED	1x16
0xD9	PLED	2x8
0xDA	PLED	2x16
0xDB	PLED	2x20
0xDC	PLED	2x24
0xDD	PLED	2x40
0xDE	PLED	4x16
0xDF	PLED	4x20
0xE0	NVFD	1x16
0xE1	NVFD	2x8
0xE2	NVFD	2x16
0xE3	NVFD	2x20
0xE4	NVFD	2x24
0xE5	NVFD	2x40
0xE6	NVFD	4x16
0xE7	NVFD	4x20

Remembered No

Compatibility A

9.2.3 Remember

Syntax	Hexadecimal	0xFE 0x93 [value]
	Decimal	254 147 [value]

Parameters

Parameter	Size	Description
value	1	Value (0 or 1)

Description

[value] specifies Remember functionality; if set to 0, Remember is off and if set to a non-zero value, Remember is on. This command allows a number of settings, such as cursor state, backlight, etc. to automatically be stored to non-volatile memory so they become new defaults. If any of the following commands are sent and the Remember function is ‘on’, they will be automatically remembered:

- Auto Line Wrap On/Off
- Auto Scroll On/Off
- Backlight On/Off
- Set Backlight Brightness
- Set Contrast
- General Purpose Output On/Off
- Turn On/Off Block (Blinking) Cursor
- Turn On/Off Underline Cursor
- Set Backlight/PLED brightness

In the Command Summary, the above commands are marked with ‘R’ which denotes that Remember function affects the command.

NOTE

Writing to non-volatile memory is time consuming and slows down the operation of the display.

WARNING

Non-volatile memory has a ‘write limit’ and may only be changed approximately 100,000 times.

Remembered Always

Compatibility A

10 Command Summary

10.1 Text Commands

Description	Syntax		Page
Set Cursor Position	Hexadecimal	0xFE 0x47 [column] [row]	24
	Decimal	254 71 [column] [row]	
	ASCII	254 “G” [column] [row]	
Send Cursor Home	Hexadecimal	0xFE 0x48	25
	Decimal	254 72	
	ASCII	254 “H”	
Turn On Underline Cursor	Hexadecimal	0xFE 0x4A	25
	Decimal	254 74	
	ASCII	254 “J”	
Turn Off Underline Cursor	Hexadecimal	0xFE 0x4B	26
	Decimal	254 75	
	ASCII	254 “K”	
Cursor Left	Hexadecimal	0xFE 0x4C	26
	Decimal	254 76	
	ASCII	254 “L”	
Cursor Right	Hexadecimal	0xFE 0x4D	26
	Decimal	254 77	
	ASCII	254 “M”	
Turn On Blinking Block Cursor	Hexadecimal	0xFE 0x53	28
	Decimal	254 83	
	ASCII	254 “S”	
Turn Off Blinking Block Cursor	Hexadecimal	0xFE 0x54	28
	Decimal	254 84	
	ASCII	254 “T”	
Clear Display	Hexadecimal	0xFE 0x58	29
	Decimal	254 88	
	ASCII	254 “X”	

Description	Syntax		Page
Auto Line Wrap On	Hexadecimal	0xFE 0x43	24
	Decimal	254 67	
	ASCII	254 “C”	
Auto Line Wrap Off	Hexadecimal	0xFE 0x44	24
	Decimal	254 68	
	ASCII	254 “D”	
Auto Scroll On	Hexadecimal	0xFE 0x51	27
	Decimal	254 81	
	ASCII	254 “Q”	
Auto Scroll Off	Hexadecimal	0xFE 0x52	27
	Decimal	254 82	
	ASCII	254 “R”	

10.2 Bar Graphs and Special Characters

Description	Syntax		Page
Load Startup Screen	Hexadecimal	0xFE 0x40 [characters]	30
	Decimal	254 64 [characters]	
	ASCII	254 “@” [characters]	
Initialize Wide Vertical Bar Graph	Hexadecimal	0xFE 0x76	36
	Decimal	254 118	
	ASCII	254 “v”	
Initialize Horizontal Bar Graph	Hexadecimal	0xFE 0x68	33
	Decimal	254 104	
	ASCII	254 “h”	
Initialize Narrow Vertical Bar Graph	Hexadecimal	0xFE 0x73	36
	Decimal	254 115	
	ASCII	254 “s”	
Draw Vertical Bar Graph	Hexadecimal	0xFE 0x3D [column] [height]	29
	Decimal	254 61 [column] [height]	
	ASCII	254 “=” [column] [height]	
Draw Horizontal Bar Graph	Hexadecimal	0xFE 0x7C [column] [row] [dir] [length]	37
	Decimal	254 124 [column] [row] [dir] [length]	
	ASCII	254 “|” [column] [row] [dir] [length]	

Description	Syntax		Page
Initialize Medium Digits	Hexadecimal	0xFE 0x6D	33
	Decimal	254 109	
	ASCII	254 “m”	
Draw Medium Digits	Hexadecimal	0xFE 0x6F [row] [column] [digit]	34
	Decimal	254 111 [row] [column] [digit]	
	ASCII	254 “o” [row] [column] [digit]	
Define Custom Character	Hexadecimal	0xFE 0x4E [address] [definiton]	31
	Decimal	254 78 [address] [definiton]	
	ASCII	254 “N” [address] [definiton]	
Remember Custom Character	Hexadecimal	0xFE 0xC2 [address] [definiton]	37
	Decimal	254 194 [address] [definiton]	

10.3 Display Functions

Description	Syntax		Page
Backlight On	Hexadecimal	0xFE 0x42 [minutes]	38
	Decimal	254 66 [minutes]	
	ASCII	254 “B” [minutes]	
Backlight Off	Hexadecimal	0xFE 0x46	39
	Decimal	254 70	
	ASCII	254 “F”	
Set Contrast	Hexadecimal	0xFE 0x50 [contrast]	39
	Decimal	254 80 [contrast]	
	ASCII	254 “P” [contrast]	
Set Backlight Brightness	Hexadecimal	0xFE 0x99 [backlight]	41
	Decimal	254 153 [backlight]	
Set and Save Contrast	Hexadecimal	0xFE 0x91 [contrast]	42
	Decimal	254 145 [contrast]	
Set and Save Backlight Brightness	Hexadecimal	0xFE 0x98 [backlight]	41
	Decimal	254 152 [backlight]	

10.4 GPO Functions

Description	Syntax		Page
General Purpose Output On	Hexadecimal	0xFE 0x57 [gpo#]	43
	Decimal	254 87 [gpo#]	
	ASCII	254 “W” [gpo#]	
General Purpose Output Off	Hexadecimal	0xFE 0x56 [gpo#]	43
	Decimal	254 86 [gpo#]	
	ASCII	254 “V” [gpo#]	
Remember GPO	Hexadecimal	0xFE 0xC3 [gpo#] [state]	44
	Decimal	254 195 [gpo#] [state]	

10.5 Communications Section

Description	Syntax		Page
Enter Flow Control Mode	Hexadecimal	0xFE 0x3A [full] [empty]	44
	Decimal	254 58 [full] [empty]	
	ASCII	254 “:” [full] [empty]	
Exit Flow Control Mode	Hexadecimal	0xFE 0x3B	45
	Decimal	254 59	
	ASCII	254 “;”	

10.6 Miscellaneous Commands

Description	Syntax		Page
Read Serial Number	Hexadecimal	0xFE 0x35	**??**
	Decimal	254 53	
	ASCII	254 “5”	
Read Module Type	Hexadecimal	0xFE 0x37	46
	Decimal	254 55	
	ASCII	254 “7”	
Read Version Number	Hexadecimal	0xFE 0x36	46
	Decimal	254 54	
	ASCII	254 “6”	
Set Serial Number	Hexadecimal	0xFE 0x34 [serial]	**??**
	Decimal	254 52 [serial]	
	ASCII	254 “4” [serial]	

Description	Syntax		Page
Remember	Hexadecimal	0xFE 0x93 [value]	48
	Decimal	254 147 [value]	

11 Appendix: Specifications and Options

11.1 Specifications

Table 55: Environmental Specifications

	Standard Temperature
Operating Temperature	-20°C to +70°C
Storage Temperature	-40°C to +80°C
Operating Relative Humidity	90% max non-condensing

Table 56: Electrical Specifications

Supply Voltage	4.75 - 5.25Vdc
Supply Current	3 mA typical (*no display attached*)

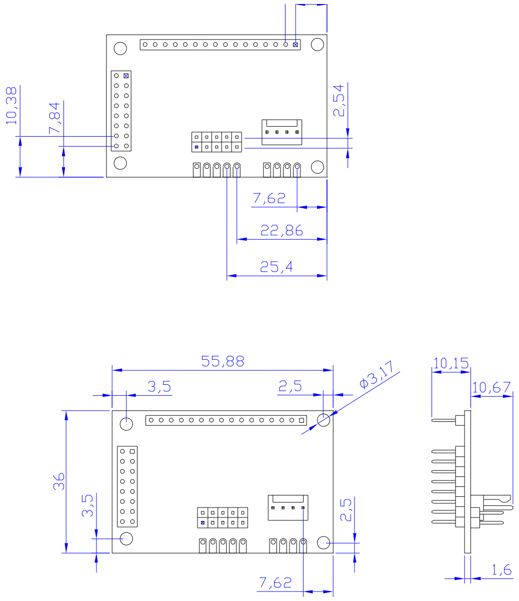


Figure 23: Physical Layout

12 Appendix: Glossary

12.1 On Numbers

Like all computerized devices, the display operates with commands and values in the form of binary numbers. These binary numbers are arranged in 8 digit (i.e., 8 bit) groups called bytes. The decimal value of a byte may have any value from 0 to 255.

Bytes are usually specified in either decimal or hexadecimal (base 16) form for convenience, since binary numbers are confusing to deal with directly. Hexadecimal (hex) numbers are particularly convenient because exactly two hexadecimal digits make up one byte, each hex digit representing 4 binary digits (4 bits) as shown in Table 57.

Table 57: Hex Value Table

Binary	Hex	Decimal	Binary	Hex	Decimal
0000	0	0	1000	8	8
0001	1	1	1001	9	9
0010	2	2	1010	A	10
0011	3	3	1011	B	11
0100	4	4	1100	C	12
0101	5	5	1101	D	13
0110	6	6	1110	E	14
0111	7	7	1111	F	15

Based on Table 57, the byte 01001011 can be represented in hex as 4B, which is usually written as any of 4Bh, 4BH, 4B hex or 0x4B. The numbers can also be expressed in decimal form if preferred.

12.2 ASCII Characters

Since computers deal internally with numbers only, but externally with both letters and numbers, several schemes were developed to 'map' written characters to numeric values. One such scheme has become universal, the American Standard Code for Information Interchange, or ASCII. ASCII tables are readily available from a number of sources. A few examples are shown in Table 58

Table 58: Example of an ASCII Table

The letter	A	has a value of	65 Decimal or	41 Hex
The letter	a	has a value of	97 Decimal or	61 Hex
The number	0	has a value of	48 Decimal or	30 Hex
The number	9	has a value of	57 Decimal or	39 Hex

This gives rise to the possibility of confusion when parameters are being set on the display. For example, the GPO ON and OFF commands use a number to indicate which GPO is being controlled. We're told that acceptable values are 1 to 3. **All such parameters must use numeric values (i.e., the actual byte values)**. If we send the ASCII number 0 by mistake it will actually give the value 48 decimal (30 hex) to the parameter, which is wrong.

Table 59: Appendix: Glossary

ASCII	American Standard Code for Information Interchange. A 7 bit binary code representing the English alphabet, decimal numbers and common punctuation marks. Also includes control characters such as carriage return or end of text. An 8 bit superset of the standard ASCII codes is often used today to include foreign characters and other symbols. These supersets are often called extended ASCII character sets.
Backlight	A backlit display is illuminated from behind to provide nighttime and improved daytime readability.
Binary Number	The (data and signaling) bit transmission rate of an RS-232 device.
Bit	A number written using binary notation which only uses zeros and ones.
Bitmap	A representation, consisting of rows and columns of dots, of a graphics image in computer memory. The value of each dot (whether it is filled in or not) is stored in one or more bits of data.
Byte	A grouping of eight binary bits
CCFL	Cold Cathode Fluorescent Lamp. A high brightness backlighting source consists of a fluorescent tube powered by a high voltage A.C. source.
Configuration	The way a system is set up, or the assortment of components that make up the system. Configuration can refer to either hardware or software, or the combination of both.
Contrast	The ratio of luminance between the light state of the display to the dark state of the display.
Controller	The micro-controller or PC used to control the Matrix Orbital display unit.
DB-9	The designation of a connector used in the RS-232 interface: 9 pin connector
Firmware	Software (programs or data) that has been written onto read-only memory (ROM). Firmware is a combination of software and hardware. ROMs, PROMs and EPROMs and flash EEPROMs that have data or programs recorded on them are firmware.
Font	A design for a set of characters. A font is the combination of typeface and other qualities, such as size, pitch, and spacing.
Font Metric	A definition of where font is to be placed, such as margins and spacing between characters and lines.

Hexadecimal	Refers to the base-16 number system, which consists of 16 unique symbols: the numbers 0 to 9 and the letters A to F. For example, the decimal number 15 is represented as F in the hexadecimal numbering system. The hexadecimal system is useful because it can represent every byte (8 bits) as two consecutive hexadecimal digits. It is easier for humans to read hexadecimal numbers than binary numbers.
I^{2}C	Short for Inter-IC, a type of bus designed by Phillips Semiconductors in the early 1980s, which is used to connect integrated circuits (ICs). I^{2}C is a multi-master bus, which means that multiple chips can be connected to the same bus and each one can act as a master by initiating a data transfer.
Interface	A means by which two systems interact.
LCD	Liquid Crystal Display
Module Type Value	This refers to the model number of the module.
PLED	Polymer Light Emitting Diode. Polymers are substances formed by a chemical reaction in which two or more molecules combine to form larger molecules. PLEDs are thin film displays that are created by sandwiching an undoped conjugated polymer between two proper electrodes at a short distance. The polymer emits light when exposed to electricity.
Pixel	The smallest individually controllable element of a display.
Pre-Generated Fonts	Pre-determined fonts which can be downloaded into graphic liquid crystal displays.
Primitive	A low-level object or operation from which higher-level, more complex objects and operations can be constructed. In graphics, primitives are basic elements, such as lines, curves, and polygons, which you can combine to create more complex graphical images.
RS-232	Short for recommended standard-232C, a standard interface approved by the Electronic Industries Association (EIA) for connecting serial devices.
Scroll	To view consecutive lines of data on the display screen. The term scroll means that once the screen is full, each new line appears at the bottom edge of the screen and all other lines move up one position.
Serial Number	A number that is one of a series and is used for identification of the module.
Serial Port	A port, or interface, that can be used for serial communication, in which only 1 bit is transmitted at a time.

Version Number	This refers to the firmware revision number of the module.
VFD	Vacuum Fluorescent Display
Volatile Memory	Temporary memory. Once the power supply is turned off volatile memory is then erased.