



LCD0821

Including LCD0821

Technical Manual

Revision 3.2

PCB Revision: 2.0 or Higher

Firmware Revision: 5.0 or Higher

Revision History

| Revision | Date | Description | Author |
|----------|-----------------|---|---------|
| 3.2 | January 4, 2018 | Correction to Set Non-Standard Baud Rate command | Divino |
| 3.1 | March 12, 2014 | Revision and correction to Colour in Ordering Options | Martino |
| 3.0 | May 7, 2013 | Initial Release | Clark |

Contents

| | |
|------------------------------------|----|
| 1 Introduction | 1 |
| 2 Quick Connect Guide..... | 2 |
| 2.1 Available Headers | 2 |
| 2.2 Standard Module | 3 |
| Recommended Parts..... | 3 |
| Serial Connections..... | 3 |
| I ² C Connections | 4 |
| 3 Software | 5 |
| 3.1 uProject..... | 5 |
| 3.2 Application Notes..... | 6 |
| 4 Hardware..... | 7 |
| 4.1 Standard Model | 7 |
| Communication/Power Header | 7 |
| Alternate Serial Header..... | 7 |
| Protocol Select Jumpers..... | 7 |
| 4.2 Common Features..... | 8 |
| General Purpose Outputs | 8 |
| 5 Troubleshooting..... | 9 |
| 5.1 Power | 9 |
| 5.2 Display..... | 9 |
| 5.3 Communication..... | 10 |
| 5.4 Manual Override | 10 |
| 6 Commands | 11 |
| 6.1 Communication..... | 11 |
| 6.2 Text..... | 12 |
| 6.3 Special Characters..... | 14 |
| 6.4 General Purpose Output | 16 |
| 6.5 Display Functions | 17 |
| 6.6 Data Security..... | 18 |
| 6.7 Miscellaneous | 19 |

| | |
|---------------------------------------|----|
| 7 Appendix | 20 |
| 7.1 Command Summary | 20 |
| 7.2 Character Sets | 22 |
| 7.3 Block Diagram | 24 |
| 7.4 Environmental Specifications..... | 24 |
| 7.5 Electrical Tolerances | 24 |
| 7.6 Dimensional Drawings | 25 |
| 7.7 Optical Characteristics | 25 |
| 8 Ordering | 26 |
| 8.1 Part Numbering Scheme | 26 |
| 8.2 Options..... | 26 |
| 8.3 Accessories..... | 26 |
| 9 Definitions..... | 27 |
| 10 Contact..... | 27 |

1 Introduction

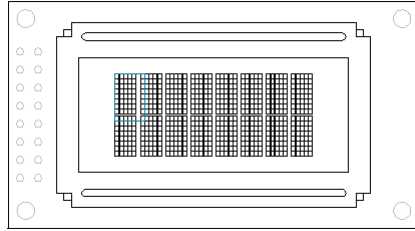


Figure 1: LCD0821 Display

The LCD0821 is an intelligent alphanumeric liquid crystal display designed to decrease development time by providing an instant solution to any project. . RS232, TTL and I2C protocols communication options allow the LCD0821 to be connected to a wide variety of host controllers. Communication speeds of up to 115.2kbps for serial protocols and 100kbps for I²C ensure lightning fast display updates.

The simple command structure permits easy software control of many settings including backlight brightness, screen contrast, and baud rate. On board memory provides up to forty custom characters which can be saved within the unit and recalled for start screens, bar graphs or larger numbers. Finally, one general purpose outputs provide simple switchable five volt sources.

The versatile LCD0821, with all the features mentioned above, is available in a variety of colour, voltage, and temperature options to suit almost any application.

2 Quick Connect Guide

2.1 Available Headers

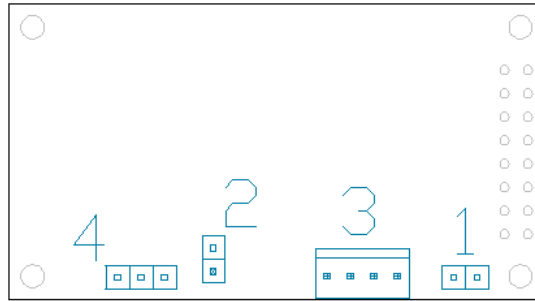


Figure 2: LCD0821 Standard Module Header Locations

Table 1: List of Available Headers

| # | Header | Mate | Population |
|---|-------------------------------|--------------|------------|
| 1 | GPO Header | None Offered | All Models |
| 2 | Manual Override | Jumper | All Models |
| 3 | Communication/Power Connector | SCCPC5V/BBC | All Models |
| 4 | Alternate Serial Header | None Offered | All Models |

2.2 Standard Module

The standard version of the LCD0821 allows for user configuration of two common communication protocols. First, the unit can communicate using serial protocol at either RS232 or TTL voltage levels. Second, it can communicate using the Inter-Integrated Circuit connect, or I²C protocol. Connections for each protocol can be accessed through the four pin Communication/Power Header as outlined in the Serial Connections and I²C Connections sections below.

Recommended Parts



Figure 3: Communication/Power Cable (SCCPC5V)

The most common cable choice for any alphanumeric Matrix Orbital Display, the Communication/ Power Cable offers a simple connection to the unit with familiar interfaces. DB9 and floppy power headers provide all necessary input to drive your display.



Figure 4: Breadboard Cable (BBC)

For a more flexible interface to the LCD0821 a Breadboard Cable may be used. This provides a simple four wire connection that is popular among developers for its ease of use in a breadboard environment.

Serial Connections

A serial interface provides a classic connection to the LCD0821. The Communication/Power Cable is most commonly used for this set up as it provides connections for DB9 serial and floppy power cables. To place your board in Serial mode, adhere to the steps laid out below.

1. Set the Protocol Select jumpers.
 - RS232: Connect the three jumpers* in the 232 protocol box with the zero ohm jumper resistors provided or an alternate wire or solder solution.
 - TTL: Connect the two jumpers* in the TTL protocol box.

***Note:** Jumpers must be removed from all protocol boxes save for the one in use.

2. Make the connections.
 - a. Connect the four pin female header of the Communication/Power Cable to the Communication/Power Header of your LCD0821.
 - b. Insert the male end of your serial cable to the corresponding DB9 header of the Communication/Power Cable and the mate the female connector with the desired communication port of your computer.
 - c. Select an unmodified floppy cable from a PC power supply and connect it to the power header of the Communication/Power Cable.
3. Create.
 - Use uProject or a terminal program to get started, and then move on with your own development. A number of application notes are available at www.matrixorbital.ca/appnotes.

I²C Connections

A more advanced connection to the LCD0821 is provided by the I²C protocol setting. This is best accomplished using a breadboard and the Breadboard Cable. Power must be supplied from your breadboard or another external source. To dive right into your application and use the LCD0821 in I²C mode, get started with the guidelines below.

1. Set the Protocol Select switches.
 - I²C: Ensure that the two I²C jumpers in the corresponding protocol box are connected while all others are open.
2. Make the connections.
 - a. Connect the Breadboard Cable to the Communication/Power Header on your LCD0821 and plug the four leads into your breadboard. The red lead will require power, while the black should be connected to ground, and the green and yellow should be connected to your controller clock and data lines respectively.
 - b. Pull up the clock and data lines to five volts using a resistance between one and ten kilohms on your breadboard.
3. Create.
 - This time you're on your own. While there are many examples within the Matrix Orbital AppNote section, www.matrixorbital.ca/appnotes, too many controllers and languages exist to cover them all. If you get stuck in development, it is possible to switch over to another protocol on the standard board, and fellow developers are always on our forums for additional support.

3 Software

The multiple communication protocols available and simple command structure of the LCD0821 means that a variety of applications can be used to communicate with the display. Text is sent to the display as a character string, for example, sending the decimal value 41 will result in an 'A' appearing on the screen. A number of control characters are also activated. Commands are merely values prefixed with a special command byte, 254 in decimal.

Table 2: Reserved Control Characters

| Control Characters | | | | | | | |
|--------------------|-----------|----|----------------------|----|-------------------------|----|-----------------|
| 8 | Backspace | 10 | Line feed / New line | 12 | Clear screen / New page | 13 | Carriage return |

Once the correct communication port is identified, the following communication settings can be applied to communicate correctly with the LCD0821.

Table 3: Communication Settings

| BPS | Data Bits | Parity | Stop Bits | Flow Control |
|-------|-----------|--------|-----------|--------------|
| 19200 | 8 | None | 1 | None |

Finally, with a communication port identified and correctly setup simple text strings or even command bytes can easily be transmitted to control your display.

3.1 uProject

The Matrix Orbital alphanumeric display tuner, or uProject, is offered as a free download from the www.matrixorbital.ca support site. It allows the basic functionality of *any display to be tested using a simple graphical user interface system.

While basic functionality can be tested using the GUI portion of the program, more advanced users will enjoy the scripting capability found in the uploader tab. Here commands can be stacked, run, and saved for later use. Although many commands are available to be dragged into the script dialog, perhaps the most powerful is the raw data command found in the other branch.

***Note:** The uProject AutoDetect function will not perform correctly when a USB display is connected. Please manually configure any USB display.

This command allows raw bytes to be sent to the display, permitting many different formats for entry and displaying in decimal notation. Any command from this manual may be entered in decimal notation separated by slashes.

/254/ /88/

Figure 5: uProject Command

Again, the clear screen command is sent to a connected display, this time using uProject raw data command style. Scripts can be run as a whole using the execute command from the script menu, or as single commands by selecting execute once. Before issuing commands, it is a good idea to ensure communication with a display is successful using some of the more basic GUI functions in the main window.

This program provides scratch pad upon which a tome of display projects and ideas can be assembled.

3.2 Application Notes

Full demonstration programs and code are available for Matrix Orbital Displays in the C# language from Simple C# AppNote Pack in the Application Note section at www.matrixorbital.ca/appnotes. Difficulty increases from beginner, with the Hello World program, to advanced with the Dallas One-Wire temperature reading application.

Many additional applications are available in a number of different programming languages. These programs are meant to showcase the capability of the display and are not intended to be integrated into a final design. For additional information regarding code, please read the On Code document also found on the support site.

4 Hardware

4.1 Standard Model

Communication/Power Header



Figure 6: Communication/Power Header

Table 4: Communication/Power Pinout

| Pin | Function |
|-----|----------|
| 1 | Vcc |
| 2 | Rx (SCL) |
| 3 | Tx (SDA) |
| 4 | Gnd |

The Communication/Power Header provides a standard connector for interfacing to the LCD0821. Voltage is applied through pins one and four of the four pin Communication/Power Header. Please ensure the correct voltage input for your display by referencing Voltage Specifications before connecting power. Pins two and three are reserved for serial transmission, using either the RS-232/TTL or clocking data through the I²C protocol, depending on what has been selected by the Protocol Select Jumpers. The versatile Tyco 640456-4-LF style header used can be mated to a number of connectors, the Molex 22-01-3047 for example.

Alternate Serial Header



Figure 7: Communication/Power Header

Table 5: Communication/Power Pinout

| Pin | Function |
|-----|----------|
| 1 | Gnd |
| 2 | Rx (SCL) |
| 3 | Vcc |

Protocol Select Jumpers

The Protocol Select Jumpers provide the means necessary to toggle the LCD0821 between RS-232, TTL and I²C protocols. As a default, the jumpers are set to RS-232 mode with solder jumps on the RS-232 jumpers. In order to place the display module in I²C mode you must first remove the solder jumps from the RS-232 jumpers and then place them on the I²C jumpers. The display will now be in I²C mode and have a default slave address of 80, unless changed with the appropriate command. Similarly, in order to change the display to TTL mode, simply remove the zero ohm resistors from the RS-232 or I²C jumpers and solder them to the TTL jumpers.

4.2 Common Features

General Purpose Outputs



Figure 8: GPO Header

Table 6: GPO Pinout

| Pin | Function |
|-----|----------|
| 1 | GPO |
| 2 | Gnd |

A unique feature of the LCD0821 is the ability to control relays* and other external devices using a single General Purpose Output. It can source up to 10mA of current at five volts when on or sink 20mA at zero volts when off. The two pin header can be interfaced to a number of female connectors to provide control to any peripheral devices required.

***Note:** 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.

5 Troubleshooting

5.1 Power

In order for your LCD0821 to function correctly, it must be supplied with the appropriate power. If the power LED near the top right corner of the board is not illuminated, power is not applied correctly. Try following the tips below.

- First, make sure that you are using the correct power connector. Standard floppy drive power cables from your PC power supply may fit on the Communication/Power Header; however they do not have the correct pin out to provide power. Matrix Orbital supplies power cable adapters for connecting to a PC, which can be found in the accessories section.
- Next, 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.
- If power is applied through the DB9 connector, ensure that the Power Through DB9 Jumper is connected.
- If changes have been made to the protocol select block, ensure all the appropriate protocol select jumpers are connected and all unused protocol jumpers are disconnected.
- The last step will be to check the interface connector in use on your display. If the power connections have become loose, or you are unable to resolve the issue, please

- Contact Matrix Orbital for more information.

5.2 Display

If your display is powered successfully, the Matrix Orbital logo, or user created screen should display on start up. If this is not the case, check out these tips.

- Ensure the contrast is not too high or too low. This can result in a darkened or blank screen respectively. See the Manual Override section to reset to default.
- Make sure that the start screen is not blank. It is possible to overwrite the Matrix Orbital logo start screen, if this happens the screen may be blank. Try writing to the display to ensure it is functional, after checking the contrast above.

5.3 Communication

When communication of either text or commands is interrupted, try the steps below.

- 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/USB Port.
- Next, please ensure that the display module is set to communicate on the protocol that you are using, by checking the Protocol Select Jumpers.
- In serial and USB protocols, ensure that the host system and display module are both communicating on the same baud rate. The default rate for the display module is 19200 bps.
- Match Rx from your display to the transmitting pin from your host and the Tx pin to the receiving pin.
- 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 80.
- In I²C mode, connect Rx to the clock line of your controller and Tx to the data output.
- Unlock the display. See the Set and Save Data Lock command for more info.
- Finally, you may reset the display to its default settings using the Manual Override procedure outlined below.

***Note:** I²C communication will always require pull up resistors on SCL and SDA of one to ten kilohms.

5.4 Manual Override

Should the settings of your display become altered in a way that dramatically impacts usability, the default settings can be temporarily restored. To override the display, please follow the steps below.

1. Disconnect power from your display.
2. Place a jumper on the manual override pins.
3. Reconnect power to your unit, and wait for the start screen before removing the jumper.
4. Settings will be temporarily** overridden to the defaults listed in the Manual Override Settings table. At this point any important settings, such as contrast, backlight, or baud rate, should not only be set but saved so they remain when the override is removed.

| Parameter | Value |
|--------------------------|-------|
| Backlight | 255 |
| Contrast | 128 |
| Baud Rate | 19200 |
| I ² C Address | 80 |

Table 7: Manual Override Settings

****Note:** The display module will revert back to the old settings once turned off, unless desired settings are saved.

6 Commands

6.1 Communication

| | | | | |
|---|-------------|-----------------------------|-------|-------------|
| 1.1 Change Baud Rate | Dec | 254 57 | Speed | v5.0 |
| | Hex | FE 39 | Speed | |
| | ASCII | ■ 9 | Speed | |
| Immediately changes the baud rate. Not available in I2C. Baud rate can be temporarily forced to 19200 by a manual override. | | | | |
| Speed | Byte | Valid settings shown below. | | |

Table 8: Accepted Baud Rate Values

| | | | | | | | | | | |
|--------------|------|------|------|------|-------|-------|-------|-------|-------|--------|
| Rate | 1200 | 2400 | 4800 | 9600 | 19200 | 28800 | 38400 | 57600 | 76800 | 115200 |
| Speed | 83 | 41 | 207 | 103 | 51 | 34 | 25 | 16 | 12 | 8 |

| | | | | |
|---|-------------|---------------|---------|-------------|
| 1.2 Change I ² C Slave Address | Dec | 254 51 | Address | v5.0 |
| | Hex | FE 33 | Address | |
| | ASCII | ■ 3 | Address | |
| Immediately changes the I ² C write address. Only even values are permitted as the next odd address will become the read address. Default is 80. | | | | |
| Address | Byte | Even value. | | |

| | | | | |
|---|-------------|---|----------|-------------|
| 1.3 Transmission Protocol Select | Dec | 254 160 | Protocol | v5.0 |
| | Hex | FE A0 | Protocol | |
| | ASCII | ■ á | Protocol | |
| Selects the protocol used for data transmission from the display. Data transmission to the display is not affected. Must be set to the protocol in use to receive data correctly. | | | | |
| Protocol | Byte | 1 for Serial (RS232/TTL) or 0 for I ² C. | | |

| | | | | |
|--|--------------|--|-------|-------------|
| 1.4 Set a Non-Standard Baud Rate | Dec | 254 164 | Speed | v5.0 |
| | Hex | FE A4 | Speed | |
| | ASCII | ■ ñ | Speed | |
| Immediately changes the baud rate to a non-standard value. Baud must be a whole number between 977 and 153800. Due to rounding, error increases with baud rate, actual baud must be within 3% of desired baud to ensure accurate communication. Not available in I2C. Can be temporarily forced to 19200 by a manual override. | | | | |
| Speed | Short | Calculations shown below, standard crystal speed is 16MHz. | | |

$$Speed = \frac{CrystalSpeed}{(8 \times DesiredBaud)} - 1 \quad ActualBaud = \frac{CrystalSpeed}{(8 \times (Speed + 1))}$$

Equation 1: Speed Byte Calculation *Equation 2: Actual Baud Rate Calculation*

$$\frac{|DesiredBaud - ActualBaud|}{DesiredBaud} < 0.03$$

Equation 3: Baud Rate Error Calculation

6.2 Text

| | | | | |
|------------------|-------|---------------|--|------|
| 2.1 Clear Screen | Dec | 254 88 | | v5.0 |
| | Hex | FE 58 | | |
| | ASCII | ■ X | | |

Clears the contents of the screen.

| | | | | |
|--------------------------------|-------|---------------|------------|------|
| 2.2 Change the Start Up Screen | Dec | 254 64 | Characters | v5.0 |
| | Hex | FE 40 | Characters | |
| | ASCII | ■ @ | Characters | |

Changes the message displayed on start up. Custom characters can be included by adding their decimal value (0-7). Characters will automatically wrap on the display.

Characters 16 bytes, space characters can be added as needed

| | | | | |
|--------------------|-------|---------------|--|------|
| 2.3 Auto Scroll On | Dec | 254 81 | | v5.0 |
| | Hex | FE 51 | | |
| | ASCII | ■ Q | | |

The entire contents of screen are shifted up one line when the end of the screen is reached. Display default is on.

| | | | | |
|---------------------|-------|---------------|--|------|
| 2.4 Auto Scroll Off | Dec | 254 82 | | v5.0 |
| | Hex | FE 52 | | |
| | ASCII | ■ R | | |

New text is written over the top line when the end of the screen is reached. Display default is Auto Scroll on.

| | | | | |
|---------------------------|-------|---------------|--|------|
| 2.5 Set Auto Line Wrap On | Dec | 254 67 | | v5.0 |
| | Hex | FE 43 | | |
| | ASCII | ■ C | | |

Text will wrap to the next consecutive line once a row becomes full. Default is Auto Line Wrap on.

| | | | | |
|----------------------------|-------|---------------|--|------|
| 2.6 Set Auto Line Wrap Off | Dec | 254 68 | | v5.0 |
| | Hex | FE 44 | | |
| | ASCII | ■ D | | |

Text will skip one line when wrapping once a row becomes full. Writing order will be rows 1, 3, 2, and then 4. Default is Auto Line Wrap on.

| | | | | |
|-------------------------|-------|---------------|------------|------|
| 2.7 Set Cursor Position | Dec | 254 71 | Column Row | v5.0 |
| | Hex | FE 47 | Column Row | |
| | ASCII | ■ G | Column Row | |

Sets the cursor to a specific cursor position where the next transmitted character is printed.

Column **Byte** Value between 1 and 8.

Row **Byte** Value between 1 and 2.

| | | | |
|---|-------|--------|------|
| 2.8 Go Home | Dec | 254 72 | v5.0 |
| | Hex | FE 48 | |
| | ASCII | ■ H | |
| Returns the cursor to the top left of the screen. | | | |

| | | | |
|--|-------|--------|------|
| 2.9 Move Cursor Back | Dec | 254 76 | v5.0 |
| | Hex | FE 4C | |
| | ASCII | ■ L | |
| Moves cursor one position to the left. Cursor will obey wrap settings. | | | |

| | | | |
|---|-------|--------|------|
| 2.10 Move Cursor Forward | Dec | 254 77 | v5.0 |
| | Hex | FE 4D | |
| | ASCII | ■ M | |
| Moves cursor one position to the right. Cursor will obey wrap settings. | | | |

| | | | |
|---|-------|--------|------|
| 2.11 Underline Cursor On | Dec | 254 74 | v5.0 |
| | Hex | FE 4A | |
| | ASCII | ■ J | |
| Displays a line under the current cursor position. Can be used with block cursor. | | | |

| | | | |
|---|-------|--------|------|
| 2.12 Underline Cursor Off | Dec | 254 75 | v5.0 |
| | Hex | FE 4B | |
| | ASCII | ■ K | |
| Removes line under current cursor position. | | | |

| | | | |
|---|-------|--------|------|
| 2.13 Blinking Block Cursor On | Dec | 254 83 | v5.0 |
| | Hex | FE 53 | |
| | ASCII | ■ S | |
| Displays a blinking block over the current cursor position. Can be used with underline. | | | |

| | | | |
|--|-------|--------|------|
| 2.14 Blinking Block Cursor Off | Dec | 254 84 | v5.0 |
| | Hex | FE 54 | |
| | ASCII | ■ T | |
| Removes blinking block over current cursor position. | | | |

6.3 Special Characters

| | | | | |
|-------------------------------|-------|---------------|---------|------|
| 3.1 Create a Custom Character | Dec | 254 78 | ID Data | v5.0 |
| | Hex | FE 4E | ID Data | |
| | ASCII | ■ N | ID Data | |

Creates a custom character. Each character is divided into 8 rows of 5 pixels; each data byte represents one row. Each byte is padded by three zero bits followed by five bits representing each pixel state. A one represents an on condition while a zero is off. Characters are lost when a new memory bank is loaded, unless they are saved.

| | | |
|-------------|----------------|--------------------------------------|
| ID | Byte | Character ID, value between 0 and 7. |
| Data | Byte[8] | Character pixel data as shown below. |

Table 9: Custom Degree Character

| | | | | | | | | |
|----------------|-----|----|----|----|----|----|----------|-----------|
| Data[1] | 000 | p1 | p2 | p3 | p4 | p5 | 00001000 | 8 |
| Data[2] | 000 | p1 | p2 | p3 | p4 | p5 | 00010100 | 20 |
| Data[3] | 000 | p1 | p2 | p3 | p4 | p5 | 00001000 | 8 |
| Data[4] | 000 | p1 | p2 | p3 | p4 | p5 | 00000011 | 3 |
| Data[5] | 000 | p1 | p2 | p3 | p4 | p5 | 00000100 | 4 |
| Data[6] | 000 | p1 | p2 | p3 | p4 | p5 | 00000100 | 4 |
| Data[7] | 000 | p1 | p2 | p3 | p4 | p5 | 00000011 | 3 |
| Data[8] | 000 | p1 | p2 | p3 | p4 | p5 | 00000000 | 0 |

| | | | | |
|----------------------------|-------|----------------|--------------|------|
| 3.2 Save Custom Characters | Dec | 254 193 | Bank ID Data | v5.0 |
| | Hex | FE C1 | Bank ID Data | |
| | ASCII | ■ ñ | Bank ID Data | |

Provides access to all memory banks to create and save custom characters, graph bars, and large digits. Any new characters saved will overwrite the old, so care should be taken when writing to any bar or digit memory bank. Bank structure is shown below.

| | | |
|-------------|----------------|--|
| Bank | Byte | 1 byte, memory bank ID, value between 0 and 4, as below. |
| ID | Byte | 1 byte, value between 0 and 7. |
| Data | Byte[8] | 8 bytes, character pixel data as above. |

Table 10: Custom Character Banks

| | | | | | | | | | |
|----------|---------------------|----------|-----------------|----------|---------------|----------|---------------|----------|--------------|
| 0 | Start-up Characters | 1 | Horizontal Bars | 2 | Vertical Bars | 3 | Medium Digits | 4 | Large Digits |
|----------|---------------------|----------|-----------------|----------|---------------|----------|---------------|----------|--------------|

| | | | | |
|----------------------------|-------|----------------|------|------|
| 3.3 Load Custom Characters | Dec | 254 192 | Bank | v5.0 |
| | Hex | FE C0 | Bank | |
| | ASCII | ■ L | Bank | |

Loads a bank of custom characters into memory for use. Must be issued before using a bank of characters. Alternatively, an appropriate initialize command can be used.

| | | |
|-------------|-------------|--|
| Bank | Byte | Memory bank ID, value between 0 and 4, as above. |
|-------------|-------------|--|

| | | | | |
|--|----------------|--|---------|-------------|
| 3.4 Save Start Up Screen Custom Characters | Dec | 254 194 | ID Data | v5.0 |
| | Hex | FE C2 | ID Data | |
| | ASCII | ■ T | ID Data | |
| Saves a custom character to memory for the start up screen or repeated use. Start up characters are displayed by sending their ID to the screen. | | | | |
| ID | Byte | Value between 0 and 7. | | |
| Data | Byte[8] | Character pixel data, see Custom Degree Character example. | | |

| | | | | |
|---|-------|----------------|--|-------------|
| 3.5 Initialize Medium Numbers | Dec | 254 109 | | v5.0 |
| | Hex | FE 6D | | |
| | ASCII | ■ m | | |
| Loads the medium number custom character bank into memory. Medium numbers must be initialized before use. | | | | |

| | | | | |
|---|-------------|----------------------------------|------------------|-------------|
| 3.6 Place Medium Numbers | Dec | 254 111 | Row Column Digit | v5.0 |
| | Hex | FE 6F | Row Column Digit | |
| | ASCII | ■ o | Row Column Digit | |
| Places a single medium decimal digit of 2 row height and 1 column width on the display at the position specified. Medium numbers must be initialized before being placed. | | | | |
| Row | Byte | Value between 1 and 8. | | |
| Column | Byte | Value between 1 and 2. | | |
| Digit | Byte | Single decimal digit to display. | | |

| | | | | |
|--|-------|----------------|--|-------------|
| 3.7 Initialize Horizontal Bar | Dec | 254 104 | | v5.0 |
| | Hex | FE 68 | | |
| | ASCII | ■ h | | |
| Loads the horizontal bar graph custom character bank into memory. Horizontal bar characters must be initialized before a graph is displayed. | | | | |

| | | | | |
|---|-------------|--|-----------------------------|-------------|
| 3.8 Place Horizontal Bar Graph | Dec | 254 124 | Column Row Direction Length | v5.0 |
| | Hex | FE 7C | Column Row Direction Length | |
| | ASCII | ■ | Column Row Direction Length | |
| Places a horizontal bar graph on the screen beginning at the column and row specified. The bar extends either right or left to the length indicated. New bars will overwrite old. | | | | |
| Column | Byte | 1 byte, value between 1 and 8. | | |
| Row | Byte | 1 byte, value between 1 and 2. | | |
| Direction | Byte | 1 byte, 0 for right and 1 for left. | | |
| Length | Byte | 1 byte, length in pixels of the graph, value between 0 and 40. | | |

| | | | | |
|--|-------|----------------|--|-------------|
| 3.9 Initialize Narrow Vertical Bar | Dec | 254 115 | | v5.0 |
| | Hex | FE 73 | | |
| | ASCII | ■ s | | |
| Loads the narrow horizontal bar graph custom character bank into memory. A narrow bar is 2 pixels wide. Horizontal bar characters must be initialized before a graph is displayed. | | | | |

| | | | | |
|-----------------------------------|-------|----------------|--|-------------|
| 3.10 Initialize Wide Vertical Bar | Dec | 254 118 | | v5.0 |
| | Hex | FE 76 | | |
| | ASCII | ■ v | | |

Loads the wide horizontal bar graph custom character bank into memory. A wide bar is 5 pixels wide. Horizontal bar characters must be initialized before a graph is displayed.

| | | | | |
|-------------------------|-------|---------------|---------------|-------------|
| 3.11 Place Vertical Bar | Dec | 254 61 | Column Length | v5.0 |
| | Hex | FE 3D | Column Length | |
| | ASCII | ■ = | Column Length | |

Places a vertical bar graph on the screen extending from the first row of the column specified. The bar extends upwards to the length indicated. A new bar will over write the old.

| | | |
|---------------|-------------|--|
| Column | Byte | Value between 1 and 8. |
| Length | Byte | Height in pixels of the graph, value between 0 and 16. |

6.4 General Purpose Output

| | | | | |
|-------------------------------|-------|---------------|--------|-------------|
| 4.1 General Purpose Output On | Dec | 254 87 | Number | v5.0 |
| | Hex | FE 57 | Number | |
| | ASCII | ■ W | Number | |

Turns the specified GPO on, sourcing current from an output of five volts.

| | | |
|---------------|-------------|----------------------|
| Number | Byte | GPO to be turned on. |
|---------------|-------------|----------------------|

| | | | | |
|--------------------------------|-------|---------------|--------|-------------|
| 4.2 General Purpose Output Off | Dec | 254 86 | Number | v5.0 |
| | Hex | FE 56 | Number | |
| | ASCII | ■ V | Number | |

Turns the specified GPO off, sinking current to an output of zero volts.

| | | |
|---------------|-------------|-----------------------|
| Number | Byte | GPO to be turned off. |
|---------------|-------------|-----------------------|

| | | | | |
|----------------------------|-------|----------------|--------------|-------------|
| 4.3 Set Start Up GPO State | Dec | 254 195 | Number State | v5.0 |
| | Hex | FE C3 | Number State | |
| | ASCII | ■ | Number State | |

Sets and saves the start up state of the specified GPO in non volatile memory. Changes will be seen on start up.

| | | |
|---------------|-------------|------------------------|
| Number | Byte | GPO to be controlled. |
| State | Byte | 1 for on or 0 for off. |

6.5 Display Functions

| | | | | |
|------------------|-------|---------------|---------|------|
| 5.1 Backlight On | Dec | 254 66 | Minutes | v5.0 |
| | Hex | FE 42 | Minutes | |
| | ASCII | ■ B | Minutes | |

Turns the display backlight on for a specified length of time. If an inverse display color is used this command will essentially turn on the text.

Minutes | **Byte** | Number of minutes to leave backlight on, a value of 0 leaves the display on indefinitely.

| | | | | |
|-------------------|-------|---------------|--|------|
| 5.2 Backlight Off | Dec | 254 70 | | v5.0 |
| | Hex | FE 46 | | |
| | ASCII | ■ F | | |

Turns the display backlight off. If an inverse display colour is used this command will turn off the text.

| | | | | |
|--------------------|-------|----------------|------------|------|
| 5.3 Set Brightness | Dec | 254 153 | Brightness | v5.0 |
| | Hex | FE 99 | Brightness | |
| | ASCII | ■ Ö | Brightness | |

Immediately sets the backlight brightness. If an inverse display color is used this represents the text colour intensity instead. Default is 255.

Brightness | **Byte** | Brightness level from 0(Dim) to 255(Bright).

| | | | | |
|-----------------------------|-------|----------------|------------|------|
| 5.4 Set and Save Brightness | Dec | 254 152 | Brightness | v5.0 |
| | Hex | FE 98 | Brightness | |
| | ASCII | ■ ÿ | Brightness | |

Immediately sets and saves the backlight brightness. Although brightness can be changed using the set command, it is reset to this saved value on start up. Default is 255.

Brightness | **Byte** | Brightness level from 0(Dim) to 255(Bright).

| | | | | |
|------------------|-------|---------------|----------|------|
| 5.5 Set Contrast | Dec | 254 80 | Contrast | v5.0 |
| | Hex | FE 50 | Contrast | |
| | ASCII | ■ P | Contrast | |

Immediately sets the contrast between background and text. If an inverse display color is used this also represents the text brightness. Default is 128.

Contrast | **Byte** | Contrast level from 0(Light) to 255(Dark).

| | | | | |
|---------------------------|-------|----------------|----------|------|
| 5.6 Set and Save Contrast | Dec | 254 145 | Contrast | v5.0 |
| | Hex | FE 91 | Contrast | |
| | ASCII | ■ æ | Contrast | |

Immediately sets and saves the contrast between background and text. Although contrast can be changed using the set command, it is reset to this saved value on start up. Default is 128.

Contrast | **Byte** | Contrast level from 0(Light) to 255(Dark).

6.6 Data Security

| | | | | |
|--|-------------|------------------------|--------|------|
| 6.1 Set Remember | Dec | 254 147 | Switch | v5.0 |
| | Hex | FE 93 | Switch | |
| | ASCII | ■ ô | Switch | |
| Allows changes to specific settings to be saved to the display memory. Writing to non-volatile memory can be slow and each change consumes 1 write of at least 100,000 available. The Command Summary outlines which commands are saved always, never, and when this command is on only. Remember is off by default. | | | | |
| Switch | Byte | 1 for on or 0 for off. | | |

| | | | | |
|--|-------------|---------------------------------------|-------|------|
| 6.2 Set Data Lock | Dec | 254 202 245 160 | Level | v5.0 |
| | Hex | FE CA F5 A0 | Level | |
| | ASCII | ■ ⌞ á | Level | |
| Temporarily locks certain aspects of the display to ensure no inadvertent changes are made. The lock is released after a power cycle. A new level overrides the old, and levels can be combined. Default is 0. | | | | |
| Level | Byte | Lock level, see Data Lock Bits table. | | |

Table 11: Data Lock Bits

| Display | Command | Reserved | Setting | Address | Reserved | Reserved | Reserved |
|---------|---------|----------|---------|---------|----------|----------|----------|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Table 12: Lock Parameters

| | |
|-----------------|--|
| Reserved | Place holders only, should be 0 |
| Address | Locks the Baud Rate and I2C address |
| Setting | Locks all settings from being saved |
| Command | Locks all commands, text can still be written |
| Display | Locks entire display, no new text can be displayed |

| | | | | |
|---|-------------|---------------------------|-------|------|
| 6.3 Set and Save Data Lock | Dec | 254 203 245 160 | Level | v5.0 |
| | Hex | FE CB F5 A0 | Level | |
| | ASCII | ■ ⌞ á | Level | |
| Locks certain aspects of the display to ensure no inadvertent changes are made. The lock is not affected by a power cycle. A new level overrides the old, and levels can be combined. Default is 0. | | | | |
| Level | Byte | See Data Lock Bits table. | | |

6.7 Miscellaneous

| | | | | |
|----------------------------|-------|---------------|------|-------------|
| 7.1 Write Customer Data | Dec | 254 52 | Data | v5.0 |
| | Hex | FE 34 | Data | |
| | ASCII | ■ 4 | Data | |

Saves a user defined block of data to non-volatile memory. Useful for storing display information for later use.

Data | **Byte [16]** | User defined data.

| | | | |
|---------------------------|-------|---------------|-------------|
| 7.2 Read Customer Data | Dec | 254 53 | v5.0 |
| | Hex | FE 35 | |
| | ASCII | ■ 5 | |

Reads data previously written to non-volatile memory. Data is only changed when written, surviving power cycles.

Response | **Byte [16]** | Previously saved user defined data.

| | | | |
|----------------------------|-------|---------------|-------------|
| 7.3 Read Version Number | Dec | 254 54 | v5.0 |
| | Hex | FE 36 | |
| | ASCII | ■ 6 | |

Causes display to respond with its firmware version number. Test.

Response | **Byte** | Convert to hexadecimal to view major and minor revision numbers.

| | | | |
|-------------------------|-------|---------------|-------------|
| 7.4 Read Module Type | Dec | 254 55 | v5.0 |
| | Hex | FE 37 | |
| | ASCII | ■ 7 | |

Causes display to respond with its module number.

Response | **Byte** | Module number, see Sample Module Type Responses for a partial list.

Table 13: Sample Module Type Responses

1 LCD0821

7 Appendix

7.1 Command Summary

Available commands below include identifying number, required parameters, the returned response and an indication of whether settings are remembered always, never, or with remember set to on.

Table 14: Communication Command Summary

| Name | Dec | Hex | ASCII | Parameters | Response | Remembered |
|---------------------------------------|-----|-----|-------|------------|----------|-------------|
| Change Baud Rate | 57 | 39 | 9 | Byte | None | Always |
| Change I ² C Slave Address | 51 | 33 | 3 | Byte | None | Always |
| Transmission Protocol Select | 160 | A0 | á | Byte | None | Remember On |
| Set a Non-Standard Baud Rate | 164 | A4 | ñ | Short | None | Always |

Table 15: Text Command Summary

| Name | Dec | Hex | ASCII | Parameters | Response | Remembered |
|----------------------------|-----|-----|-------|------------|----------|-------------|
| Clear Screen | 88 | 58 | X | None | None | Never |
| Change the Start Up Screen | 64 | 40 | @ | Byte[] | None | Always |
| Auto Scroll On | 81 | 51 | Q | None | None | Remember On |
| Auto Scroll Off | 82 | 52 | R | None | None | Remember On |
| Set Auto Line Wrap On | 67 | 43 | C | None | None | Remember On |
| Set Auto Line Wrap Off | 68 | 44 | D | None | None | Remember On |
| Set Cursor Position | 71 | 47 | G | Byte[2] | None | Never |
| Go Home | 72 | 48 | H | None | None | Never |
| Move Cursor Back | 76 | 4C | L | None | None | Never |
| Move Cursor Forward | 77 | 4D | M | None | None | Never |
| Underline Cursor On | 74 | 4A | J | None | None | Remember On |
| Underline Cursor Off | 75 | 4B | K | None | None | Remember On |
| Blinking Block Cursor On | 83 | 53 | S | None | None | Remember On |
| Blinking Block Cursor Off | 84 | 54 | T | None | None | Remember On |

Table 16: Special Character Command Summary

| Name | Dec | Hex | ASCII | Parameters | Response | Remembered |
|--|-----|-----|-------|-----------------------|----------|-------------|
| Create a Custom Character | 78 | 4E | N | Byte[9] | None | Remember On |
| Save Custom Characters | 193 | C1 | ñ | Byte[10] | None | Always |
| Load Custom Characters | 192 | C0 |  | Byte | None | Never |
| Save Start Up Screen Custom Characters | 194 | C2 |  | Byte[9] | None | Always |
| Initialize Medium Numbers | 109 | 6D | m | None | None | Never |
| Place Medium Numbers | 111 | 6F | o | Row, Col, Digit | None | Never |
| Initialize Horizontal Bar | 104 | 68 | h | None | None | Never |
| Place Horizontal Bar Graph | 124 | 7C | | Col, Row, Dir, Length | None | Never |
| Initialize Narrow Vertical Bar | 115 | 73 | s | None | None | Never |
| Initialize Wide Vertical Bar | 118 | 76 | v | None | None | Never |
| Place Vertical Bar | 61 | 3D | = | Col, Length | None | Never |

Table 17: General Purpose Output Command Summary

| Name | Dec | Hex | ASCII | Parameters | Response | Remembered |
|----------------------------|-----|-----|-------|------------|----------|------------|
| General Purpose Output On | 86 | 56 | V | Byte | None | Never |
| General Purpose Output Off | 87 | 57 | W | Byte | None | Never |
| Set Start Up GPO State | 195 | C3 |  | Byte[2] | None | Always |

Table 18: Display Functions Command Summary

| Name | Dec | Hex | ASCII | Parameters | Response | Remembered |
|-------------------------|-----|-----|-------|------------|----------|-------------|
| Backlight On | 66 | 42 | B | Byte | None | Remember On |
| Backlight Off | 70 | 46 | F | None | None | Remember On |
| Set Brightness | 153 | 99 |  | Byte | None | Remember On |
| Set and Save Brightness | 152 | 98 |  | Byte | None | Always |
| Set Contrast | 80 | 50 | P | Byte | None | Remember On |
| Set and Save Contrast | 145 | 91 |  | Byte | None | Always |

Table 19: Data Security Command Summary

| Name | Dec | Hex | ASCII | Parameters | Response | Remembered |
|------------------------|---------------|------------|---------|------------|----------|-------------|
| Set Remember | 147 | 93 |  | Byte | None | Always |
| Set Data Lock | 202, 245, 160 | CA, F5, A0 | , ,  | Byte | None | Remember On |
| Set and Save Data Lock | 203, 245, 160 | CB, F5, A0 | , ,  | Byte | None | Always |

Table 20: Miscellaneous Command Summary

| Name | Dec | Hex | ASCII | Parameters | Response | Remembered |
|---------------------|-----|-----|-------|------------|----------|------------|
| Write Customer Data | 52 | 34 | 4 | Byte[16] | None | Always |
| Read Customer Data | 53 | 35 | 5 | None | Byte[16] | Never |
| Read Version Number | 54 | 36 | 6 | None | Byte | Never |
| Read Module Type | 55 | 37 | 7 | None | Byte | Never |

7.2 Character Sets

| | | 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 | Q | P | ' | P | G | â | ' | ° | ° | ° | ° | |
| | 1 | CG RAM (2) | ≡ | ! | 1 | A | Q | a | q | o | a | i | ' | ° | ° | ° | ° |
| | 2 | CG RAM (3) | 7 | " | 2 | B | R | b | r | e | â | é | ' | ° | ° | ° | ° |
| | 3 | CG RAM (4) | ∟ | ∗ | 3 | O | S | s | â | ô | ' | ° | ° | ° | ° | ° | ° |
| | 4 | CG RAM (5) | ∟ | ∗ | 4 | D | T | t | â | â | ' | ° | ° | ° | ° | ° | ° |
| | 5 | CG RAM (6) | ∟ | ∗ | 5 | E | U | u | â | â | ' | ° | ° | ° | ° | ° | ° |
| | 6 | CG RAM (7) | ∟ | ∗ | 6 | F | V | v | â | â | ' | ° | ° | ° | ° | ° | ° |
| | 7 | CG RAM (8) | ∟ | ∗ | 7 | a | w | w | s | U | R | X | + | + | + | + | + |
| | 8 | CG RAM (1) | ∟ | ∗ | 8 | H | X | x | e | â | ' | ° | ° | ° | ° | ° | ° |
| | 9 | CG RAM (2) | ∟ | ∗ | 9 | I | Y | y | e | â | ' | ° | ° | ° | ° | ° | ° |
| | A | CG RAM (3) | ∟ | ∗ | A | J | Z | z | e | â | ' | ° | ° | ° | ° | ° | ° |
| | B | CG RAM (4) | ∟ | ∗ | B | K | C | c | i | â | ' | ° | ° | ° | ° | ° | ° |
| | C | CG RAM (5) | ∟ | ∗ | C | L | V | v | i | â | ' | ° | ° | ° | ° | ° | ° |
| | D | CG RAM (6) | ∟ | ∗ | D | M | N | n | i | â | ' | ° | ° | ° | ° | ° | ° |
| | E | CG RAM (7) | ∟ | ∗ | E | N | ^ | ^ | â | â | ' | ° | ° | ° | ° | ° | ° |
| | F | CG RAM (8) | ∟ | ∗ | F | O | _ | _ | â | â | ' | ° | ° | ° | ° | ° | ° |

Figure 9: LCD Model European 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) | | | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | | | | | |
| | 1 | CG RAM (2) | | ! | ! | ! | ! | ! | ! | ! | ! | ! | ! | ! | ! | ! | ! |
| | 2 | CG RAM (3) | | " | " | " | " | " | " | " | " | " | " | " | " | " | " |
| | 3 | CG RAM (4) | | # | # | # | # | # | # | # | # | # | # | # | # | # | # |
| | 4 | CG RAM (5) | | \$ | \$ | \$ | \$ | \$ | \$ | \$ | \$ | \$ | \$ | \$ | \$ | \$ | \$ |
| | 5 | CG RAM (6) | | % | % | % | % | % | % | % | % | % | % | % | % | % | % |
| | 6 | CG RAM (7) | | & | & | & | & | & | & | & | & | & | & | & | & | & | & |
| | 7 | CG RAM (8) | | ' | ' | ' | ' | ' | ' | ' | ' | ' | ' | ' | ' | ' | ' |
| | 8 | CG RAM (1) | | (| (| (| (| (| (| (| (| (| (| (| (| (| (|
| | 9 | CG RAM (2) | |) |) |) |) |) |) |) |) |) |) |) |) |) |) |
| | A | CG RAM (3) | | * | * | * | * | * | * | * | * | * | * | * | * | * | * |
| | B | CG RAM (4) | | + | + | + | + | + | + | + | + | + | + | + | + | + | + |
| | C | CG RAM (5) | | , | , | , | , | , | , | , | , | , | , | , | , | , | , |
| | D | CG RAM (6) | | - | - | - | - | - | - | - | - | - | - | - | - | - | - |
| | E | CG RAM (7) | | . | . | . | . | . | . | . | . | . | . | . | . | . | . |
| | F | CG RAM (8) | | / | / | / | / | / | / | / | / | / | / | / | / | / | / |

Figure 10: LCD Model Japanese Character Set

7.3 Block Diagram

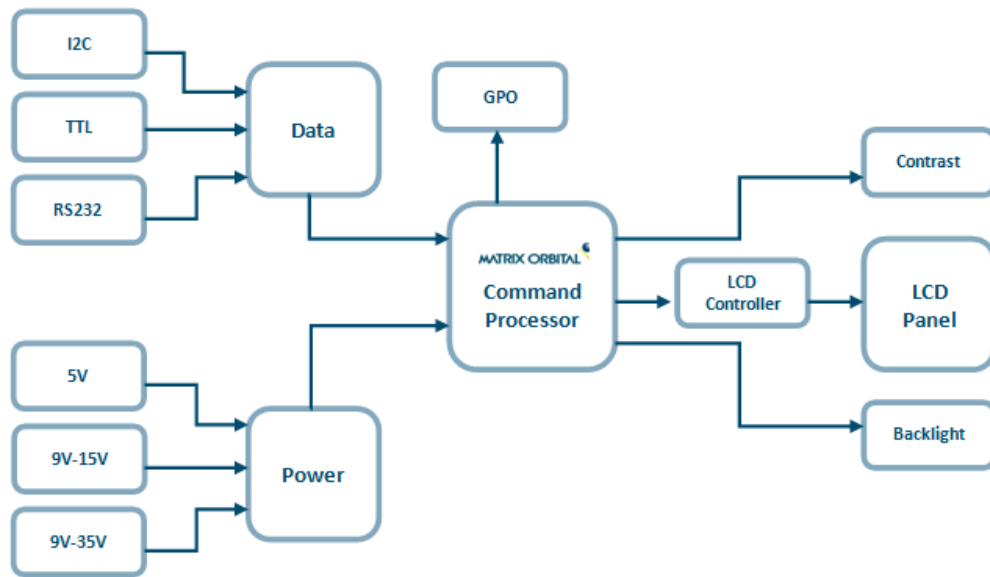


Figure 11: Functional Diagram

7.4 Environmental Specifications

Table 21: Environmental Limits

| | LCD Standard | LCD Extended (-E) |
|-----------------------------|----------------------------|-------------------|
| Operating Temperature | 0°C to +50°C | -20°C to +70°C |
| Storage Temperature | -10°C to +60°C | -30°C to +80°C |
| Operating Relative Humidity | Maximum 90% non-condensing | |
| Thermal Shock | Maximum 10°C /min | |

7.5 Electrical Tolerances

Current Consumption

Table 22: Current Consumption

| Board | + | Backlight | + | GPOs |
|-------|---|------------|---|-------------------|
| 45mA | | 20 to 65mA | | 20mA each maximum |

Table 23: Backlight Current Draw

| YG & IY | R | GW & WB |
|---------|------|---------|
| 65mA | 60mA | 20mA |

Input Voltage Specifications

Table 24: Voltage Specifications

| Standard | Wide Voltage (-V) |
|------------|-------------------|
| 4.75-5.25V | 9.0-15.0V |

7.6 Dimensional Drawings

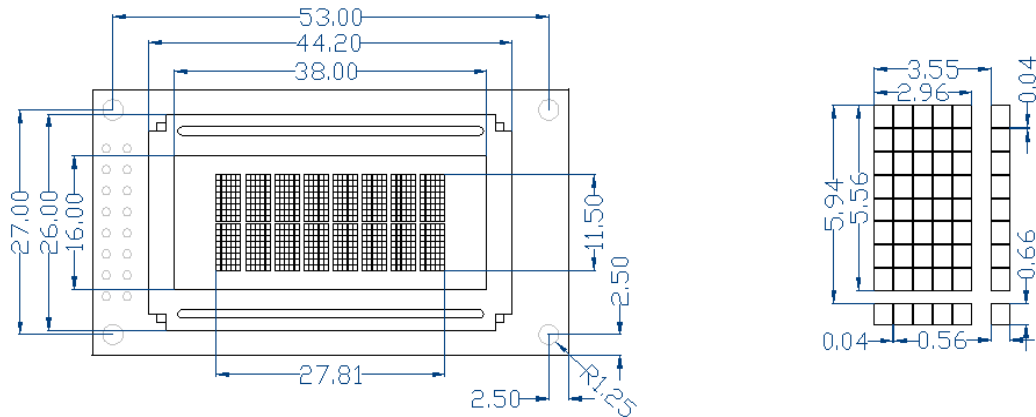


Figure 12: Display Dimensional Drawing*

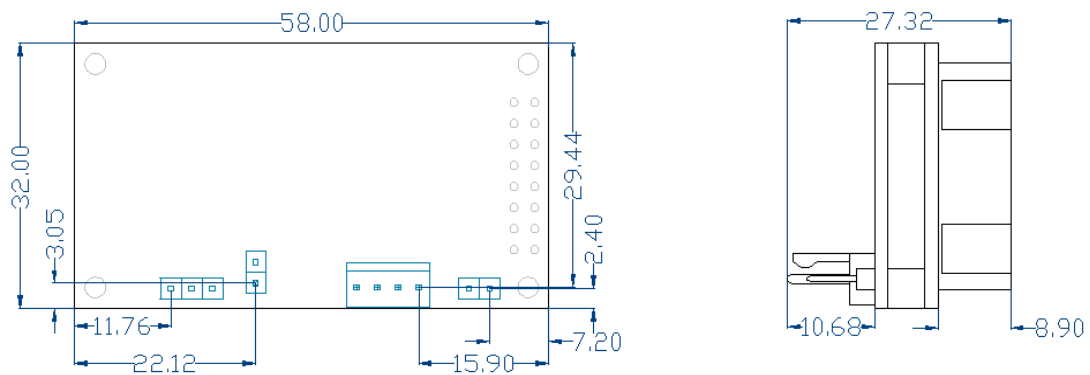


Figure 13: Standard Model Dimensional Drawing

***Note:** Mounting hole diameter was decreased to 2.5mm in 2013 as per PCN 20130410-01.

7.7 Optical Characteristics

Table 25: Display Optics

| | | |
|-------------------------------|-----------------------|---------|
| Module Size | 58.00 x 32.00 x 26.80 | mm |
| Viewing Area | 38.0 x 16.0 | mm |
| Active Area | 27.80 x 11.49 | mm |
| Character Size | 2.96 x 5.56 | mm |
| Character Pitch | 3.55 x 5.94 | mm |
| Pixel Size | 0.56 x 0.66 | mm |
| Pixel Pitch | 0.60 x 0.70 | mm |
| Viewing Direction | 12 | O'clock |
| Viewing Angle | -30 to +30 | ° |
| Contrast Ratio | 3 | |
| Backlight Half-Life (IY & YG) | 50,000 | Hours |
| Backlight Half-Life (R) | 20,000 | Hours |
| Backlight Half-Life (GW & WB) | 10,000 | Hours |

8 Ordering

8.1 Part Numbering Scheme

Table 26: Part Numbering Scheme

| LCD | 0821 | -R | -V | -E |
|-----|------|----|----|----|
| 1 | 2 | 3 | 4 | 5 |

8.2 Options

Table 27: Display Options



| # | Designator | Options |
|---|--------------|--|
| 1 | Product Type | LCD: Liquid Crystal Display |
| 2 | Display Size | 0821: 8 columns by 2 rows |
| 3 | Colour | *NP: Standard (YG for LCD/ no filter for VFD) -GW: Grey Text with Grey-White Background -WB: White Text with Blue Background -IY: Yellow-Green Text with Black Background (Inverse Display) -R: Red Text with Black Background (Inverse Display) |
| 4 | Voltage | *NP: Standard Voltage -V: Wide Voltage |
| 5 | Temperature | *NP: Standard -E: Extended Temperature |

***Note:** NP means No Populate; skip this designator in the part number and move to the next option.

8.3 Accessories

Communication

Table 28: Communication Accessories

| | | |
|---------|-------------------------------------|---|
| SCCPC5V | Serial Communication/5V Power Cable |  |
| BBC | Breadboard Cable |  |

9 Definitions

ASCII: American standard code for information interchange used to give standardized numeric codes to alphanumeric characters.

BPS: Bits per second, a measure of transmission speed.

GPO: General purpose output, used to control peripheral devices from a display.

GUI: Graphical user interface.

Hexadecimal: A base 16 number system utilizing symbols 0 through F to represent the values 0-15.

I²C: Inter-integrated circuit protocol uses clock and data lines to communicate short distances at slow speeds from a master to up to 128 addressable slave devices. A display is a slave device.

LSB: Least significant bit or byte in a transmission, the rightmost when read.

MSB: Most significant bit or byte in a transmission, the leftmost when read.

RS232: Recommended standard 232, a common serial protocol. A low level is -30V, a high is +30V.

SDA: Serial data line used to transfer data in I²C protocol. This open drain line should be pulled high through a resistor. Nominal values are between 1K and 10K Ω.

SCL: Serial clock line used to designate data bits in I²C protocol. This open drain line should be pulled high through a resistor. Nominal values are between 1K and 10K Ω.

STN: Super-twisted nematic in reference to an LCD. In a relaxed or nematic state, crystals orientate themselves in the same direction and allow light to pass. In an excited state these crystals align to block light. Super-twisted crystals move from 180 to 270 degrees providing greater contrast than TN models.

TTL: Transistor-transistor logic applied to serial protocol. Low level is 0V while high logic is 5V.

10 Contact

Sales

Phone: 403.229.2737

Email: sales@matrixorbital.ca

Support

Phone: 403.204.3750

Email: support@matrixorbital.ca

Online

Purchasing: www.matrixorbital.com

Support: www.matrixorbital.ca