



LCD2041
Technical Manual

Revision: 2.1

Contents

| | |
|--|-----------|
| Contents | ii |
| 1 Getting Started | 1 |
| 1.1 Display Options Available | 1 |
| 1.2 Accessories | 2 |
| 1.3 Features | 3 |
| 1.4 Connecting to a PC | 5 |
| 1.5 Installing the Software | 6 |
| 1.5.1 uProject | 6 |
| 2 Hardware Information | 7 |
| 2.1 DB-9 Connector | 7 |
| 2.1.1 Power Through DB-9 Jumper | 8 |
| 2.2 Power/Data Connector | 10 |
| 2.2.1 Legacy Data Connector Jumpers | 11 |
| 2.3 Protocol Select Jumpers | 11 |
| 2.4 General Purpose Outputs | 13 |
| 2.5 Manual Override | 13 |
| 3 Troubleshooting | 15 |
| 3.1 The display does not turn on when power is applied. | 15 |
| 3.2 The display module is not communicating. | 15 |
| 3.3 The display module is communicating, however text cannot be displayed. | 16 |
| 4 Communications | 16 |
| 4.1 Introduction | 16 |
| 4.1.1 I ² C Communication Summary | 16 |
| 4.1.2 I ² C Transaction Example | 17 |
| 4.1.3 Serial Communication | 17 |
| 4.2 Changing the I ² C Slave Address | 18 |
| 4.3 Changing the Baud Rate | 18 |
| 4.4 Setting a Non-Standard Baud Rate | 19 |
| 5 Text | 20 |
| 5.1 Introduction | 20 |
| 5.1.1 Character Set | 21 |
| 5.1.2 Control Characters | 21 |
| 5.2 Auto Scroll On | 21 |
| 5.3 Auto Scroll Off | 22 |
| 5.4 Clear Screen | 22 |
| 5.5 Changing the Startup Screen | 22 |
| 5.6 Set Auto Line Wrap On | 23 |
| 5.7 Set Auto Line Wrap Off | 23 |
| 5.8 Set Cursor Position | 23 |

| | | |
|----------|---|-----------|
| 5.9 | Go Home | 24 |
| 5.10 | Move Cursor Back | 24 |
| 5.11 | Move Cursor Forward | 24 |
| 5.12 | Underline Cursor On | 25 |
| 5.13 | Underline Cursor Off | 25 |
| 5.14 | Blinking Block Cursor On | 25 |
| 5.15 | Blinking Block Cursor Off | 25 |
| 6 | Special Characters | 26 |
| 6.1 | Introduction | 26 |
| 6.2 | Creating a Custom Character | 26 |
| 6.3 | Saving Custom Characters | 27 |
| 6.4 | Loading Custom Characters | 28 |
| 6.5 | Save Startup Screen Custom Characters | 28 |
| 6.6 | Initialize Medium Number | 29 |
| 6.7 | Place Medium Numbers | 29 |
| 6.8 | Initialize Large Numbers | 30 |
| 6.9 | Place Large Number | 30 |
| 6.10 | Initialize Horizontal Bar | 30 |
| 6.11 | Place Horizontal Bar Graph | 31 |
| 6.12 | Initialize Narrow Vertical Bar | 31 |
| 6.13 | Initialize Wide Vertical Bar | 31 |
| 6.14 | Place Vertical Bar | 32 |
| 7 | General Purpose Output | 32 |
| 7.1 | Introduction | 32 |
| 7.2 | General Purpose Output Off | 33 |
| 7.3 | General Purpose Output On | 33 |
| 7.4 | Set Startup GPO state | 33 |
| 8 | Display Functions | 34 |
| 8.1 | Introduction | 34 |
| 8.2 | Display On | 34 |
| 8.3 | Display Off | 34 |
| 8.4 | Set Brightness | 35 |
| 8.5 | Set and Save Brightness | 35 |
| 8.6 | Set Contrast | 35 |
| 8.7 | Set and Save Contrast | 36 |
| 9 | Data Security | 37 |
| 9.1 | Introduction | 37 |
| 9.2 | Set Remember | 37 |
| 9.3 | Data Lock | 38 |
| 9.4 | Set and Save Data Lock | 39 |
| 9.5 | Write Customer Data | 39 |
| 9.6 | Read Customer Data | 39 |

| | | |
|-----------|-------------------------------------|-----------|
| 10 | Miscellaneous | 39 |
| 10.1 | Introduction | 39 |
| 10.2 | Read Version Number | 40 |
| 10.3 | Read Module Type | 40 |
| 11 | Command Summary | 42 |
| 11.1 | Communications | 42 |
| 11.2 | Text | 42 |
| 11.3 | Special Characters | 43 |
| 11.4 | General Purpose Output | 44 |
| 11.5 | Display Functions | 44 |
| 11.6 | Data Security | 45 |
| 11.7 | Miscellaneous | 45 |
| 11.8 | Command By Number | 45 |
| 12 | Appendix | 47 |
| 12.1 | Specifications | 47 |
| 12.1.1 | Environmental | 47 |
| 12.1.2 | Electrical | 47 |
| 12.2 | Optical Characteristics | 47 |
| 12.3 | Physical Layout | 49 |
| 12.4 | Ordering Information | 50 |
| 12.5 | Definitions | 50 |
| 12.6 | Contacting Matrix Orbital | 51 |
| 12.7 | Revision History | 51 |

1 Getting Started

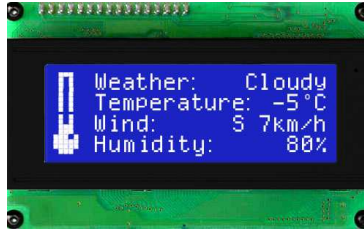


Figure 1: LCD2041-WB

The LCD2041 is an intelligent LCD display designed to decrease development time by providing an instant solution to any project. With the ability to communicate via serial RS-232/TTL and I²C protocols, the versatile LCD2041 can be used with virtually any controller. The ease of use is further enhanced by an intuitive command structure to allow display settings such as backlight brightness, contrast and baud rate to be software controlled. Additionally, up to thirty-two custom characters such as character sets for bar graphs, medium and large numbers may be stored in the non-volatile memory to be easily recalled and displayed at any time.

1.1 Display Options Available

The LCD2041 comes in a wide variety of colors including the standard yellow/green or inverse yellow, the popular blue/white and the crisp white/grey as well as inverse red which is excellent for viewing at night. Extended voltage, and temperature options are also available, to allow you to select the display which will best fit your project needs.

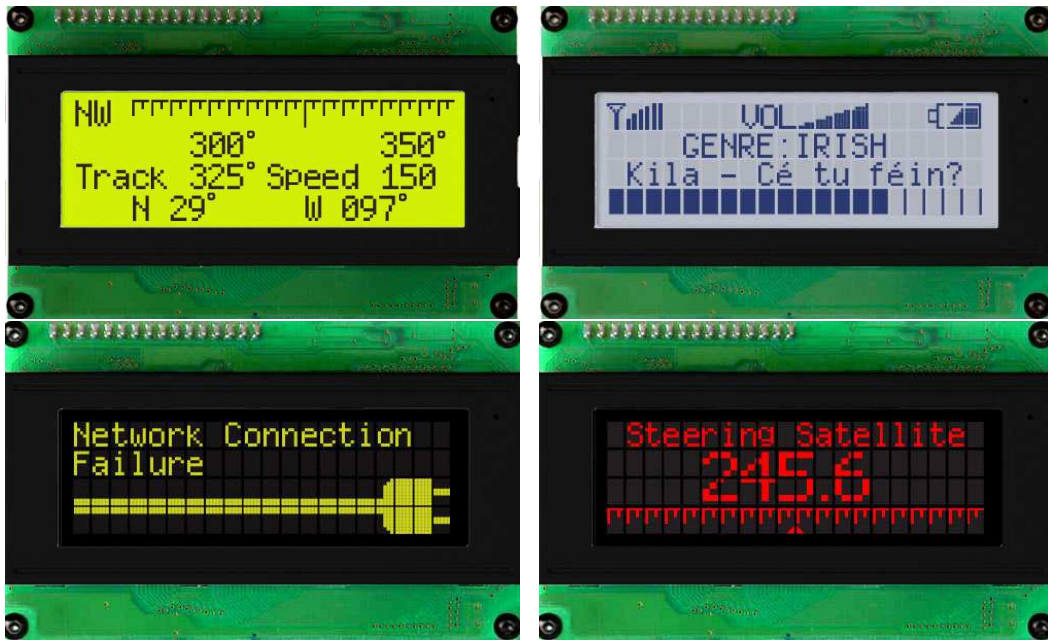


Figure 2: LCD2041 Options

1.2 Accessories

NOTE Matrix Orbital provides all the interface accessories needed to get your display up and running. You will find these accessories and others on our e-commerce website at <http://www.matrixorbital.com>. To contact a sales associate see Section 12.6 on page 51 for contact information.



Figure 3: 5V Power Cable Adapter



Figure 4: 12V Power Cable Adapter (V/VPT Models)



Figure 5: Breadboard Cable

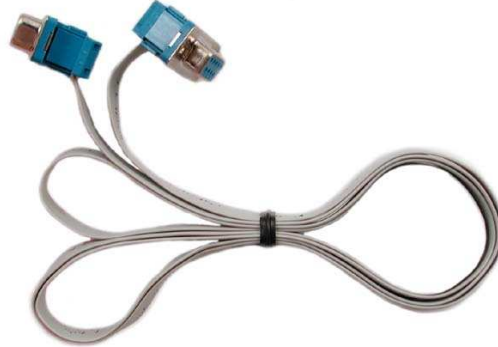


Figure 6: Serial Cable 4FT

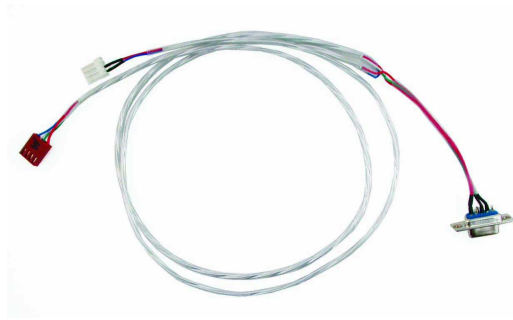


Figure 7: Communication and 5V Power Cable

1.3 Features

- 20 column by 4 line alphanumeric liquid crystal display
- Selectable communication protocol, serial at RS-232 or TTL levels or I²C
- Three, 5V -20mA, general purpose outputs for a variety of applications
- Lightning fast communication speeds, up to 57.6 kbps for RS-232 and 100 kbps for I²C
- Default 19.2 kbps serial communication speed



Figure 8: Aluminum Mountings

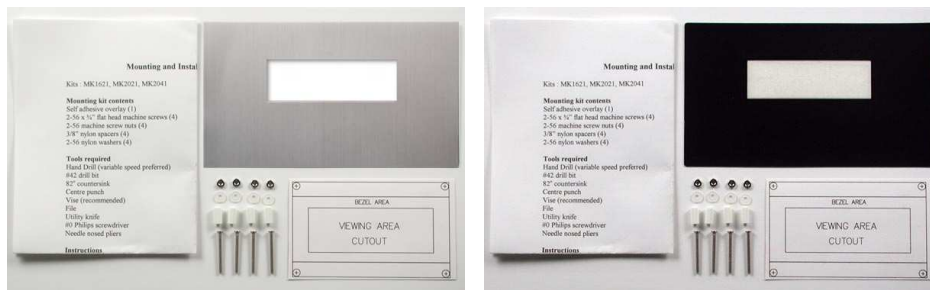


Figure 9: Mounting Kits

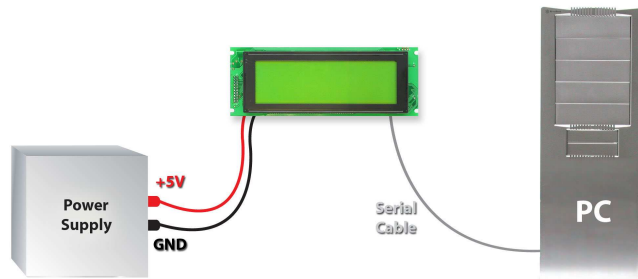
- Extended temperature available for extreme environments of -20C to 70C
- Extended voltage and efficient power supply available
- Built in font with provision for up to 8 user defined characters
- Use of up to 127 modules on the same 2 wire I²C interface
- Ability to add a customized splash / startup screen
- Software controlled contrast and brightness with configurable time-out setting up to 90 minutes
- Horizontal or vertical bar graphs
- Fits Matrix Orbital's mountings without any modifications

1.4 Connecting to a PC

The LCD2041 connects seamlessly to a PC and it is an excellent means of testing the functionality. To connect your display to a PC, you will require a standard RS-232 9-pin serial cable such as the one pictured in *figure 6 on page 3*, as well as a modified 5V power adapter such as the one pictured in *figure 3 on page 2*.

In order to connect your display to a personal computer follow these easy instructions:

1. Plug the serial cable into the com port you wish to use.
2. Connect the modified 5V power adapter to a power lead from your PC power supply (you will have to open your computer case).
3. Connect the serial cable to the DB-9 connector on the back of the display.
4. Connect the 5V power adapter to the 4-pin connector on the back of the display.



WARNING DO NOT use the standard floppy drive power connector, as this will not provide you with the correct voltage and will damage the display module.

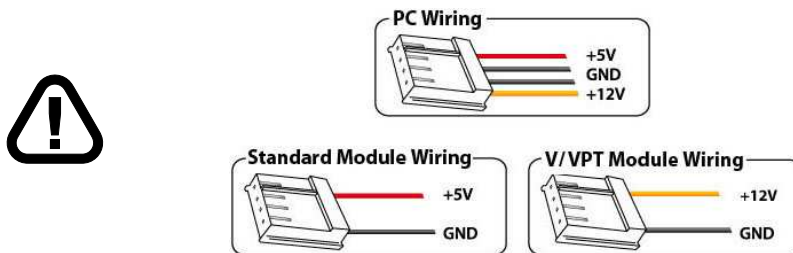


Figure 10: PC vs Matrix Orbital Display Module Wiring

1.5 Installing the Software

1.5.1 uProject

uProject was designed by Matrix Orbital to provide a simple and easy to use interface that will allow you to test all of the features of our alpha numeric displays.

To install uProject from the Matrix Orbital CD, follow the following steps:

1. Insert the Matrix Orbital CD-ROM into the CD drive
2. Locate the file, *uProject.exe*, which should be in the “CD-drive:\Download” directory.
3. Copy *uProject.exe* to a directory that you wish to run it from.
4. Double click on "uProject.exe"

Be sure to check the information selected in the COM Setup the first time uProject is run. Once this information is entered correctly the program can be used to control all functions of the graphic display.



- Comport** The serial port the display is plugged in to.
- Baudrate** The communication speed the display module is set to. (Default 19,200)

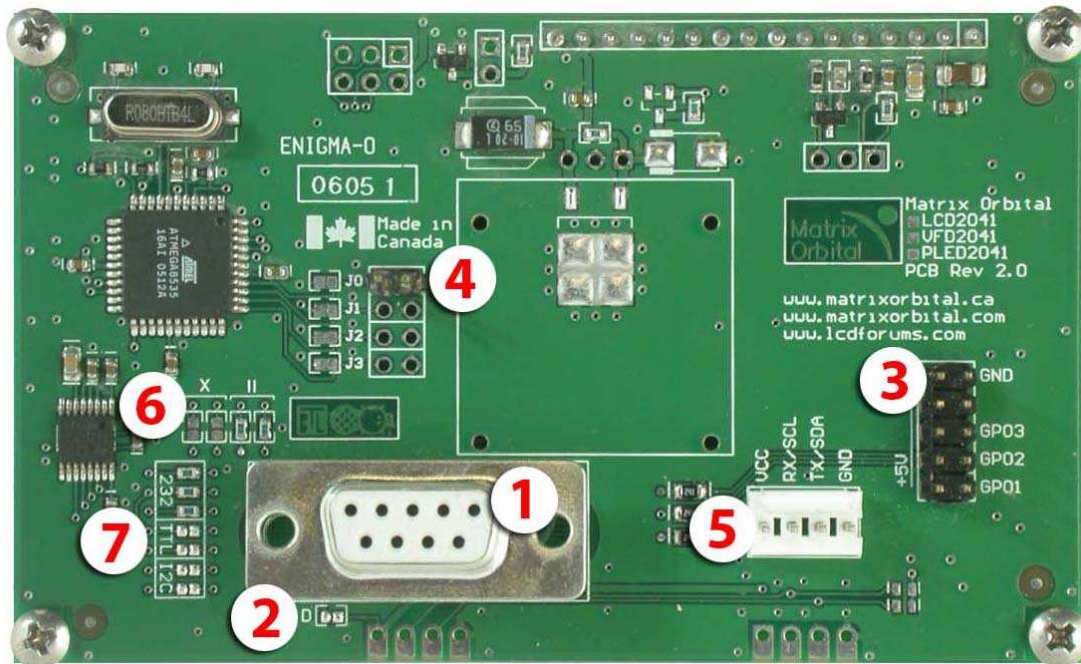
Figure 11: uProject Settings

NOTES

- uProject and other alphanumeric software may also be downloaded from Matrix Orbital's support site at http://www.matrixorbital.ca/software/software_alpha/

2 Hardware Information

Refer to the following diagram for this chapter:

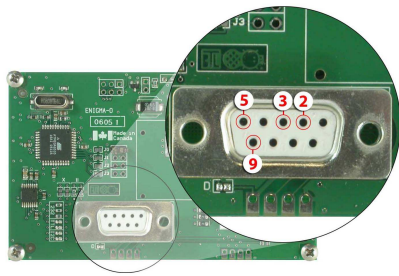


- | | |
|----------------------------|---------------------------|
| 1 DB-9 Connector | 5 Power / Data Connector |
| 2 Power Through DB9 Jumper | 6 Legacy Connector Jumper |
| 3 GPOs | 7 Protocol Select Jumpers |
| 4 Manual Override | |

Figure 12: LCD2041

2.1 DB-9 Connector

The LCD2041 provides a *DB-9 Connector* to readily interface with serial devices which use the EIA232 standard signal levels of $\pm 12V$ to $\pm 12V$. It is also possible to communicate at TTL levels of 0 to +5V by setting the *Protocol Select Jumpers* to TTL. As an added feature it is also possible to apply power through pin 9 of the *DB-9 Connector* in order to reduce cable clutter. However, in order to accomplish this you must set the *Power Through DB-9 Jumper*.

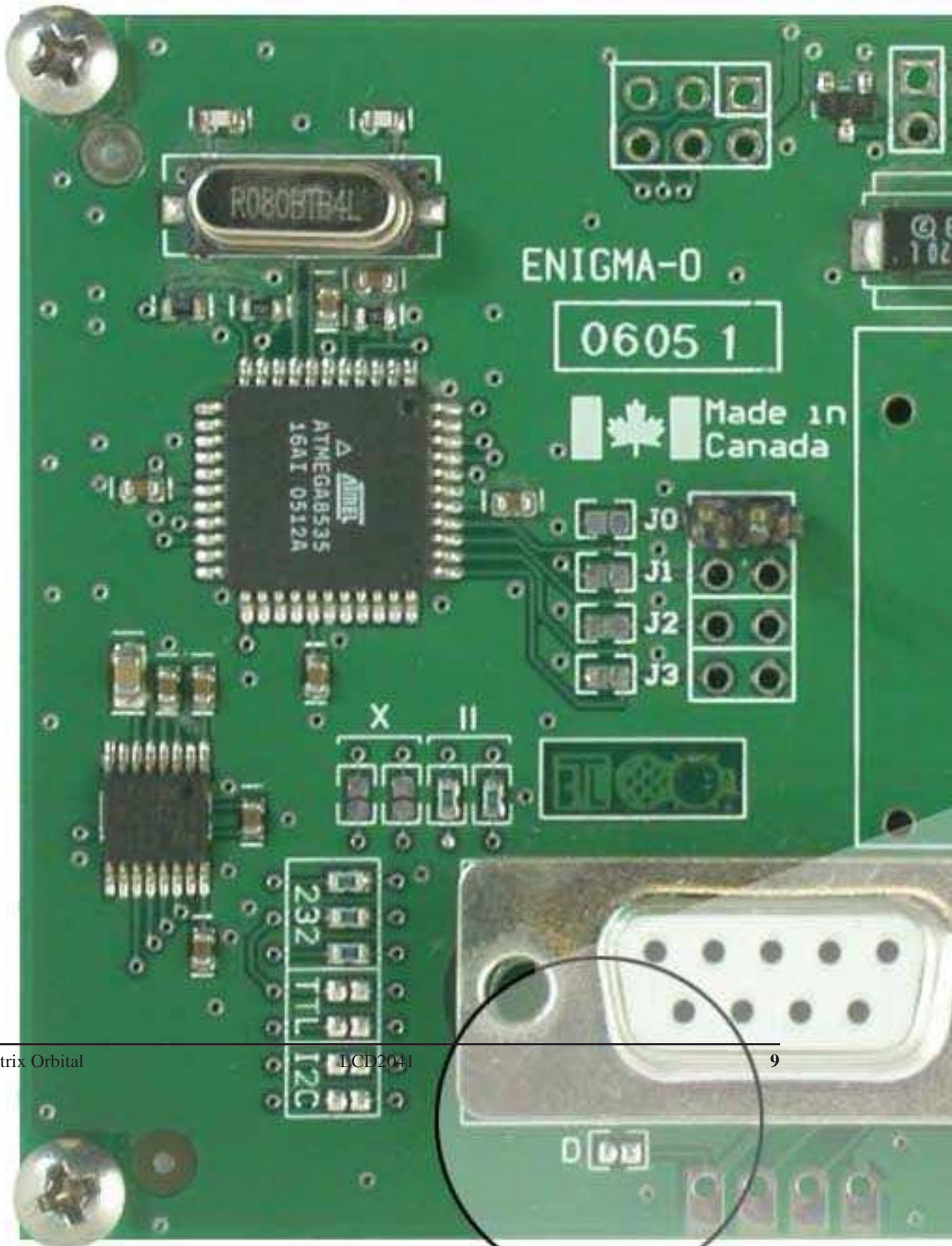


- Pin 2** Tx \ SDA (I²C data)
- Pin 3** Rx \ SCL (I²C clock)
- Pin 5** GND
- Pin 9** PWR (*Must solder Power Through DB-9 Jumper. See table 1 on page 10 for power requirements.*)

Figure 13: RS-232 Pin out

2.1.1 Power Through DB-9 Jumper

In order to provide power through pin 9 of the *DB-9 Connector* you must place a solder jumper on the *Power through DB-9 Jumper* pictured in *figure 14* below. The LCD2041 allows all voltage models to use the power through DB-9 option, see table 1 on page 10 for display module voltage requirements.

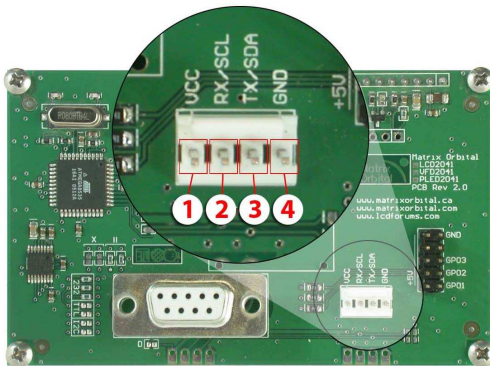




WARNING Do not apply voltage through pin 9 of the DB-9 connector AND through the Power/Data Connector at the same time.

2.2 Power/Data Connector

The *Power/Data Connector* provides a standard connector for powering the display module. The LCD2041 requires five volts for the standard display module, between nine to fifteen for the wide voltage (V) and between nine to thirty-five volts for the wide voltage with efficient power supply module (VPT). The voltage is applied through pins one and four of the four pin *Power/Data connector*. Pins two and three are reserved for serial transmission, using either the RS-232/TTL or the I²C protocol, depending on what has been selected by the *Protocol Select Jumpers*. Pins two and three may be reversed by changing the *Legacy Connector Jumpers* in order to be compatible with previous PCB revisions.



- Pin 1** PWR (See table 1)
- Pin 2** Rx \ SCL (I²C clock)
- Pin 3** Tx \ SDA (I²C data)
- Pin 4** GND

Figure 15: Power Connector and Pin out

Table 1: Power Requirements

| | Standard | -V | -VPT |
|---------------------------------|-------------------|-------------|-------------|
| Supply Voltage | +5Vdc \pm 0.25V | +9V to +15V | +9V to +35V |
| Supply Current | 11 mA typical | | |
| Supply Backlight Current | 160 mA typical | | |

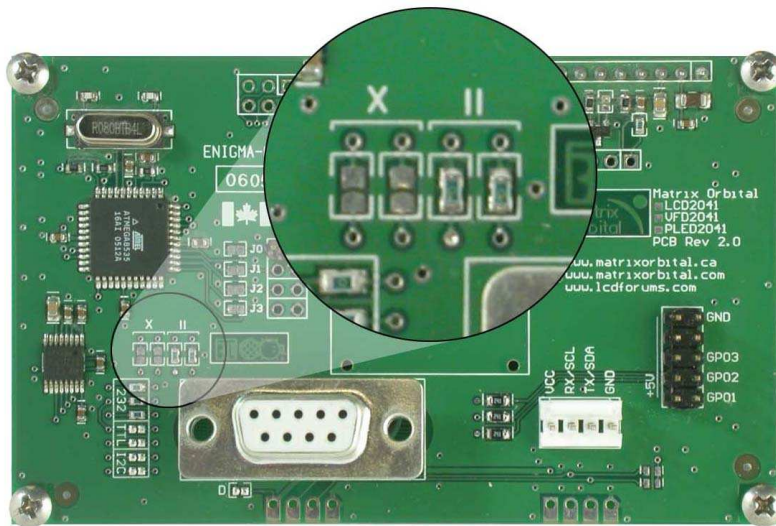


WARNINGS

- Do not apply any power with reversed polarization.
- Do not apply any voltage other than the specified voltage.

2.2.1 Legacy Data Connector Jumpers

To reverse pins two and three of the *Power/Data Connector* remove the zero ohm resistors from the *Legacy Data Connector Jumpers*, labeled with the \equiv symbol and place them on the jumpers labeled with the **X** symbol. This will allow you to transmit on pin two, and receive data on pin three instead of the default of receiving on pin two and transmitting on pin three of the *Power/Data Connector*.

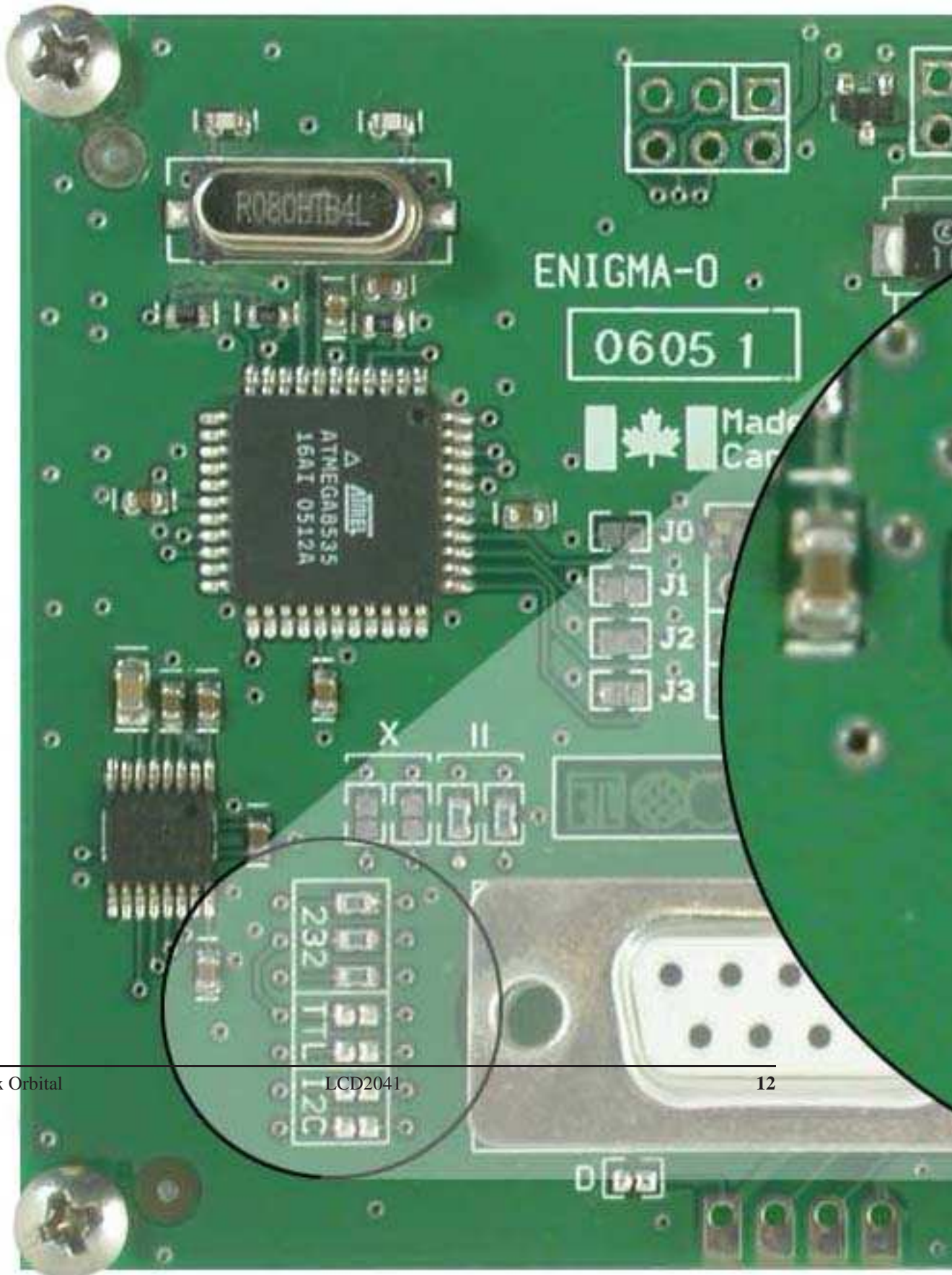


| Power/Data Connector | |
|----------------------|-----------------------------------|
| Pin 2 | Rx \ SCL (I ² C clock) |
| Pin 3 | Tx \ SDA (I ² C data) |
| Pin 2 | Tx \ SDA (I ² C data) |
| Pin 3 | Rx \ SCL (I ² C clock) |

Figure 16: Legacy Data Connector Jumpers

2.3 Protocol Select Jumpers

The *Protocol Select Jumpers*, pictured below in *figure 17*, provide the means necessary to toggle the display module between RS-232, TTL and I²C protocols. As a default, the jumpers are set to RS-232 mode with zero ohm resistors on the 232 jumpers. In order to place the display module in I²C mode you must first remove the zero ohm resistors from the 232 jumpers and then solder the resistors on to the I²C jumpers. The display will now be in I²C mode and have a default slave address of 0x50 unless it has been changed. Similarly, in order to change the display to TTL mode, simply remove the zero ohm resistors from the 232 or I²C jumpers and solder them to the TTL jumpers.



2.4 General Purpose Outputs

A unique feature of the LCD2041 is the ability to control relays and other external devices using a *General Purpose Output*, which can provide up to 20 mA of current and +5Vdc from the positive side of the GPO. If the device, which is being driven by a GPO, requires a relatively high current (such as a relay) and has an internal resistance of its own greater than 250 ohms, then the 240 ohm resistor, which is associated with the GPO may be removed and replaced with a Jumper.

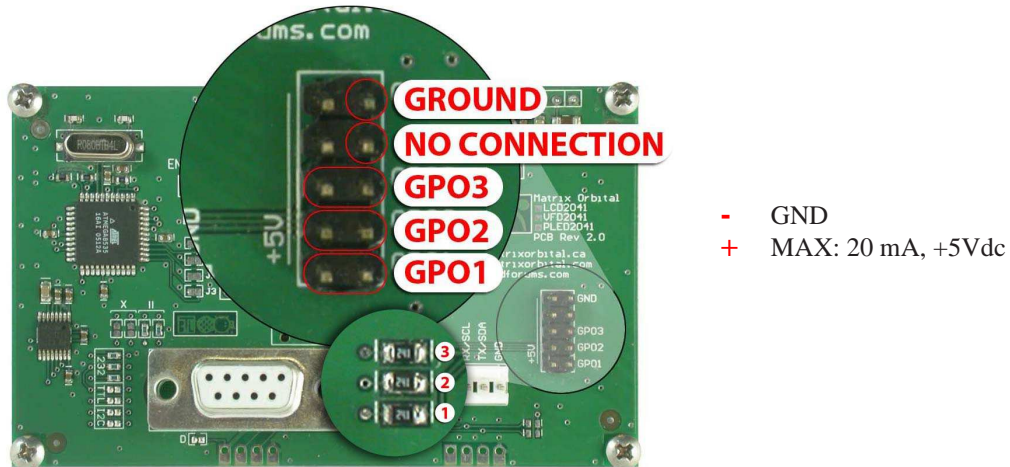


Figure 18: General Purpose Output



WARNING If connecting a relay, be sure that it is fully clamped using a diode and capacitor in order to absorb any electro-motive force (EMF) which will be generated.

2.5 Manual Override

The *Manual Override* is provided to allow the LCD2041 to be reset to factory defaults. This can be particularly helpful if the display module has been set to an unknown baud rate or I²C Slave Address and you are no longer able to communicate with it. If you wish to return the module to its default settings you must:

1. Power off the display module.
2. Place a Jumper on the *Manual Override* pins.
3. Power up the display module.
4. The display module is now set to its default values listed below in *table 2*.
5. Edit and save settings.

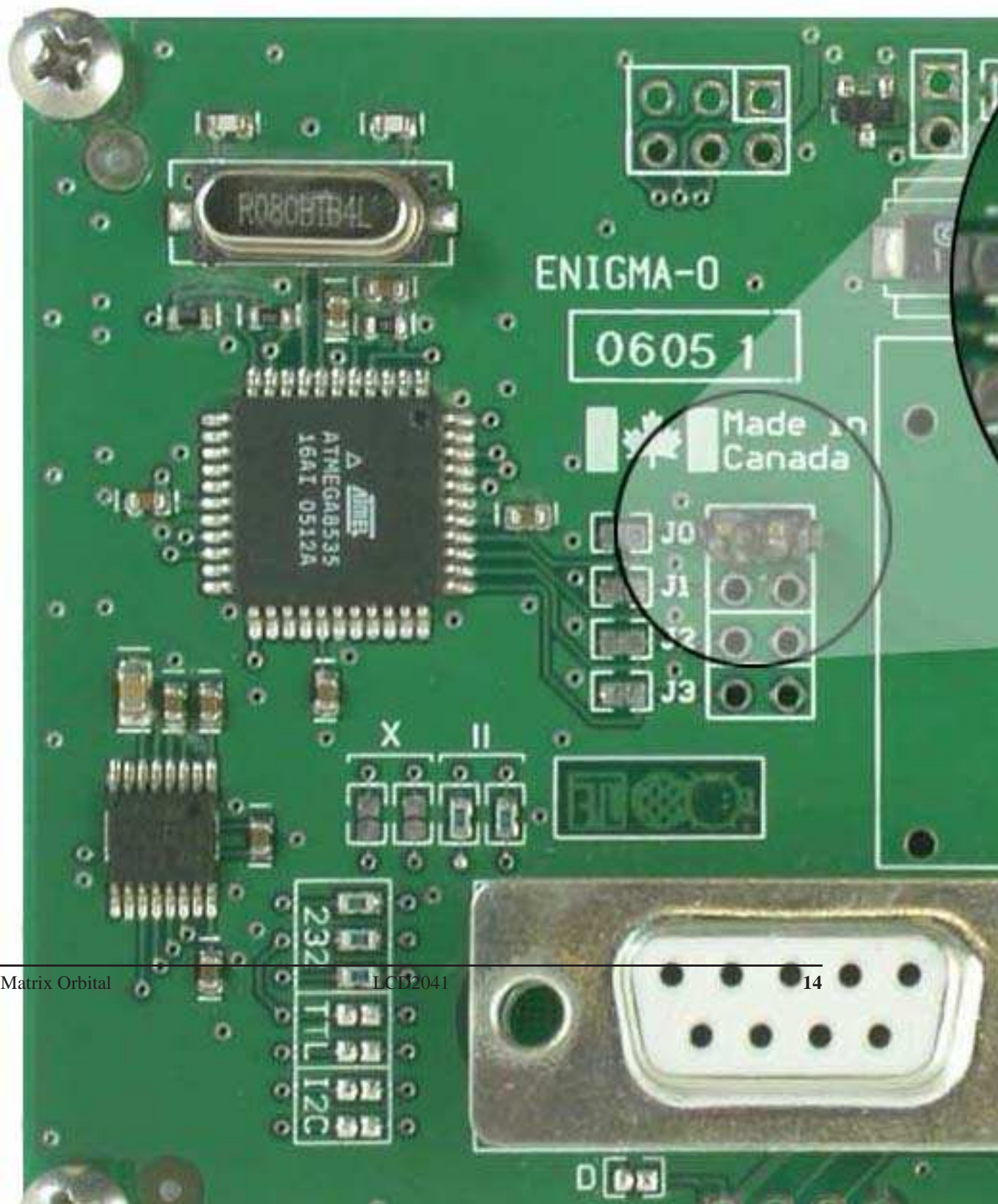


Table 2: Default Values

| | |
|-------------------------------------|-----------|
| Contrast | 128 |
| Backlight | 255 |
| Baud Rate | 19.2 kbps |
| I²C Slave Address | 0x50 |
| Data Lock | False |
| RS232AutoTransmitData | True |

NOTE The display module will revert back to the old settings once turned off, unless the settings are saved.

3 Troubleshooting

3.1 The display does not turn on when power is applied.

- First, you will want to make sure that you are using the correct power connector. Standard floppy drive power cables from your PC power supply may fit on the Power/Data Connector however they do not have the correct pin out as can be seen in *figure 10 on page 5*. Matrix Orbital supplies power cable adapters for connecting to a PC, which can be found in the *Accessories Section on page 2*.
- The next step is to check the power cable which you are using for continuity. If you don't have an ohm meter, try using a different power cable, if this does not help try using a different power supply.
- The last step will be to check the *Power / Data Connector* on the LCD2041. If the *Power / Data Connector* has become loose, or you are unable to resolve the issue, please contact Matrix Orbital, see *12.6 on page 51* for contact information.

3.2 The display module is not communicating.

- First, check the communication cable for continuity. If you don't have an ohm meter, try using a different communication cable. If you are using a PC try using a different Com Port.
- Second, please ensure that the display module is set to communicate on the protocol that you are using, by checking the *Protocol Select Jumpers*. To change the protocol used by the display module see *Section 2.3 on page 11*.
- Third, ensure that the host system and display module are both communicating on the same baud rate. The default baud rate for the display module is 19200 bps.
- If you are communicating to the display via I²C please ensure that the data is being sent to the correct address. The default slave address for the display module is 0x50.

NOTE I²C communication will always require pull up resistors.

- Finally, you may reset the display to it's default settings using the *Manual Override Jumper*, see *Section 2.5 on page 13*.

3.3 The display module is communicating, however text cannot be displayed.

- A common cause may be that the contrast settings have been set to low. The solution to this problem is to adjust the contrast settings. The default setting that will work in most environments is 128.

NOTE Optimal contrast settings may vary according to factors such as temperature, viewing angle and lighting conditions.
If you are unable to resolve any issue please contact Matrix Orbital. See *12.6 on page 51* for contact information.

4 Communications

4.1 Introduction

The commands listed in this chapter describe how to configure data flow on the LCD2041.

4.1.1 I²C Communication Summary

The LCD2041 is capable of communicating at 100 KHz in I²C mode, with 127 units addressable on a single I²C communication line. However, in order to communicate via I²C you must first ensure that pull up resistors, with a nominal value of 1K to 10K, are placed on the SCL and SDA communication lines coming from pins two and three of the Data / Power Connector respectively. Data responses by the module are automatically output via RS232, in case the host will be querying the module, it is necessary for the host to inform the module that its responses are to be output via I²C. This can be done by sending command 254 / 160 / 0 to turn off auto transmission of data in RS232. This will keep the data in the buffer until the master clocks a read of the slave. The I²C data lines operate at 5V normally or 3.3V for -1U style units. The LCD2041 uses 8-bit addressing, with the 8th or Least Significant Bit (LSB) bit designated as the read/write bit, a 0 designates a write address and a 1 designates a read address. The default read address of the display module will be 0x51, whereas the write address is 0x50 by default. This address may be changed by using cmd 254 / 51 / <address>. The LCD2041 should only be sent addresses that are even (LSB is 0). When the I²C master wishes to write to the display, the effective address is \$50 (0101 0000) , since the LSB has to be 0 for an I²C master write. When the I²C master wishes to read the LCD2041, the effective address is \$51 (0101 0001), since the LSB has to be 1 for an I²C master read.

If we take a standard Phillips 7 bit address of \$45 (100 0101), Matrix Orbital's LCD2041 would describe this Phillips I²C address as \$8A (1000 1010). The read address would be \$8B (1000 1011).

The unit does not respond to general call address (\$00).

When communicating in I²C the LCD2041 will send an ACK on the 9th clock cycle when addressed. When writing to the display module, the display will respond with a ACK when the write has successfully been completed. However if the buffer has been filled, or the module is too busy processing data it will respond with a NAK. When performing a multiple byte read within one I²C transaction, each byte read from the slave should be followed by an ACK to indicate that the master still needs data, and a NAK to indicate

that the transmission is over.

The LCD2041 has some speed limitations, especially when run in I²C mode. Here are some considerations when writing I²C code:

- * to be able to read the replies of query commands (eg. cmds 54, 55) the following command must be sent (only needs to be sent once, so this can be done somewhere in init): 254 / 160 / 0 this command puts the reply data in the I²C output buffer instead of the RS232 output buffer. Please note that due to a 16 byte output buffer, query commands that reply with more than 16 bytes cannot be read (eg cmd Get FileSystem Directory)

- * 3ms delay between the read commands
- * 625us delay in between data bytes within a transaction is necessary
- * 375us between transactions is necessary

NOTE These delays are conservative, and may be decreased based on performance

4.1.2 I²C Transaction Example

The typical I²C transaction contains four parts: the start sequence, addressing, information, and stop sequence. To begin a transaction the data line, SDA, must toggle from high to low while the clock line, SCL, is high. Next, the display must be addressed using a one byte hexadecimal value, the default to write to the unit is 0x50, while read is 0x51. Then information can be sent to the unit; even when reading, a command must first be sent to let the unit know what type of information it is required to return. After each bit is sent, the display will issue an ACK or NACK as described above. Finally, when communication is complete, the transaction is ended by toggling the data line from low to high while the clock line is high. An example of the use of this algorithm to write a simple “HELLO” message can be seen in 3.

Table 3: I²C Transaction Algorithm

| | |
|--------------------|--------------------------|
| START | Toggle SDA high to low |
| Address | 0x50 |
| Information | 0x48 0x45 0x4C 0x4C 0x4F |
| STOP | Toggle SDA low to high |

4.1.3 Serial Communication

In addition to being able to communicate via I²C the LCD2041 communicates natively through the RS-232 protocol at a default baud rate of 19,200 bps and is capable of standard baud rates from 9600 to 115,200 bps. Furthermore the LCD2041 is also capable of reproducing any non-standard baud rate in between using values entered into our baud rate generation algorithm and set through command 164 (0xA4). The display module communicates at standard voltage levels of -30V to +30V or at TTL levels of 0 to +5V by setting the *Protocol Select Jumpers* to TTL.

4.2 Changing the I²C Slave Address

| | | | |
|---|---|-----------------|---|
| Syntax | Hexadecimal | 0xFE 0x33 [adr] | |
| | Decimal | 254 51 [adr] | |
| | ASCII | 254 “3” [adr] | |
| Parameters | Parameter | Length | Description |
| | adr | 1 | The new I ² C write address (0x00 - 0xFF). |
| Description | This command sets the I ² C write address of the module between 0x00 and 0xFF. The I ² C write address must be an even number and the read address is automatically set to one higher. For example if the I ² C write address is set to 0x50, then the read address is 0x51. | | |
| <hr/> | | | |
| NOTE The change in address is immediate. | | | |
| <hr/> | | | |
| Remembered | Always | | |
| Default | 0x50 | | |

4.3 Changing the Baud Rate

| | | | |
|------------|-------------|-------------------|---|
| Syntax | Hexadecimal | 0xFE 0x39 [speed] | |
| | Decimal | 254 57 [speed] | |
| | ASCII | 254 “9” [speed] | |
| Parameters | Parameter | Length | Description |
| | speed | 1 | Hex value corresponding to a baud rate. |

Description This command sets the RS-232 port to the specified [speed]. The change takes place immediately. [speed] is a single byte specifying the desired port speed. Valid speeds are shown in the table below. The display module can be manually reset to 19,200 baud in the event of an error during transmission, including transmitting a value not listed below, by setting the manual override jumper during power up. However, it should be noted that this command will be ignored until the manual override jumper is removed again.

| Hex Value | Baud Rate |
|-----------|-----------|
| 53 | 1200 |
| 29 | 2400 |
| CF | 4800 |
| 67 | 9600 |
| 33 | 19200 |
| 22 | 28800 |
| 19 | 38400 |
| 10 | 57600 |
| 8 | 115200 |

NOTE This command is not available in I²C mode.

Remembered Always
 Default 19,200 bps

4.4 Setting a Non-Standard Baud Rate

Syntax Hexadecimal 0xFE 0xA4 [speed]
 Decimal 254 164 [speed]

| Parameter | Length | Description |
|-----------|--------|---|
| speed | 2 | Inputed LSB MSB from baud rate formula (12-2047). |

Description This command sets the RS-232 port to a non-standard baud rate. The command accepts a two byte parameter that goes directly into the modules baud generator. Use the formula, $speed = \frac{CrystalSpeed}{8 \times DesiredBaud} - 1$ to calculate the [speed] for any baud rate setting. The speed can be anywhere from 12 to 2047 which corresponds to a baud range of 977 to 153,800 baud. Setting the baud rate out of this range could cause the display to stop working properly and require the Manual Override jumper to be set.

Remembered Always

Examples

Crystal Speed 16 Mhz

Desired BAUD 13,500

$$speed = \frac{crystalspeed}{8 * DesiredBaud} - 1 \quad speed = \frac{16,000,000}{8 * 13,500} - 1$$

$$speed = 148.15 - 1$$

$$speed = 147.15$$

- **LSB** = 0x93 (rounded)
- **MSB** = 0x00
- Intended Baud Rate: 13,500 baud Actual Baud Rate:
 $\frac{16,000,000}{8(147+1)} = 13,514$ Percent Difference: 0.1%

NOTES

- Results from the formula are rounded down to the nearest whole number (i.e 73.07 = 73).
- This formula becomes less accurate as baud rates increase, due to rounding.
- Place the speed result backwards into the formula to receive the actual baud rate.
($Baud = \frac{CrystalSpeed}{8(speed+1)}$)
- The actual baud rate must be within 3% of the intended baud rate for the device to communicate.

NOTES

- This command is not available in I²C mode.
-

5 Text

5.1 Introduction

The LCD2041 is an intelligent display module, designed to reduce the amount of code necessary to begin displaying data. This means that it is able to display all ASCII formatted characters and strings that are sent to it, which are defined in the current character set. The display module will begin displaying text at the top left corner of the display area, known as home, and continue to print to the display as if it was a page on a typewriter. When the text reaches the bottom right row, it is able to automatically scroll all of the lines up and continue to display text, with the auto scroll option set to on.

5.1.1 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) | 0 | CG RAM (1) | ± | | 0EP | ' | PGEá | ' | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | |
| | 1 | CG RAM (2) | ≡ | ' | 1A0a | q | 0a1 | ' | U | t | y | o | | | | | |
| | 2 | CG RAM (3) | ŕ | " | ZBR | br | eŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | |
| | 3 | CG RAM (4) | ŕ | # | 30S | c | 3a | ŕ | ' | P | ŕ | e | ŕ | | | | |
| | 4 | CG RAM (5) | ŕ | # | 4DT | d | t | a | ŕ | ' | e | ŕ | ŕ | ŕ | ŕ | ŕ | |
| | 5 | CG RAM (6) | ŕ | z | z | e | u | e | u | a | b | e | ŕ | ' | a | ŕ | |
| | 6 | CG RAM (7) | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | |
| | 7 | CG RAM (8) | ŕ | ' | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | |
| | 8 | CG RAM (1) | ŕ | ' | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | |
| | 9 | CG RAM (2) | ŕ | ' | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | |
| | A | CG RAM (3) | ŕ | ' | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | |
| | B | CG RAM (4) | ŕ | ' | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | |
| | C | CG RAM (5) | ŕ | ' | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | |
| | D | CG RAM (6) | ŕ | ' | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | |
| | E | CG RAM (7) | ŕ | ' | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | |
| | F | CG RAM (8) | ŕ | ' | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | ŕ | |

Figure 20: Character Set

5.1.2 Control Characters

In addition to a full text set, the LCD2041 display supports the following ASCII Control characters:

0x08 Backspace

0x0C Clear screen / New page

0x0D Carriage return

0x0A Line feed / New line

5.2 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 end of the last row. | |
| Remembered | Yes | |
| Default | On | |

5.3 Auto Scroll Off

| | | |
|-------------|--|-----------|
| Syntax | Hexadecimal | 0xFE 0x52 |
| | Decimal | 254 82 |
| | ASCII | 254 “R” |
| Description | When auto scrolling is disabled the text will wrap to the top left corner of the display area when the text reaches the end of last row. | |
| Remembered | Yes | |

5.4 Clear Screen

| | | |
|-------------|---|-----------|
| Syntax | Hexadecimal | 0xFE 0x58 |
| | Decimal | 254 88 |
| | ASCII | 254 “X” |
| Description | This command will immediately clear all of the contents of the display. | |
| Remembered | No | |

5.5 Changing the Startup Screen

| | | |
|-------------|--|-----------|
| Syntax | Hexadecimal | 0xFE 0x40 |
| | Decimal | 254 64 |
| | ASCII | 254 “@” |
| Description | In order to change the text that is displayed by the LCD2041 when it starts up simply send the command bytes 254 64 followed by the characters that you wish to display, starting from the top left. This command will automatically line wrap the characters that are sent to it. | |
| Remembered | Yes | |

5.6 Set Auto Line Wrap On

| | | |
|-------------|--|-----------|
| Syntax | Hexadecimal | 0xFE 0x43 |
| | Decimal | 254 67 |
| | ASCII | 254 "C" |
| Description | Enabling Auto Line Wrap will allow the cursor to automatically wrap over to the next line when the current line is full. | |

NOTE Line wraps may occur in the middle of a word.

Remembered Yes

5.7 Set Auto Line Wrap Off

| | | |
|-------------|---|-----------|
| Syntax | Hexadecimal | 0xFE 0x44 |
| | Decimal | 254 68 |
| | ASCII | 254 "D" |
| Description | Disabling Auto Line Wrap will allow you to change the line configuration. The normally sequential progression becomes an alternating pattern. Rather than moving from line 1 to 2 to 3, the display will write from line 1 to 3 to 2. For a two line display, this means that a row's worth of characters written between the first and second lines or after the second will not be displayed on the screen or wrapped. The four line models will see only an alteration in line flow. | |

Remembered Yes

5.8 Set Cursor Position

| | | | |
|------------|-------------|-----------------------|-------------|
| Syntax | Hexadecimal | 0xFE 0x47 [col] [row] | |
| | Decimal | 254 71 [col] [row] | |
| | ASCII | 254 "G" [col] [row] | |
| Parameters | Parameter | Length | Description |
| | col | 1 | Column |
| | row | 1 | Row |

Description This command will allow you to manually set the cursor position, which controls the text insertion point, by specifying the [col] and [row] of the new proposed cursor position.

NOTE If the cursor position is set past the end of a line it will wrap to the beginning of the next line.

Remembered No

5.9 Go Home

Syntax Hexadecimal 0xFE 0x48
Decimal 254 72
ASCII 254 "H"

Description This command will return the cursor to the top left corner of the display area, identified as row one, column one.

Remembered No

5.10 Move Cursor Back

Syntax Hexadecimal 0xFE 0x4C
Decimal 254 76
ASCII 254 "L"

Description This command will move the cursor back one space. If this command is sent when the cursor is at the home position the cursor will wrap to the last row / column position if line wrap is on. Sending this command will not effect the text displayed on the module, however any characters that are sent will over write the current characters that are being displayed.

Remembered No

5.11 Move Cursor Forward

Syntax Hexadecimal 0xFE 0x4D
Decimal 254 77
ASCII 254 "M"

| | |
|-------------|--|
| Description | This command will move the cursor forward one space. If this command is sent when the cursor is at the bottom right position the cursor will wrap back to the home position if line wrap is on. Sending this command will not effect the text displayed on the module, however any characters that are sent will over write the current characters that are being displayed. |
| Remembered | No |

5.12 Underline Cursor On

| | |
|-------------|---|
| Syntax | Hexadecimal 0xFE 0x4A Decimal 254 74 ASCII 254 “J” |
| Description | This command will cause the LCD2041 to display an underline cursor at the current text insertion point. |
| Remembered | Yes |

5.13 Underline Cursor Off

| | |
|-------------|--|
| Syntax | Hexadecimal 0xFE 0x4B Decimal 254 75 ASCII 254 “K” |
| Description | This command will turn the the underline cursor off. |
| Remembered | Yes |

5.14 Blinking Block Cursor On

| | |
|-------------|--|
| Syntax | Hexadecimal 0xFE 0x53 Decimal 254 83 ASCII 254 “S” |
| Description | This command will cause the LCD2041 to display a block cursor at the current text insertion point. |
| Remembered | Yes |

5.15 Blinking Block Cursor Off

| | | |
|-------------|--|-----------|
| Syntax | Hexadecimal | 0xFE 0x54 |
| | Decimal | 254 84 |
| | ASCII | 254 “T” |
| Description | This command will turn the block cursor off. | |
| Remembered | Yes | |

6 Special Characters

6.1 Introduction

The LCD2041 has the ability to create four different sets of eight custom characters and save them to internal banks of memory. Each set of eight can be recalled from memory at any time, and selected characters can be written to the display screen. Characters and sets can be created at any time, saved for later use, and displayed to the screen through the intuitive command structure described below.

6.2 Creating a Custom Character

| | | | |
|------------|-------------|--------------------------|-------------------------------|
| Syntax | Hexadecimal | 0xFE 0x4E [refID] [data] | |
| | Decimal | 254 78 [refID] [data] | |
| | ASCII | 254 “N” [refID] [data] | |
| Parameters | Parameter | Length | Description |
| | refID | 1 | Character reference ID (0-7). |
| | data | 8 | Character data. |

Description The LCD2041 allows for up to eight custom defined characters to be added onto the character set. A custom character is a five by eight pixel matrix with each row represented by a byte value. For example:

| Custom Character 'h' | | | | | Decimal | Hex |
|----------------------|----------|----------|----------|----------|---------|------|
| 1 | 0 | 0 | 0 | 0 | 16 | 0x10 |
| 1 | 0 | 0 | 0 | 0 | 16 | 0x10 |
| 1 | 0 | 0 | 0 | 0 | 16 | 0x10 |
| 1 | 0 | 0 | 0 | 0 | 16 | 0x10 |
| 1 | 0 | 1 | 1 | 0 | 22 | 0x16 |
| 1 | 1 | 0 | 0 | 1 | 25 | 0x19 |
| 1 | 0 | 0 | 0 | 1 | 17 | 0x11 |
| 1 | 0 | 0 | 0 | 1 | 17 | 0x11 |

Each bit value of one, in the table, represents an on pixel, whereas a value of zero represents a pixel that is turned off. Therefore in order to define custom character 'h' you would send the command byte prefix 254 followed by the command 78. Next, you will have to select the memory location in which you wish to save the character in. The available memory locations for this command are zero through to seven. After sending the memory location, or [refID], you may then send the eight byte custom character data in sequence from the top to the bottom.

Once you have defined a custom character you may display it by sending the display module the [refID]. For example if a custom character was saved in position one, the command to display the custom character, at the current cursor position, would be simply to send the number one to the display module without quotes.

Remembered No

6.3 Saving Custom Characters

| | | | |
|------------|-------------|------------------------------|-------------------------------|
| Syntax | Hexadecimal | 0xFE 0xC1 [Bank] [ID] [Data] | |
| | Decimal | 254 193 [Bank] [ID] [Data] | |
| Parameters | Parameter | Length | Description |
| | Bank | 1 | Memory bank to save to (0-4). |
| | ID | 1 | Character ID (0-7) |
| | Data | 8 | Character Definition |

Description New to the LCD2041 has added five non-volatile memory banks for custom character storage. This is intended to allow you to create your own custom bar graphs, medium/large numbers and startup screen. However, each memory bank may be used to store a set of any eight custom characters; with the only provision being that memory bank zero contains the characters that will be used in the startup screen. By default the memory banks will be loaded as follows:

| [Bank] | Description |
|--------|----------------------------|
| 0 | Startup screen characters. |
| 1 | Horizontal bars |
| 2 | Vertical bars |
| 3 | Medium numbers |
| 4 | Large numbers |

In order to save new custom characters into a memory bank, follow the same process as you would for creating a custom character, see Section 6.2 on page 26, only use 254 193 [Bank Number] before sending the [ID] and character [Data].

Remembered Yes

6.4 Loading Custom Characters

Syntax Hexadecimal 0xFE 0xC0 [Bank]
 Decimal 254 192 [Bank]

| Parameter | Length | Description |
|-----------|--------|-------------------------------|
| Bank | 1 | Memory bank to save to (0-4). |

Description This command is used to load the custom characters into the volatile memory so that they may be used. If custom bar graph or number characters are stored in the memory banks, this command may be used instead of initializing the bar graph / number. To use this command send the command bytes followed by the [Bank] that contains the custom character data that you want to retrieve.

Remembered No

6.5 Save Startup Screen Custom Characters

Syntax Hexadecimal 0xFE 0xC2 [refID] [data]
 Decimal 254 194 [refID] [data]

| Parameter | Length | Description |
|-----------|--------|-------------------------------|
| refID | 1 | Character reference ID (0-7). |
| data | 8 | Character data. |

Description Using this command you may create the custom characters. that will be stored in memory bank zero, which will be used in the startup screen. For more information about creating custom characters see **Section 6.2 on page 26**.

NOTES

- Changes only take place once the power has been cycled.
- This command is the same as sending CMD 254 / 193 / 0 / [ID] / [DATA]

Remembered Yes

6.6 Initialize Medium Number

Syntax Hexadecimal 0xFE 0x6D
Decimal 254 109
ASCII 254 "m"

Description This command will load the default medium number characters into the volatile memory. If you have stored your own custom medium numbers, use the 'Load Custom Characters' command to load your custom character data into the volatile memory. This command will allow you to use the 'Place Medium Numbers' command.

Remembered No

6.7 Place Medium Numbers

Syntax Hexadecimal 0xFE 0x6F [Row] [Col] [Digit]
Decimal 254 111 [Row] [Col] [Digit]
ASCII 254 "o" [Row] [Col] [Digit]

| Parameter | Length | Description |
|-----------|--------|-------------------------------|
| Row | 1 | The row number. |
| Col | 1 | The column number. |
| Digit | 1 | Medium number to place (0-9). |

Description This command will place a medium number (two columns high) at the [row] and [col] specified.

NOTE Medium Numbers must be initialized before this command is executed.

Remembered No

6.8 Initialize Large Numbers

| | | |
|-------------|--|-----------|
| Syntax | Hexadecimal | 0xFE 0x6E |
| | Decimal | 254 110 |
| | ASCII | 254 "n" |
| Description | This command will load the default large number characters into the volatile memory. If you have stored your own custom large numbers, use the 'Load Custom Characters' command instead to load your custom character data into the volatile memory. This command will allow you to use the 'Place Large Numbers' command. | |
| Remembered | No | |

6.9 Place Large Number

| | | | |
|-------------|--|-------------------------|------------------------------|
| Syntax | Hexadecimal | 0xFE 0x23 [Col] [Digit] | |
| | Decimal | 254 35 [Col] [Digit] | |
| | ASCII | 254 "#" [Col] [Digit] | |
| Parameters | Parameter | Length | Description |
| | Col | 1 | The column number. |
| | Digit | 1 | Large number to place (0-9). |
| Description | This command will place a large number (four columns high) at the [row] and [col] specified. | | |

NOTE Large Numbers must be initialized before this command is executed.

| | |
|------------|----|
| Remembered | No |
|------------|----|

6.10 Initialize Horizontal Bar

| | | |
|-------------|---|-----------|
| Syntax | Hexadecimal | 0xFE 0x68 |
| | Decimal | 254 104 |
| | ASCII | 254 "h" |
| Description | This command will load the default horizontal bar characters into the volatile memory. If you have stored your own custom horizontal bar data, use the 'Load Custom Characters' command instead to load your custom bar data into the volatile memory. This command will allow you to use the 'Place Horizontal Bar' command. | |
| Remembered | No | |

6.11 Place Horizontal Bar Graph

| | | | |
|-------------|---|--------------------------------------|---|
| Syntax | Hexadecimal | 0xFE 0x7C [Col] [Row] [Dir] [Length] | |
| | Decimal | 254 124 [Col] [Row] [Dir] [Length] | |
| | ASCII | 254 “ ” [Col] [Row] [Dir] [Length] | |
| Parameters | Parameter | Length | Description |
| | Col | 1 | The column number. |
| | Row | 1 | The row number. |
| | Dir | 1 | The direction of the bar data (0 or 1). |
| | Length | 1 | The length of the bar data. |
| Description | This command will place a bar graph at [row], [column]. A [Dir] value of zero will cause the bar to go right, and one will cause the bar to go left. The [Length] is the size in pixels of the bar graph. | | |

NOTES

- Horizontal Bars must be initialized before this command is executed.
 - Bar graphs may be one directional only.
-

Remembered No

6.12 Initialize Narrow Vertical Bar

| | | |
|-------------|--|-----------|
| Syntax | Hexadecimal | 0xFE 0x73 |
| | Decimal | 254 115 |
| | ASCII | 254 “s” |
| Description | This command will load the narrow vertical bar characters into the volatile memory. If you have stored your own custom vertical bar data, use the 'Load Custom Characters' command instead to load your custom bar data into the volatile memory. This command will allow you to use the 'Place Vertical Bar' command. | |

NOTE Narrow bars have a width of two pixels.

Remembered No

6.13 Initialize Wide Vertical Bar

| | | |
|-------------|--|-----------|
| Syntax | Hexadecimal | 0xFE 0x76 |
| | Decimal | 254 118 |
| | ASCII | 254 “v” |
| Description | This command will load the wide vertical bar characters into the volatile memory. If you have stored your own custom vertical bar data, use the 'Load Custom Characters' command instead to load your custom bar data into the volatile memory. This command will allow you to use the 'Place Vertical Bar' command. | |

NOTE Wide bars have a width of five pixels.

Remembered No

6.14 Place Vertical Bar

| | | | |
|-------------|---|-----------------------------|-----------------------------|
| Syntax | Hexadecimal | 0xFE 0x3D [Column] [Length] | |
| | Decimal | 254 61 [Column] [Length] | |
| | ASCII | 254 “=” [Column] [Length] | |
| Parameters | Parameter | Length | Description |
| | Column | 1 | The column number. |
| | Length | 1 | The length of the bar data. |
| Description | This command will place a bar graph at the specified [Column] with the specified [Length]. The [Length] is the size in pixels of the bar graph. | | |

NOTES

- A Vertical Bar style must be initialized before this command is executed.
 - Bar graphs may be one directional only.
-

Remembered No

7 General Purpose Output

7.1 Introduction

General purpose outputs allow you to connect devices, such as LEDs, to the LCD2041 and supply them with up to 20mA of current at 5V. The LCD2041 has 3 GPOs which are software controlled, with functions to turn them on/off and set the power state for the next startup.

7.2 General Purpose Output Off

| | | | |
|-------------|--|-----------------|-------------|
| Syntax | Hexadecimal | 0xFE 0x56 [Num] | |
| | Decimal | 254 86 [Num] | |
| | ASCII | 254 “V” [Num] | |
| Parameters | Parameter | Length | Description |
| | Num | 1 | GPO number. |
| Description | This command turns OFF general purpose output [num]. | | |

NOTE OFF means that the output is pulled HIGH.

Remembered Yes

7.3 General Purpose Output On

| | | | |
|-------------|---|-----------------|-------------|
| Syntax | Hexadecimal | 0xFE 0x57 [Num] | |
| | Decimal | 254 87 [Num] | |
| | ASCII | 254 “W” [Num] | |
| Parameters | Parameter | Length | Description |
| | Num | 1 | GPO number. |
| Description | This command turns ON general purpose output [num]. The standard GPO’s on the LCD2041 output 20mA of current at 5V. | | |

NOTE ON means the output is pulled LOW.

Remembered Yes

7.4 Set Startup GPO state

| | | | |
|------------|-------------|-------------------------|-------------------------------|
| Syntax | Hexadecimal | 0xFE 0xC3 [Num] [state] | |
| | Decimal | 254 195 [Num] [state] | |
| Parameters | Parameter | Length | Description |
| | Num | 1 | GPO number. |
| | state | 1 | Startup state (0: Off, 1: On) |

Description This command will set the startup state for the GPO on the next power up. A value of one will cause the GPO to be off on the next startup while a value of one will cause the GPO to be on.

NOTE This command does not affect the current state of the GPO.

Remembered Always

8 Display Functions

8.1 Introduction

The LCD2041 employs software controlled display settings, which allow for control over, clearing the screen, changing the brightness and contrast or setting timers for turning it on or off. The combination of these allow you complete software control over your display's appearance.

8.2 Display On

| | | | |
|-------------|---|-----------------|--|
| Syntax | Hexadecimal | 0xFE 0x42 [min] | |
| | Decimal | 254 66 [min] | |
| | ASCII | 254 "B" [min] | |
| Parameters | Parameter | Length | Description |
| | min | 1 | Minutes before turning the display on (0 to 90). |
| Description | This command turns the backlight on after the [minutes] timer has expired, with a ninety minute maximum timer. A time of 0 specifies that the backlight should turn on immediately and stay on. When this command is sent while the remember function is on, the timer will reset and begin after power up. | | |
| Remembered | Yes | | |
| Default | 0 | | |

8.3 Display Off

| | | |
|--------|-------------|-----------|
| Syntax | Hexadecimal | 0xFE 0x46 |
| | Decimal | 254 70 |
| | ASCII | 254 "F" |

Description This command turns the backlight off immediately. The backlight will remain off until a 'Display On' command has been received.

Remembered Yes

8.4 Set Brightness

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

| Parameter | Length | Description |
|------------|--------|--|
| brightness | 1 | Display brightness setting (0 to 255). |

Description This command sets the display [brightness]. If the remember function is on, this command acts the same as 'Set and Save Brightness'.

Remembered Yes

Default 255

8.5 Set and Save Brightness

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

| Parameter | Length | Description |
|------------|--------|-------------------------------|
| brightness | 1 | Backlight setting (0 to 255). |

Description This command sets and saves the display [brightness] as default.

Remembered Always

8.6 Set Contrast

Syntax Hexadecimal 0xFE 0x50 [contrast]
Decimal 254 80 [contrast]
ASCII 254 "P" [contrast]

| Parameter | Length | 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 and temperature 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 is the same as the Set and Save Contrast command.

NOTE This command has only 32 levels for X-Board based displays, meaning eight contrast settings will have the same single effect. Effectively, values 0 through 7, 8 through 15, and so on will result in the same setting.

Remembered Yes
 Default 128

8.7 Set and Save Contrast

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

| Parameters | Parameter | Length | 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.

NOTE This command saves the [contrast] value so that it is not lost after power down.

Remembered Yes
 Default 128

9 Data Security

9.1 Introduction

Ensuring that your LCD2041 display's exactly what you want it to can be the difference between a projects success and failure. This is why we incorporate features such as Data Lock into the LCD2041 With this new feature you now are in control over of how and when settings will be changed so there is no need to worry about the module acting exactly like you expected it to because all the settings may be locked and remembered for the next power up.

9.2 Set Remember

| | | | |
|-------------|--|--------------------|---------------------------------|
| Syntax | Hexadecimal | 0xFE 0x93 [switch] | |
| | Decimal | 254 147 [switch] | |
| Parameters | Parameter | Length | Description |
| | switch | 1 | 0: Do not remember, 1: Remember |
| Description | This command allows you to switch the remember function on and off. To use the remember function, set remember to on, then set all of the settings that you wish to save, settings that are listed as 'Remember: Yes' support being saved into the non-volatile memory. After you have set all of the commands that you wish to save, you may then cycle the power and check the display settings to ensure that all the settings have been saved. If you wish to use remember again after cycling the power, you must set it to on again. | | |

NOTES

- Writing to non-volatile memory is time consuming and slows down the operation of the display.
- Non-volatile memory has a 'write limit' and may only be changed approximately 100,000 times.

| | |
|------------|-----------------|
| Remembered | No |
| Default | Do not remember |

9.3 Data Lock

| | | | |
|-------------|-------------|-----------------------------|--------------------------|
| Syntax | Hexadecimal | 0xFE 0xCA 0xF5 0xA0 [level] | |
| | Decimal | 254 202 245 160 [level] | |
| Parameters | Parameter | Length | Description |
| | level | 1 | Sets the data lock level |
| Description | | | |

Paranoia allows you to lock the module from displaying information, as well as enables the protection of the filesystem and module settings.

Each bit corresponds corresponds to a different lock level, while sending a zero will unlock your display as the following tables explains:

| Bit | Data Lock Level | Description |
|-----|--------------------------|--|
| 0-2 | Reserved | Should be left 0 |
| 3 | Communication Speed Lock | When this bit is set (1) the Baud Rate and I ² C Slave address are locked |
| 4 | Setting Lock | When this bit is set (1) the display settings such as backlight, contrast and GPO settings are locked. (Internal EEPROM) |
| 5 | Reserved | Should be left 0 |
| 6 | Command Lock | When this bit is set (1) all commands but commands 202/203 are locked. (cmd lock) |
| 7 | Display Lock | When this bit is set (1) the module is locked from displaying any new information. (text lock) |

NOTES

- Sending a new data lock level will override the previous data lock level.
 - Data lock levels may be combined.
-

| | |
|------------|--------|
| Remembered | Always |
| Default | 0 |
| Examples | |

| Hex | Dec | Binary | Description |
|------|-----|----------|--------------------------|
| 0x00 | 0 | 0 | Unlock |
| 0x50 | 80 | 01010000 | Setting and Command Lock |

9.4 Set and Save Data Lock

| | | | |
|-------------|---|-----------------------------|--------------------------|
| Syntax | Hexadecimal | 0xFE 0xCB 0xF5 0xA0 [level] | |
| | Decimal | 254 203 245 160 [level] | |
| Parameters | Parameter | Length | Description |
| | level | 1 | Sets the data lock level |
| Description | This command will set and save the data lock level. See the Data Lock section for more information. | | |
| Remembered | Always | | |
| Default | 0 | | |

9.5 Write Customer Data

| | | | |
|-------------|---|------------------|--------------------------|
| Syntax | Hexadecimal | 0xFE 0x34 [data] | |
| | Decimal | 254 52 [data] | |
| | ASCII | 254 "4" [data] | |
| Parameters | Parameter | Length | Description |
| | data | 16 | Writes the customer data |
| Description | Writes the customer Data. 16 Bytes of data can be saved in non-volatile memory. | | |
| Remembered | No | | |

9.6 Read Customer Data

| | | |
|-------------|--|-----------|
| Syntax | Hexadecimal | 0xFE 0x35 |
| | Decimal | 254 53 |
| | ASCII | 254 "5" |
| Description | Reads whatever was written by Write Customer Data. | |
| Remembered | No | |

10 Miscellaneous

10.1 Introduction

This chapter covers the 'Report Version Number' and 'Read Module Type' commands. These commands can be particularly useful to find out more information about the display module before contacting technical support.

10.2 Read Version Number

| | | |
|-------------|--|-----------|
| Syntax | Hexadecimal | 0xFE 0x36 |
| | Decimal | 254 54 |
| | ASCII | 254 “6” |
| Description | This command will return a byte representing the version of the module, see the following table as an example: | |

| Hex Value | Version Number |
|-----------|----------------|
| 0x19 | Version 1.9 |
| 0x57 | Version 5.7 |

Remembered No

10.3 Read Module Type

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

Description

This command will return a hex value corresponding to the the model number of the module see the following table:

| Hex | Product ID | Hex | Product ID |
|-----|--------------------|-----|-----------------|
| 1 | LCD0821 | 2 | LCD2021 |
| 5 | LCD2041 | 6 | LCD4021 |
| 7 | LCD4041 | 8 | LK202-25 |
| 9 | LK204-25 | A | LK404-55 |
| B | VFD2021 | C | VFD2041 |
| D | VFD4021 | E | VK202-25 |
| F | VK204-25 | 10 | GLC12232 |
| 13 | GLC24064 | 14 | Unused |
| 15 | GLK24064-25 | 16 | Unused |
| 21 | Unused | 22 | GLK12232-25 |
| 23 | Unused | 24 | GLK12232-25-SM |
| 25 | GLK24064-16-1U-USB | 26 | GLK24064-16-1U |
| 27 | GLK19264-7T-1U-USB | 28 | GLK12232-16 |
| 29 | GLK12232-16-SM | 2A | GLK19264-7T-1U |
| 2B | LK204-7T-1U | 2C | LK204-7T-1U-USB |
| 31 | LK404-AT | 32 | MOS-AV-162A |
| 33 | LK402-12 | 34 | LK162-12 |
| 35 | LK204-25PC | 36 | LK202-24-USB |
| 37 | VK202-24-USB | 38 | LK204-24-USB |
| 39 | VK204-24-USB | 3A | PK162-12 |
| 3B | VK162-12 | 3C | MOS-AP-162A |
| 3D | PK202-25 | 3E | MOS-AL-162A |
| 3F | MOS-AL-202A | 40 | MOS-AV-202A |
| 41 | MOS-AP-202A | 42 | PK202-24-USB |
| 43 | MOS-AL-082 | 44 | MOS-AL-204 |
| 45 | MOS-AV-204 | 46 | MOS-AL-402 |
| 47 | MOS-AV-402 | 48 | LK082-12 |
| 49 | VK402-12 | 4A | VK404-55 |
| 4B | LK402-25 | 4C | VK402-25 |
| 4D | PK204-25 | 4E | Unused |
| 4F | MOS | 50 | MOI |
| 51 | XBoard-S | 52 | XBoard-I |
| 53 | MOU | 54 | XBoard-U |
| 55 | LK202-25-USB | 56 | VK202-25-USB |
| 57 | LK204-25-USB | 58 | VK204-25-USB |
| 5B | LK162-12-TC | 5C | Unused |
| 71 | Unused | 72 | GLK240128-25 |
| 73 | LK404-25 | 74 | VK404-25 |
| 77 | Unused | 78 | GLT320240 |
| 79 | GLT480282 | 7A | GLT240128 |

Remembered

No

11 Command Summary

11.1 Communications

| Description | Syntax | Page | |
|---|-------------|-------------------|----|
| Changing the I ² C Slave Address | Hexadecimal | 0xFE 0x33 [adr] | 18 |
| | Decimal | 254 51 [adr] | |
| | ASCII | 254 “3” [adr] | |
| Changing the Baud Rate | Hexadecimal | 0xFE 0x39 [speed] | 18 |
| | Decimal | 254 57 [speed] | |
| | ASCII | 254 “9” [speed] | |
| Setting a Non-Standard Baud Rate | Hexadecimal | 0xFE 0xA4 [speed] | 19 |
| | Decimal | 254 164 [speed] | |

11.2 Text

| Description | Syntax | Page | |
|-----------------------------|-------------|-----------------------|----|
| Auto Scroll On | Hexadecimal | 0xFE 0x51 | 21 |
| | Decimal | 254 81 | |
| | ASCII | 254 “Q” | |
| Auto Scroll Off | Hexadecimal | 0xFE 0x52 | 22 |
| | Decimal | 254 82 | |
| | ASCII | 254 “R” | |
| Clear Screen | Hexadecimal | 0xFE 0x58 | 22 |
| | Decimal | 254 88 | |
| | ASCII | 254 “X” | |
| Changing the Startup Screen | Hexadecimal | 0xFE 0x40 | 22 |
| | Decimal | 254 64 | |
| | ASCII | 254 “@” | |
| Set Auto Line Wrap On | Hexadecimal | 0xFE 0x43 | 23 |
| | Decimal | 254 67 | |
| | ASCII | 254 “C” | |
| Set Auto Line Wrap Off | Hexadecimal | 0xFE 0x44 | 23 |
| | Decimal | 254 68 | |
| | ASCII | 254 “D” | |
| Set Cursor Position | Hexadecimal | 0xFE 0x47 [col] [row] | 23 |
| | Decimal | 254 71 [col] [row] | |
| | ASCII | 254 “G” [col] [row] | |
| Go Home | Hexadecimal | 0xFE 0x48 | 24 |
| | Decimal | 254 72 | |
| | ASCII | 254 “H” | |

| Description | Syntax | Page |
|---------------------------|-------------|-----------|
| Move Cursor Back | Hexadecimal | 0xFE 0x4C |
| | Decimal | 254 76 |
| | ASCII | 254 “L” |
| Move Cursor Forward | Hexadecimal | 0xFE 0x4D |
| | Decimal | 254 77 |
| | ASCII | 254 “M” |
| Underline Cursor On | Hexadecimal | 0xFE 0x4A |
| | Decimal | 254 74 |
| | ASCII | 254 “J” |
| Underline Cursor Off | Hexadecimal | 0xFE 0x4B |
| | Decimal | 254 75 |
| | ASCII | 254 “K” |
| Blinking Block Cursor On | Hexadecimal | 0xFE 0x53 |
| | Decimal | 254 83 |
| | ASCII | 254 “S” |
| Blinking Block Cursor Off | Hexadecimal | 0xFE 0x54 |
| | Decimal | 254 84 |
| | ASCII | 254 “T” |

11.3 Special Characters

| Description | Syntax | Page |
|---------------------------------------|-------------|-------------------------------|
| Creating a Custom Character | Hexadecimal | 0xFE 0x4E [refID] [data] |
| | Decimal | 254 78 [refID] [data] |
| | ASCII | 254 “N” [refID] [data] |
| Saving Custom Characters | Hexadecimal | 0xFE 0xC1 [Bank] [ID] [Data] |
| | Decimal | 254 193 [Bank] [ID] [Data] |
| Loading Custom Characters | Hexadecimal | 0xFE 0xC0 [Bank] |
| | Decimal | 254 192 [Bank] |
| Save Startup Screen Custom Characters | Hexadecimal | 0xFE 0xC2 [refID] [data] |
| | Decimal | 254 194 [refID] [data] |
| Initialize Medium Number | Hexadecimal | 0xFE 0x6D |
| | Decimal | 254 109 |
| | ASCII | 254 “m” |
| Place Medium Numbers | Hexadecimal | 0xFE 0x6F [Row] [Col] [Digit] |
| | Decimal | 254 111 [Row] [Col] [Digit] |
| | ASCII | 254 “o” [Row] [Col] [Digit] |
| Initialize Large Numbers | Hexadecimal | 0xFE 0x6E |
| | Decimal | 254 110 |
| | ASCII | 254 “n” |
| Place Large Number | Hexadecimal | 0xFE 0x23 [Col] [Digit] |
| | Decimal | 254 35 [Col] [Digit] |
| | ASCII | 254 “#” [Col] [Digit] |

| Description | Syntax | Page |
|--------------------------------|-------------|--------------------------------------|
| Initialize Horizontal Bar | Hexadecimal | 0xFE 0x68 |
| | Decimal | 254 104 |
| | ASCII | 254 "h" |
| Place Horizontal Bar Graph | Hexadecimal | 0xFE 0x7C [Col] [Row] [Dir] [Length] |
| | Decimal | 254 124 [Col] [Row] [Dir] [Length] |
| | ASCII | 254 " " [Col] [Row] [Dir] [Length] |
| Initialize Narrow Vertical Bar | Hexadecimal | 0xFE 0x73 |
| | Decimal | 254 115 |
| | ASCII | 254 "s" |
| Initialize Wide Vertical Bar | Hexadecimal | 0xFE 0x76 |
| | Decimal | 254 118 |
| | ASCII | 254 "v" |
| Place Vertical Bar | Hexadecimal | 0xFE 0x3D [Column] [Length] |
| | Decimal | 254 61 [Column] [Length] |
| | ASCII | 254 "=" [Column] [Length] |

11.4 General Purpose Output

| Description | Syntax | Page |
|----------------------------|-------------|-------------------------|
| General Purpose Output Off | Hexadecimal | 0xFE 0x56 [Num] |
| | Decimal | 254 86 [Num] |
| | ASCII | 254 "V" [Num] |
| General Purpose Output On | Hexadecimal | 0xFE 0x57 [Num] |
| | Decimal | 254 87 [Num] |
| | ASCII | 254 "W" [Num] |
| Set Startup GPO state | Hexadecimal | 0xFE 0xC3 [Num] [state] |
| | Decimal | 254 195 [Num] [state] |

11.5 Display Functions

| Description | Syntax | Page |
|-------------------------|-------------|------------------------|
| Display On | Hexadecimal | 0xFE 0x42 [min] |
| | Decimal | 254 66 [min] |
| | ASCII | 254 "B" [min] |
| Display Off | Hexadecimal | 0xFE 0x46 |
| | Decimal | 254 70 |
| | ASCII | 254 "F" |
| Set Brightness | Hexadecimal | 0xFE 0x99 [brightness] |
| | Decimal | 254 153 [brightness] |
| Set and Save Brightness | Hexadecimal | 0xFE 0x98 [brightness] |
| | Decimal | 254 152 [brightness] |

| Description | Syntax | | Page |
|-----------------------|-------------|----------------------|------|
| Set Contrast | Hexadecimal | 0xFE 0x50 [contrast] | 35 |
| | Decimal | 254 80 [contrast] | |
| | ASCII | 254 “P” [contrast] | |
| Set and Save Contrast | Hexadecimal | 0xFE 0x91 [contrast] | 36 |
| | Decimal | 254 145 [contrast] | |

11.6 Data Security

| Description | Syntax | | Page |
|------------------------|-------------|-----------------------------|------|
| Set Remember | Hexadecimal | 0xFE 0x93 [switch] | 37 |
| | Decimal | 254 147 [switch] | |
| Data Lock | Hexadecimal | 0xFE 0xCA 0xF5 0xA0 [level] | 38 |
| | Decimal | 254 202 245 160 [level] | |
| Set and Save Data Lock | Hexadecimal | 0xFE 0xCB 0xF5 0xA0 [level] | 39 |
| | Decimal | 254 203 245 160 [level] | |
| Write Customer Data | Hexadecimal | 0xFE 0x34 [data] | 39 |
| | Decimal | 254 52 [data] | |
| Read Customer Data | ASCII | 254 “4” [data] | 39 |
| | Hexadecimal | 0xFE 0x35 | |
| | Decimal | 254 53 | |
| | ASCII | 254 “5” | |

11.7 Miscellaneous

| Description | Syntax | | Page |
|---------------------|-------------|-----------|------|
| Read Version Number | Hexadecimal | 0xFE 0x36 | 40 |
| | Decimal | 254 54 | |
| | ASCII | 254 “6” | |
| Read Module Type | Hexadecimal | 0xFE 0x37 | 40 |
| | Decimal | 254 55 | |
| | ASCII | 254 “7” | |

11.8 Command By Number

| Command | Description | | Page |
|---------|-------------|-------|--|
| Hex | Dec | ASCII | |
| 0x23 | 35 | “#” | Place Large Number 30 |
| 0x33 | 51 | “3” | Changing the I ² C Slave Address 18 |
| 0x34 | 52 | “4” | Write Customer Data 39 |
| 0x35 | 53 | “5” | Read Customer Data 39 |
| 0x36 | 54 | “6” | Read Version Number 40 |

| Command | Description | Page | | |
|---------|-------------|-------|---------------------------------------|----|
| Hex | Dec | ASCII | | |
| 0x37 | 55 | “7” | Read Module Type | 40 |
| 0x39 | 57 | “9” | Changing the Baud Rate | 18 |
| 0x3D | 61 | “=” | Place Vertical Bar | 32 |
| 0x40 | 64 | “@” | Changing the Startup Screen | 22 |
| 0x42 | 66 | “B” | Display On | 34 |
| 0x43 | 67 | “C” | Set Auto Line Wrap On | 23 |
| 0x44 | 68 | “D” | Set Auto Line Wrap Off | 23 |
| 0x46 | 70 | “F” | Display Off | 34 |
| 0x47 | 71 | “G” | Set Cursor Position | 23 |
| 0x48 | 72 | “H” | Go Home | 24 |
| 0x4A | 74 | “J” | Underline Cursor On | 25 |
| 0x4B | 75 | “K” | Underline Cursor Off | 25 |
| 0x4C | 76 | “L” | Move Cursor Back | 24 |
| 0x4D | 77 | “M” | Move Cursor Forward | 24 |
| 0x4E | 78 | “N” | Creating a Custom Character | 26 |
| 0x50 | 80 | “P” | Set Contrast | 35 |
| 0x51 | 81 | “Q” | Auto Scroll On | 21 |
| 0x52 | 82 | “R” | Auto Scroll Off | 22 |
| 0x53 | 83 | “S” | Blinking Block Cursor On | 25 |
| 0x54 | 84 | “T” | Blinking Block Cursor Off | 25 |
| 0x56 | 86 | “V” | General Purpose Output Off | 33 |
| 0x57 | 87 | “W” | General Purpose Output On | 33 |
| 0x58 | 88 | “X” | Clear Screen | 22 |
| 0x68 | 104 | “h” | Initialize Horizontal Bar | 30 |
| 0x6D | 109 | “m” | Initialize Medium Number | 29 |
| 0x6E | 110 | “n” | Initialize Large Numbers | 30 |
| 0x6F | 111 | “o” | Place Medium Numbers | 29 |
| 0x73 | 115 | “s” | Initialize Narrow Vertical Bar | 31 |
| 0x76 | 118 | “v” | Initialize Wide Vertical Bar | 31 |
| 0x7C | 124 | “ ” | Place Horizontal Bar Graph | 31 |
| 0x91 | 145 | | Set and Save Contrast | 36 |
| 0x93 | 147 | | Set Remember | 37 |
| 0x98 | 152 | | Set and Save Brightness | 35 |
| 0x99 | 153 | | Set Brightness | 35 |
| 0xA4 | 164 | | Setting a Non-Standard Baud Rate | 19 |
| 0xC0 | 192 | | Loading Custom Characters | 28 |
| 0xC1 | 193 | | Saving Custom Characters | 27 |
| 0xC2 | 194 | | Save Startup Screen Custom Characters | 28 |
| 0xC3 | 195 | | Set Startup GPO state | 33 |

12 Appendix

12.1 Specifications

12.1.1 Environmental

Table 58: Environmental Specifications

| | Standard Temperature | Extended Temperature |
|------------------------------------|--------------------------------------|-----------------------------|
| Operating Temperature | 0°C to +50°C | -20°C to +70°C |
| Storage Temperature | -20°C to +70°C | -30°C to +80°C |
| Operating Relative Humidity | 90% max non-condensing | |
| Vibration (Operating) | 4.9 m/s ² XYZ directions | |
| Vibration (Non-Operating) | 19.6 m/s ² XYZ directions | |
| Shock (Operating) | 29.4 m/s ² XYZ directions | |
| Shock (Non-Operating) | 490 m/s ² XYZ directions | |

12.1.2 Electrical

Table 59: Electrical Specifications

| | Standard | -V | -VPT |
|-----------------------------------|---------------------------|-------------|-------------|
| Supply Voltage | +5Vdc ±0.25V | +9V to +15V | +9V to +35V |
| Minimum Current | 40mA typical | | |
| Backlight On (YG & IY) | add 130mA (170mA) typical | | |
| Backlight On (R) | add 160mA (200mA) typical | | |
| Backlight On (GW & WB) | add 35mA (75mA) typical | | |

12.2 Optical Characteristics

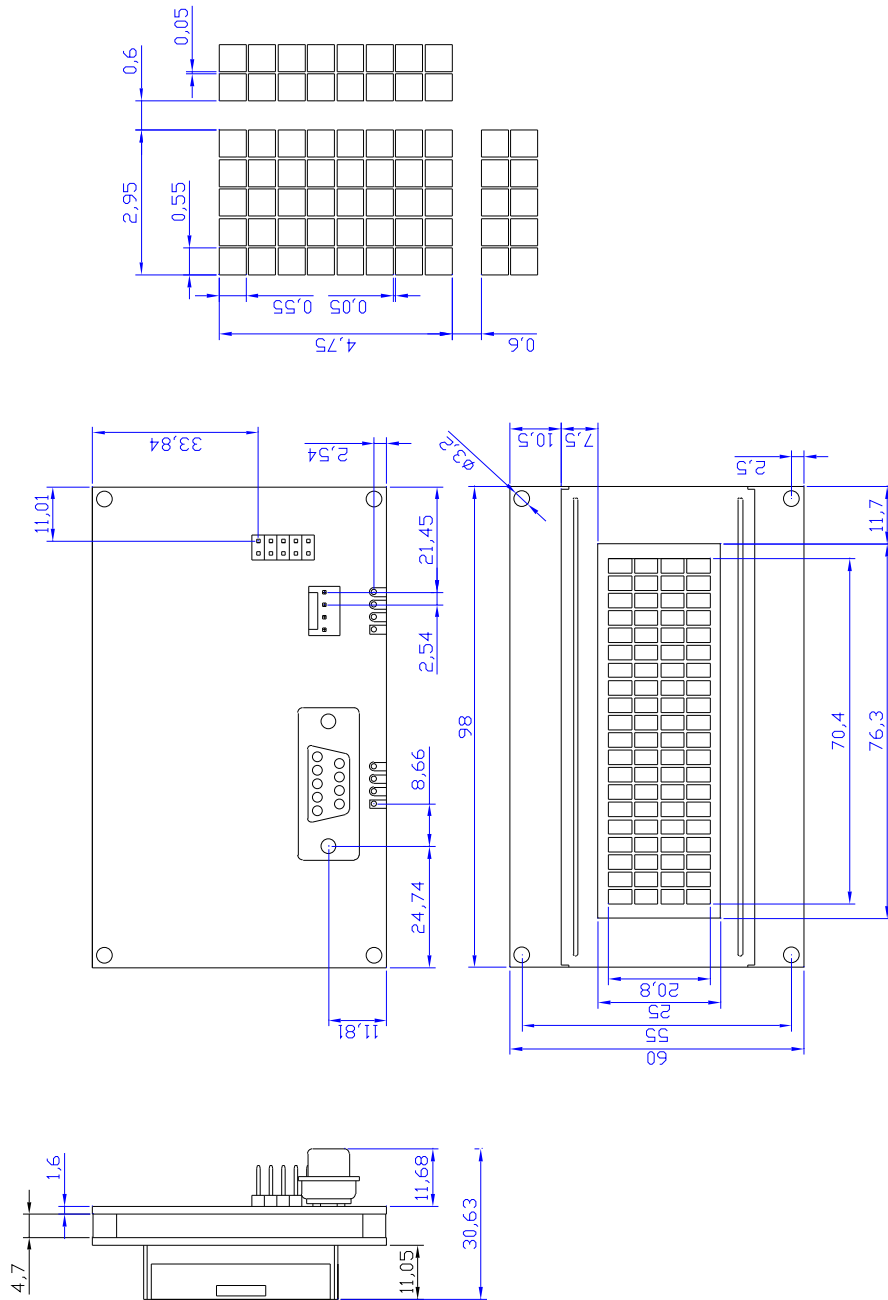
Table 60: Optical Characteristics

| | |
|--|--------------------------------|
| Character x Lines | 20 columns x 4 rows |
| Module Size | 98.00 mm x 60.00 mm x 27.50 mm |
| Character Size | 2.95 mm x 4.75 mm |
| Display Size | 76.30 mm x 25.00 mm |
| LED Backlight Half-Life (YG & IY) | 50,000 hours typical |
| LED Backlight Half-Life (R) | 20,000 hours typical |
| LED Backlight Half-Life (GW & WB) | 10,000 hours typical |

NOTE To prolong life, it is recommended that the backlight be turned off when the display is not in use.

12.3 Physical Layout

Figure 21: Physical Diagram



12.4 Ordering Information

| | | | | | | |
|-----|----|---|---|-----|------|----|
| LCD | 20 | 4 | 1 | -IY | -VPT | -E |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |

Table 61: Part Numbering Scheme

| # | Description | Options |
|---|-------------------------|--|
| 1 | Screen Type | LCD: Liquid Crystal Display* |
| 2 | Width | 20: Twenty Character Columns |
| 3 | Height | 4: Four Character Rows |
| 4 | GPO | 1: Single GPO |
| 5 | Color (Text/Background) | NP: Standard Grey/Yellow-Green GW: Grey/White WB: White/Blue R: Red/Black IY: Yellow/Black |
| 6 | Input Voltage | NP: Standard (4.75-5.25V) -V: Extended Voltage (9.00-15.0V) -VPT: Wide voltage with ESPS (9.00-35.00V) |
| 7 | Temperature | NP: Standard (0°C to +50°C) -E: Extended Temperature (-20°C to +70°C) |

Table 62: Part Options

NOTE * Also available with a Vacuum Florescent Display in the VFD2041 model

12.5 Definitions

E Extended Temperature (-20C to 70C)

VPT Wide Voltage with Efficient Switching Power Supply (+9 to +35Vdc)

V Wide Voltage (+9 to +15Vdc)

GW Grey Text / White Background

WB White Text / Blue Background

R Inverse Red

IY Inverse Yellow

MSB Most Significant Byte

LSB Least Significant Byte

12.6 Contacting Matrix Orbital

Telephone

Sales: 1(403)229-2737

Support: 1(403)204-3750

On The Web

Sales: <http://www.MatrixOrbital.com>

Support: <http://www.MatrixOrbital.ca>

Forums: <http://www.lcdforums.com>

12.7 Revision History

Table 63: Revision History

| Revision | Description | Author |
|-----------------|------------------------|----------------|
| 2.0 | Initial Manual | Matrix Orbital |
| 2.1 | Updated Backlight Life | Clark |

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for [LCD Touch Panels](#) category:

Click to view products by [Matrix Orbital](#) manufacturer:

Other Similar products are found below :

[98-0003-2669-8](#) [CABLE/RTP/USB](#) [98-0003-2933-8](#) [17-8041-226](#) [17-8041-227](#) [CABLE/RTP/RS232](#) [N010-0557-T002](#) [N010-0557-T011](#)
[NA5-15W101B-V1](#) [LCDAL04-0430TL0NCNTBR1.0](#) [TP01106W-4KB](#) [FTAS00-15AN-4](#) [FTAS00-12.1AW-4A](#) [17-8021-227](#) [17-8031-205](#)
[17-8031-227](#) [17-8091-203](#) [17-8421-205](#) [17-8421-227](#) [17-8441-205](#) [17-8511-205](#) [17-8541-205](#) [17-8541-226](#) [17-8891-227](#) [17-8901-226](#) [98-](#)
[0003-2185-5](#) [98-0003-2617-7](#) [98-0003-2919-7](#) [98-0003-2955-1](#) [98-0003-3140-9](#) [98-0003-3190-4](#) [98-0003-3505-3](#) [98-1100-0606-5](#) [98-1100-](#)
[0638-8](#) [EA TOUCH240-4](#) [N010-0519-T742](#) [N010-0554-T043](#) [N010-0554-T703](#) [EA TOUCH102-1](#) [EA TOUCH128-1](#) [EA TOUCH128-2](#)
[EA TOUCH160-1](#) [LCD-OLINUXINO-10CTS](#) [TS-320240BRNO](#) [TS-TFT3.5Z](#) [FTAS00-104A5](#) [FTAS00-104AS4](#) [FTAS00-121A4](#) [FTAS00-](#)
[121A5](#) [FTAS00-121AS4](#)