# OK162-12

Including the OK162-12-USB

## Technical Manual

Revision 1.1

**PCB Revision: 3.0 or Higher (Standard Model), 1.0 or Higher (USB Model)**

**Firmware Revision: 5.1 or Higher**

## Revision History

| Revision | Date | Description | Author |
|---|---|---|---|
| 1.1 | January 4, 2018 | Correction to Set Non-Standard Baud Rate command | Divino |
| 1.0 | January 18, 2016 | Initial Release | Divino |

# 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

$I^2C$ Connections ..... 4

2.3 USB Module ..... 5

Recommended Parts ..... 5

USB Connections ..... 5

3 Software ..... 6

3.1 uProject ..... 6

3.2 Application Notes ..... 7

4 Hardware ..... 8

4.1 Standard Model ..... 8

Communication/Power Header ..... 8

Serial DB9 Connector ..... 8

Protocol Select Jumpers ..... 9

4.2 USB Model ..... 9

Mini USB Connector ..... 9

Alternate USB Header ..... 10

Alternate Power Connector ..... 10

4.3 Common Features ..... 10

General Purpose Input/Output ..... 10

Dallas One-Wire Connector ..... 11

5 Troubleshooting ..... 12

5.1 Power ..... 12

5.2 Display ..... 12

5.3 OLED Burn-In ..... 12

5.4 Communication ..... 13

5.5 Manual Override ..... 13
6 Commands ..... 14
6.1 Communication ..... 14
6.2 Text ..... 15
6.3 Special Characters ..... 17
6.4 General Purpose Output ..... 19
6.5 Dallas One-Wire ..... 20
6.6 Keypad ..... 21
6.7 Display Functions ..... 22
6.8 Data Security ..... 23
6.9 Miscellaneous ..... 24
7 Appendix ..... 25
7.1 Command Summary ..... 25
7.2 Character Sets ..... 28
7.3 Block Diagram ..... 29
7.4 Environmental Specifications ..... 29
7.5 Electrical Tolerances ..... 29
7.6 Dimensional Drawings ..... 30
7.7 Optical Characteristics ..... 32
8 Ordering ..... 32
8.1 Part Numbering Scheme ..... 32
8.2 Options ..... 32
8.3 Accessories ..... 33
9 Definitions ..... 35
10 Contact ..... 35

# 1 Introduction

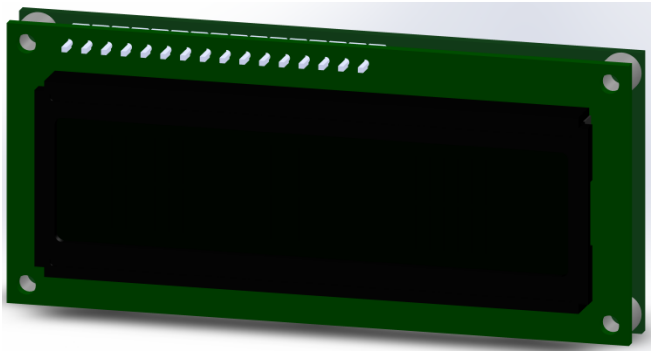


*Figure 1: OK162-12 Display*

The OK162-12 is an intelligent alphanumeric organic light emitting diode display designed to decrease development time by providing an instant solution to any project. In addition to the RS232, TTL and I2C protocols available in the standard model, the USB communication model allows the OK162-12 to be connected to a wide variety of host controllers. Communication speeds of up to 115.2kbps for serial protocols and 100kbps for $I^2C$ ensure lightning fast display updates.

The simple command structure permits easy software control of many settings including screen brightness 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.

User input on the OK162-12 is available through a three by four matrix style keypad, and seven general purpose outputs provide simple switchable five volt sources. In addition, the option of a Dallas One-Wire header provides a communication interface for up to thirty-two devices.

The versatile OK162-12, 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

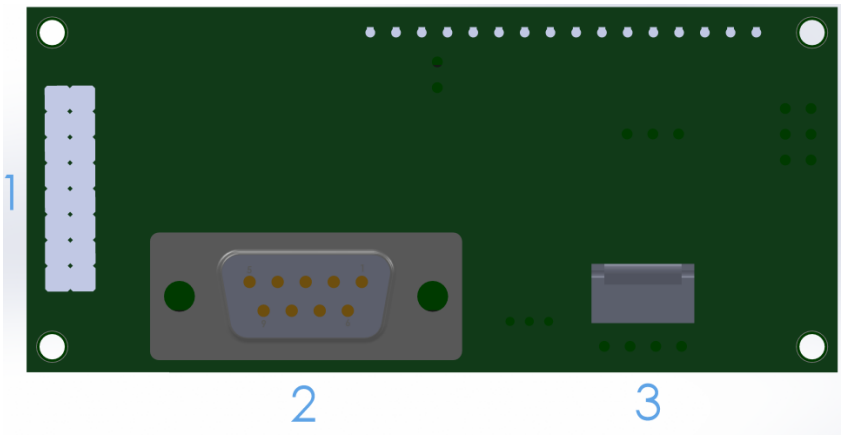



*Figure 2: OK162-12 Header Locations*

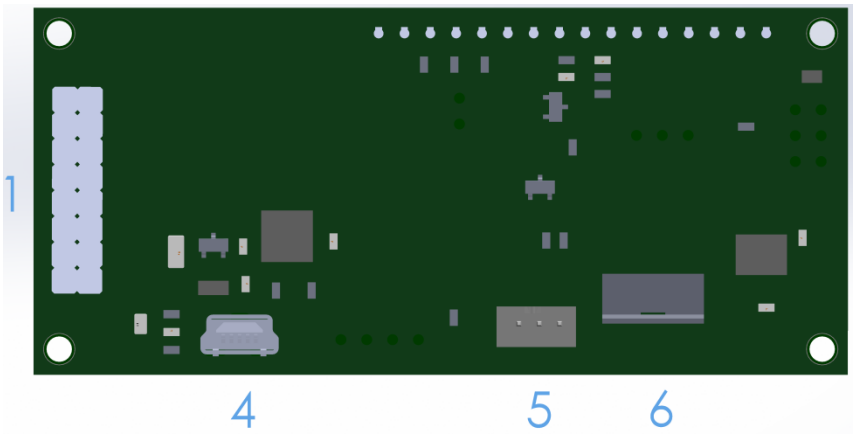



*Figure 3: OK162-12-USB Header Locations*

*Table 1: List of Available Headers*

| # | Header | Mate | Population |
|---|---|---|---|
| 1 | GPIO Header | None Offered | All Models |
| 2 | DB9 Serial Header | CSS1FT/CSS4FT | Serial Model Only |
| 3 | Communication/Power Connector | SCCPC5V/BBC | Serial Model Only |
| 4 | Mini USB Connector | EXTMUSB3FT/INTMUSB3FT | USB Model Only |
| 5 | Dallas One-Wire | Temperature Probe | USB Model Only |
| 6 | Alternate Power Connector | PCS | USB Model Only |

## 2.2 Standard Module

The OK162-12 allows for user configuration of two common communication protocols. First, the unit can communicate using serial protocol at either RS323 or TTL voltage levels. Second, it can communicate using the Inter-Integrated Circuit connect, or $I^2C$ protocol. Connections for each protocol can be accessed through the four pin Communication/Power Header as outlined in the Serial Connections and $I^2C$ Connections sections below.

### Recommended Parts

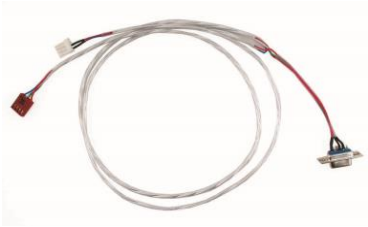


*Figure 4: Communication/Power Cable (SCCPC5V)*

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 5: Breadboard Cable (BBC)*

For a more flexible interface to the OK162-12 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 OK162-12. 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 OK162-12.
    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.

### $I^2C$ Connections

A more advanced connection to the OK162-12 is provided by the $I^2C$ 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 OK162-12 in $I^2C$ mode, get started with the guidelines below.

1. Set the Protocol Select switches.
    - $I^2C$: Ensure that the two $I^2C$ 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 OK162-12 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.

## 2.3 USB Module

The OK162-12-USB offers a single USB protocol for an easy connection to a host computer. This simple and widely available protocol can be accessed using the on board mini B style USB connector as outlined in the USB Connections section.

### Recommended Parts

*Figure 6: External Mini USB Cable (EXTMUSB3FT)*

The External Mini USB cable is recommended for the OK162-12-USB display. It will connect to the miniB style header on the unit and provide a connection to a regular A style USB connector, commonly found on a PC.

### USB Connections

The USB connection is the quickest, easiest solution for PC development. After driver installation, the OK162-12-USB will be accessible through a virtual serial port, providing the same result as a serial setup without the cable hassle. To connect to your OK162-12-USB please follow the steps below.

1. Set the Protocol Select jumpers.
   - USB: The OK162-12-USB offers USB protocol only. Model specific hardware prevents this unit from operating in any other protocol, and does not allow other models to operate in USB. Protocol Select jumpers on the USB model cannot be moved.
2. Make the connections.
   - Plug the mini-B header of your External Mini USB cable into your OK162-12-USB and the regular USB header into your computer USB jack.
3. Install the drivers.
   a. Download the latest drivers at www.matrixorbital.ca/drivers, and save them to a known location.
   b. When prompted, install the USB bus controller driver automatically
   c. If asked, continue anyway, even though the driver is not signed
   d. When the driver install is complete, your display will turn on, but communication will not yet be possible.
   e. At the second driver prompt, install the serial port driver automatically
   f. Again, if asked, continue anyway
4. Create.
   - Use uProject or a terminal program to get started, and then move on with your own development. Instructions for the former can be found below and a number of application notes are available for the latter at www.matrixorbital.ca/appnotes.

# 3 Software

The multiple communication protocols available and simple command structure of the OK162-12 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 OK162-12.

*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.

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 7: uProject Command*

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

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

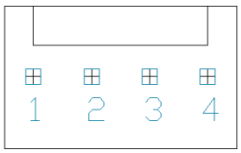



*Figure 8: Communication/Power Header*

*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 OK162-12. 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.

### Serial DB9 Connector

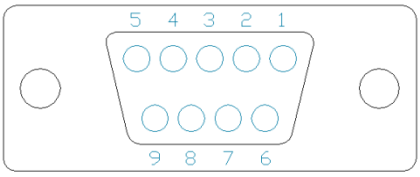



*Figure 9: Serial DB9 Connector*

*Table 5: Serial DB9 Pinout*

| Pin | Function |
|---|---|
| 2 | Tx |
| 3 | Rx |
| 5 | Gnd |
| 9 | NC/Vcc* |

The OK162-12 provides a DB-9 Connector to readily interface with serial devices using EIA232 standard signal levels. 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. A standard male DB9 header will provide the perfect mate for this connector.

***Note:** Do not apply voltage through pin 9 of the DB-9 Connector AND through the Communication/Power Header at the same time.

### Power Through DB9 Jumper

In order to provide power through pin 9 of the DB-9 Connector you must connect the Power Through DB-9 Jumper labelled D, as illustrated below. This connection can be made using a zero ohm resistor, recommended size 0603, or a solder bridge. The OK162-12 allows all voltage models to use the power through DB-9 option, see the Voltage Specifications for power requirements.

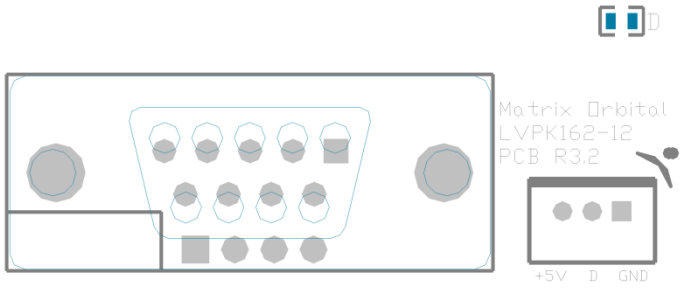



*Figure 10: Power Through DB9 Jumper*

### Protocol Select Jumpers

The Protocol Select Jumpers provide the means necessary to toggle the OK162-12 between RS-232, TTL and I²C protocols. As a default, the jumpers are set to RS-232 mode with solder jumps on the RS232 jumpers. In order to place the display module in I²C mode you must first remove the solder jumps from the RS232 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 RS232 or I²C jumpers and solder them to the TTL jumpers.

## 4.2 USB Model

### Mini USB Connector

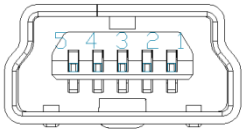



*Figure 11: Mini USB Connector*

*Table 6: Mini USB Pinout*

| Pin | Function |
|---|---|
| 1 | Vcc |
| 2 | D- |
| 3 | D+ |
| 5 | Gnd |

The OK162-12-USB comes with a familiar Mini USB Connector to fulfill both communication and power needs. The standard MiniB style header can be connected to any other USB style using the appropriate cable. Most commonly used with a PC, this connection creates a virtual com port that offers a simple power solution with a familiar communication scheme.

### Alternate USB Header

Some advanced applications may prefer the straight four pin connection offered through the Optional Alternate USB Header. This header offers power and communication access in a simple interface package. The Optional Alternate USB Header may be added to the OK162-12-USB for an added charge as part of a custom order. Please use the Contact section to request more information from the friendly Matrix Orbital sales team.

### Alternate Power Connector

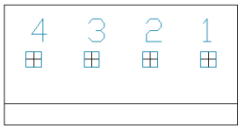



*Figure 12: Alternate Power Connector*

*Table 7: Alternate Power Pinout*

| Pin | Function |
|---|---|
| 1 | Vcc |
| 2 | Gnd |
| 3 | Gnd |
| 4 | -V/-VPT* |

The Alternate Power Connector provides the ability to power the OK162-12-USB using a second cable. The Tyco 171825-4 style header is particularly useful for connecting to an unmodified floppy power cable, a 171822-4 for example, from a PC power supply for a simple bench power solution.

***Note:** If a –V or –VPT variant of the OK162-12 is used, power can be applied to pin four of the Alternate Power Connector. Please see the Voltage Specifications for power requirements for more information.

## 4.3 Common Features

### General Purpose Input/Output

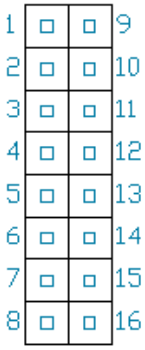



*Figure 13: GPIO Header*

*Table 8: GPIO Pinout*

| Pin | Function | Pin | Function |
|---|---|---|---|
| 1 | GPO 1/Row 1 | 9 | Gnd |
| 2 | GPO 2/Row 2 | 10 | Gnd |
| 3 | GPO 3/Row 3 | 11 | Gnd |
| 4 | GPO 4/ Col 1 | 12 | Gnd |
| 5 | GPO 5/ Col 2 | 13 | Gnd |
| 6 | GPO 6/ Col 3 | 14 | Gnd |
| 7 | GPO 7/Col 4 | 15 | Gnd |
| 8 | Vcc | 16 | Gnd |

A unique feature of the OK162-12 is the ability to control relays* and other external devices using either one or seven General Purpose Outputs. Each can source up to 10mA of current at five volts when on or sink 20mA at zero volts when off.

In keypad input mode, a matrix style keypad of up to twelve keys to be directly connected to the display module. Key presses are generated when a short is detected between a row and a column. When a key press is generated, a character specific to that key press is automatically sent on the Tx communication

line. If the display module is running in I²C mode, the “Auto Transmit Keypress” function may be turned off to allow the key presses to remain in the buffer so that they may be polled. The character that is associated with each key press may also be altered using the “Assign Key Codes” command.

The straight fourteen pin header of the General Purpose Input/Output connector will interface to a variety of different devices including the Matrix Orbital KPP4x4 keypad.

***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.

### Dallas One-Wire Connector

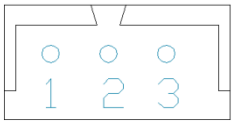



*Figure 14: Dallas One-Wire Connector*

*Table 9: Dallas One-Wire Pinout*

| Pin | Function |
|---|---|
| 1 | Vcc |
| 2 | D |
| 3 | Gnd |

In addition to the six general purpose outputs the OK162-12 offers an Optional Dallas One-Wire bridge, to allow for an additional thirty two one-wire devices to be connected to the display. This header can be populated with a Tyco 173979 connector at an added cost by custom order only. Please use the Contact section to request more information from the Matrix Orbital sales team.

# 5 Troubleshooting

## 5.1 Power

In order for your Matrix Orbital Display to function correctly, it must be supplied with the appropriate power. If the display is not illuminated, power is not applied correctly. Try following the tips below.

- First, 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 brightness is not too low. This can result in 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 brightness above.

## 5.3 OLED Burn-In

OLED technology is susceptible to a burn-in effect; whereby the brightness of active pixels may differ from inactive ones over an extended period of activation. To reduce the burn-in effect, please use a dynamic screen saver or simply turn the screen off when not in active use.

## 5.4 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 $I^2C$***** please ensure that the data is being sent to the correct address. The default slave address for the display module is 80.
- In $I^2C$ 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:** $I^2C$ communication will always require pull up resistors on SCL and SDA of one to ten kilohms.

## 5.5 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 GPIO pins labelled 3 and 4.
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 brightness, or baud rate, should not only be set but saved so they remain when the override is removed.

| Parameter | Value |
|---|---|
| Brightness | 255 |
| Baud Rate | 19200 |
| $I^2C$ Address | 80 |

*Table 10: 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 | | | | v5.1 |
|---|---|---|---|---|
| | Dec | **254 57** | Speed | |
| | 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 11: 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 $I^2C$ Slave Address | | | | v5.1 |
|---|---|---|---|---|
| | Dec | **254 51** | Address | |
| | Hex | **FE 33** | Address | |
| | ASCII | **■ 3** | Address | |
| Immediately changes the $I^2C$ 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 | | | | v5.1 |
|---|---|---|---|---|
| | Dec | **254 160** | Protocol | |
| | 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/RS422/TTL/USB) or 0 for $I^2C$. | | |

| 1.4 Set a Non-Standard Baud Rate | | | | v5.0 |
|---|---|---|---|---|
| | Dec | **254 164** | Speed | |
| | 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. | | |

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

*Equation 1: Speed Byte Calculation*

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

*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 | | v5.1 |
|---|---|---|---|---|
| | Hex | FE 58 | | |
| | ASCII | ■ X | | |

Clears the contents of the screen.

| 2.2 Change the Start Up Screen | Dec | 254 64 | Characters | v5.1 |
|---|---|---|---|---|
| | 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 | 32 bytes, space characters can be added as needed |
|---|---|

| 2.3 Auto Scroll On | Dec | 254 81 | v5.1 |
|---|---|---|---|
| | 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 | v5.1 |
|---|---|---|---|
| | 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 | v5.1 |
|---|---|---|---|
| | 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 | v5.1 |
|---|---|---|---|
| | 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 | v5.1 |
|---|---|---|---|---|
| | 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 number of character columns. |
|---|---|---|
| Row | Byte | Value between 1 and number of character rows. |

| 2.8 Go Home | Dec | 254 72 | v5.1 |
|---|---|---|---|
| | Hex | FE 48 | |
| | ASCII | ■ H | |

Returns the cursor to the top left of the screen.

| 2.9 Move Cursor Back | Dec | 254 76 | v5.1 |
|---|---|---|---|
| | 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 | v5.1 |
|---|---|---|---|
| | 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 | v5.1 |
|---|---|---|---|
| | Hex | FE 4A | |
| | ASCII | ■ J | |

Displays a line under the current cursor position. Can be used with blinking cursor.

| 2.12 Underline Cursor Off | Dec | 254 75 | v5.1 |
|---|---|---|---|
| | Hex | FE 4B | |
| | ASCII | ■ K | |

Removes line under current cursor position.

| 2.13 Blinking Cursor On | Dec | 254 83 | v5.1 |
|---|---|---|---|
| | Hex | FE 53 | |
| | ASCII | ■ S | |

Displays a blinking cursor over the current cursor position. Can be used with underline.

| 2.14 Blinking Cursor Off | Dec | 254 84 | v5.1 |
|---|---|---|---|
| | Hex | FE 54 | |
| | ASCII | ■ T | |

Removes blinking cursor over current cursor position.

## 6.3 Special Characters

| 3.1 Create a Custom Character | Dec | 254 78 | ID Data | v5.1 |
|---|---|---|---|---|
| | 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 12: 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 | v5.1 |
|---|---|---|---|---|
| | 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 13: 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 | v5.1 |
|---|---|---|---|---|
| | Hex | FE C0 | Bank | |
| | ASCII | ■ └ | 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 | v5.1 |
|---|---|---|---|---|
| | Hex | FE C2 | ID Data | |
| | ASCII | ■ ┬ | 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 | v5.1 |
|---|---|---|---|
| | 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 | v5.1 |
|---|---|---|---|---|
| | 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 16. |
| Column | Byte | Medium numbers can be placed on row 1 only. |
| Digit | Byte | Single decimal digit to display. |

| 3.7 Initialize Horizontal Bar | Dec | 254 104 | v5.1 |
|---|---|---|---|
| | 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 | v5.1 |
|---|---|---|---|---|
| | 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 16. |
| 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 100. |

| 3.9 Initialize Narrow Vertical Bar | Dec | 254 115 | v5.1 |
|---|---|---|---|
| | 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 | | v5.1 |
|---|---|---|---|---|
| | 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 | v5.1 |
|---|---|---|---|---|
| | 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 16. | | |
| 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 | v5.1 |
|---|---|---|---|---|
| | 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 | v5.1 |
|---|---|---|---|---|
| | 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 | v5.1 |
|---|---|---|---|---|
| | 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. | | |

| 4.4 Set GPO Mode | Dec | 254 37 | Mode | v5.1 |
|---|---|---|---|---|
| | Hex | FE 25 | Mode | |
| | ASCII | ■ % | Mode | |
| Toggle the single GPIO header between general purpose output and keypad input. | | | | |
| Mode | Byte | General Purpose Output/Keypad Input Mode. 1 for GPO or 0 for keypad. | | |

## 6.5 Dallas One-Wire

| 5.1 Search for a One-Wire Device | Dec | 254 200 2 | v5.1 |
|---|---|---|---|
| | Hex | FE C8 02 | |
| | ASCII | ■ ╚ SOT | |

Sends a search query to each of the up to 32 devices on the one wire bus. Any connected device will respond with an identification packet.

| Response | Bytes [14] | Dallas One-Wire identification packet as shown below. |
|---|---|---|

*Table 14: Dallas One-Wire Packet Information*

| Offset | Length | Value | Description |
|---|---|---|---|
| 0 | 2 | 9002 | Preamble |
| 2 | 1 | 138 | Another device packet will follow OR |
| | | 10 | Last device packet |
| 3 | 1 | 49 | Packet Type |
| 4 | 1 | 0 | Error Code (0 indicates success) |
| 5 | 8 | | Device Address |
| 13 | 1 | 0 | CRC8 address check (0 indicates validity) |

| 5.2 Dallas One-Wire Transaction | Dec | 254 200 1 | Flags Send Bits Receive Bits Data | v5.1 |
|---|---|---|---|---|
| | Hex | FE C8 01 | Flags Send Bits Receive Bits Data | |
| | ASCII | ■ ╚ STX | Flags Send Bits Receive Bits Data | |

Performs a single Dallas 1-Wire transaction. Consult your device documentation for information regarding device specific protocols. If an error is encountered, a corresponding value will be returned by the device.

| Flags | Byte | Flags for transaction, see below. |
|---|---|---|
| Send Bits | Byte | Number of bytes to be sent to the device. |
| Receive Bits | Byte | Number of bytes expected to be received from the device. |
| Data | Byte(s) | Data to be transmitted LSB to MSB. |

*Table 15: Dallas One-Wire Flags*

| Bit | Flag Description |
|---|---|
| 7 | |
| 6 | Unused |
| 5 | |
| 4 | 0 (Future Compatibility) |
| 3 | Add CRC8 to transaction |
| 2 | 0 (Future Compatibility) |
| 1 | Read CRC8 from transaction |
| 0 | Reset Bus prior to transaction |

*Table 16: Dallas One-Wire Errors*

| Code | Error Description |
|---|---|
| 0 | Success |
| 1 | Unknown Command |
| 2 | No Devices Found |
| 3 | Fatal Search Error |

## 6.6 Keypad

| 6.1 Auto Transmit Key Presses On | | | | v5.1 |
|---|---|---|---|---|
| | Dec | 254 65 | | |
| | Hex | FE 41 | | |
| | ASCII | ■ A | | |

Key presses are automatically sent to the host when received by the display. Default is Auto Transmit on.

| 6.2 Auto Transmit Key Presses Off | | | | v5.1 |
|---|---|---|---|---|
| | Dec | 254 79 | | |
| | Hex | FE 4F | | |
| | ASCII | ■ O | | |

Key presses are held in the 10 key buffer to be polled by the host using the Poll Key Press command. Use this mode for I2C transactions. Default is Auto Transmit on.

| 6.3 Poll Key Press | | | | v5.1 |
|---|---|---|---|---|
| | Dec | 254 38 | | |
| | Hex | FE 26 | | |
| | ASCII | ■ & | | |

Reads the last unread key press from the 10 key display buffer. If another key is stored in the buffer the MSB will be 1, the MSB will be 0 when the last key press is read. If there are no stored key presses a value of 0 will be returned. Auto transmit key presses must be turned off for this command to be successful.

| Response | Byte | Value of key pressed (MSb determines additional keys to be read). |
|---|---|---|

| 6.4 Clear Key Buffer | | | | v5.1 |
|---|---|---|---|---|
| | Dec | 254 69 | | |
| | Hex | FE 45 | | |
| | ASCII | ■ E | | |

Clears all key presses from the key buffer.

| 6.5 Set Debounce Time | | | | v5.1 |
|---|---|---|---|---|
| | Dec | 254 85 | Time | |
| | Hex | FE 55 | Time | |
| | ASCII | ■ U | Time | |

Sets the time between a key press and a key read by the display. Most switches will bounce when pressed; the debounce time allows the switch to settle for an accurate read. Default is 8 representing approximately 52ms.

| Time | Byte | Debounce increment (debounce time = Time * 6.554ms). |
|---|---|---|

| 6.6 Set Auto Repeat Mode | | | | v5.1 |
|---|---|---|---|---|
| | Dec | 254 126 | Mode | |
| | Hex | FE 7E | Mode | |
| | ASCII | ■ DEL | Mode | |

Sets key press repeat mode to typematic or hold. In typematic mode if a key press is held, the key value is transmitted immediately, then 5 times a second after a 1 second delay. In hold mode, the key down value is transmitted once when pressed, and then the key up value is sent when the key is released. Default is typematic.

| Mode | Byte | 1 for hold mode or 0 for typematic. |
|---|---|---|

| 6.7 Auto Repeat Mode Off | Dec | 254 96 | | v5.1 |
|---|---|---|---|---|
| | Hex | FE 60 | | |
| | ASCII | ■ ` | | |
| Turns auto repeat mode off. Default is on (typematic). | | | | |

| 6.8 Assign Keypad Codes | Dec | 254 213 | Key Down Key Up | v5.1 |
|---|---|---|---|---|
| | Hex | FE D5 | Key Down Key Up | |
| | ASCII | ■ Ƒ | Key Down Key Up | |
| Assigns the key down and key up values sent to the host when a key press is detected. A key up and key down value must be sent for every key, a value of 255 will leave the key unaltered. Defaults are shown below. | | | | |
| Key Down | Bytes [12] | Key down values, beginning at row one column one moving right then down. | | |
| Key Up | Bytes [12] | Key up values, beginning at row one column one moving right then down. | | |

*Table 17: Default Key Down Values*

| Key Down | | | |
|---|---|---|---|
| A(65) | B(66) | C(67) | D(68) |
| E(69) | F(70) | G(71) | H(72) |
| I(73) | J(74) | K(75) | L(76) |

*Table 18: Default Key Up Values*

| Key Up | | | |
|---|---|---|---|
| a(97) | b(98) | c(99) | d(100) |
| e(101) | f(102) | g(103) | h(104) |
| i(105) | j(106) | k(107) | l(108) |

## 6.7 Display Functions

| 7.1 Brightness On | Dec | 254 66 | Minutes | v5.1 |
|---|---|---|---|---|
| | Hex | FE 42 | Minutes | |
| | ASCII | ■ B | Minutes | |
| Turns the display brightness 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 brightness on, a value of 0 leaves the display on indefinitely. | | |

| 7.2 Brightness Off | Dec | 254 70 | v5.1 |
|---|---|---|---|
| | Hex | FE 46 | |
| | ASCII | ■ F | |
| Turns the display brightness off. If an inverse display colour is used this command will turn off the text. | | | |

| 7.3 Set Brightness | Dec | 254 153 | Brightness | v5.1 |
|---|---|---|---|---|
| | Hex | FE 99 | Brightness | |
| | ASCII | ■ Ö | Brightness | |
| Immediately sets the 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). | | |

| 7.4 Set and Save Brightness | Dec | 254 152 | Brightness | v5.1 |
|---|---|---|---|---|
| | Hex | FE 98 | Brightness | |
| | ASCII | ■ ÿ | Brightness | |
| Immediately sets and saves the 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). | | |

## 6.8 Data Security

| 8.1 Set Remember | Dec | 254 147 | Switch | v5.1 |
|---|---|---|---|---|
| | 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. | | |

| 8.2 Set Data Lock | Dec | 254 202 245 160 | Level | v5.1 |
|---|---|---|---|---|
| | 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 19: Data Lock Bits*

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

*Table 20: 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 |

| 8.3 Set and Save Data Lock | Dec | 254 203 245 160 | Level | v5.1 |
|---|---|---|---|---|
| | 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. | | |

## 6.9 Miscellaneous

| 9.1 Write Customer Data | Dec | 254 52 | Data | v5.1 |
|---|---|---|---|---|
| | 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. | | |

| 9.2 Read Customer Data | Dec | 254 53 | v5.1 |
|---|---|---|---|
| | 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. | |

| 9.3 Read Version Number | Dec | 254 54 | v5.1 |
|---|---|---|---|
| | 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. | |

| 9.4 Read Module Type | Dec | 254 55 | v5.1 |
|---|---|---|---|
| | 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 21: Sample Module Type Responses*

| | |
|---|---|
| 16 | OK162-12 |
| 45 | OK162-12-USB |

# 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 22: Communication* Command Summary

| Name | Dec | Hex | ASCII | Parameters | Response | Remembered |
|---|---|---|---|---|---|---|
| Change Baud Rate | 57 | 39 | 9 | Byte | None | Always |
| Change $I^2C$ 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 23: 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 Cursor On | 83 | 53 | S | None | None | Remember On |
| Blinking Cursor Off | 84 | 54 | T | None | None | Remember On |

*Table 24: 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 25: 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 |
| Set GPO Mode | 37 | 25 | % | Byte | None | Remember On |

*Table 26: Dallas One-Wire Command Summary*

| Name | Dec | Hex | ASCII | Parameters | Response | Remembered |
|---|---|---|---|---|---|---|
| Search for a One-Wire Device | 200, 2 | C8, 02 | └, SOT | None | Byte[14] | Never |
| Dallas One-Wire Transaction | 200, 1 | C8, 01 | └, STX | Byte[3], Byte[] | Byte[] | Never |

*Table 27: Keypad* Command Summary

| Name | Dec | Hex | ASCII | Parameters | Response | Remembered |
|---|---|---|---|---|---|---|
| Auto Transmit Key Presses On | 65 | 41 | A | None | None | Remember On |
| Auto Transmit Key Presses Off | 79 | 4F | ` | None | None | Remember On |
| Poll Key Press | 38 | 26 | & | None | Byte | Never |
| Clear Key Buffer | 69 | 45 | E | None | None | Never |
| Set Debounce Time | 85 | 55 | U | Byte | None | Remember On |
| Set Auto Repeat Mode | 126 | 7E | DEL | Mode | None | Remember On |
| Auto Repeat Mode Off | 96 | 60 | ` | None | None | Remember On |
| Assign Keypad Codes | 213 | D5 | ┌ | Byte[25], Byte[25] | None | Always |

*Table 28: Display Functions Command Summary*

| Name | Dec | Hex | ASCII | Parameters | Response | Remembered |
|---|---|---|---|---|---|---|
| Brightness On | 66 | 42 | B | Byte | None | Remember On |
| Brightness Off | 70 | 46 | F | None | None | Remember On |
| Set Brightness | 153 | 99 | Ö | Byte | None | Remember On |
| Set and Save Brightness | 152 | 98 | ÿ | Byte | None | Always |

Table 29: Data Security Command Summary

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

Table 30: 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

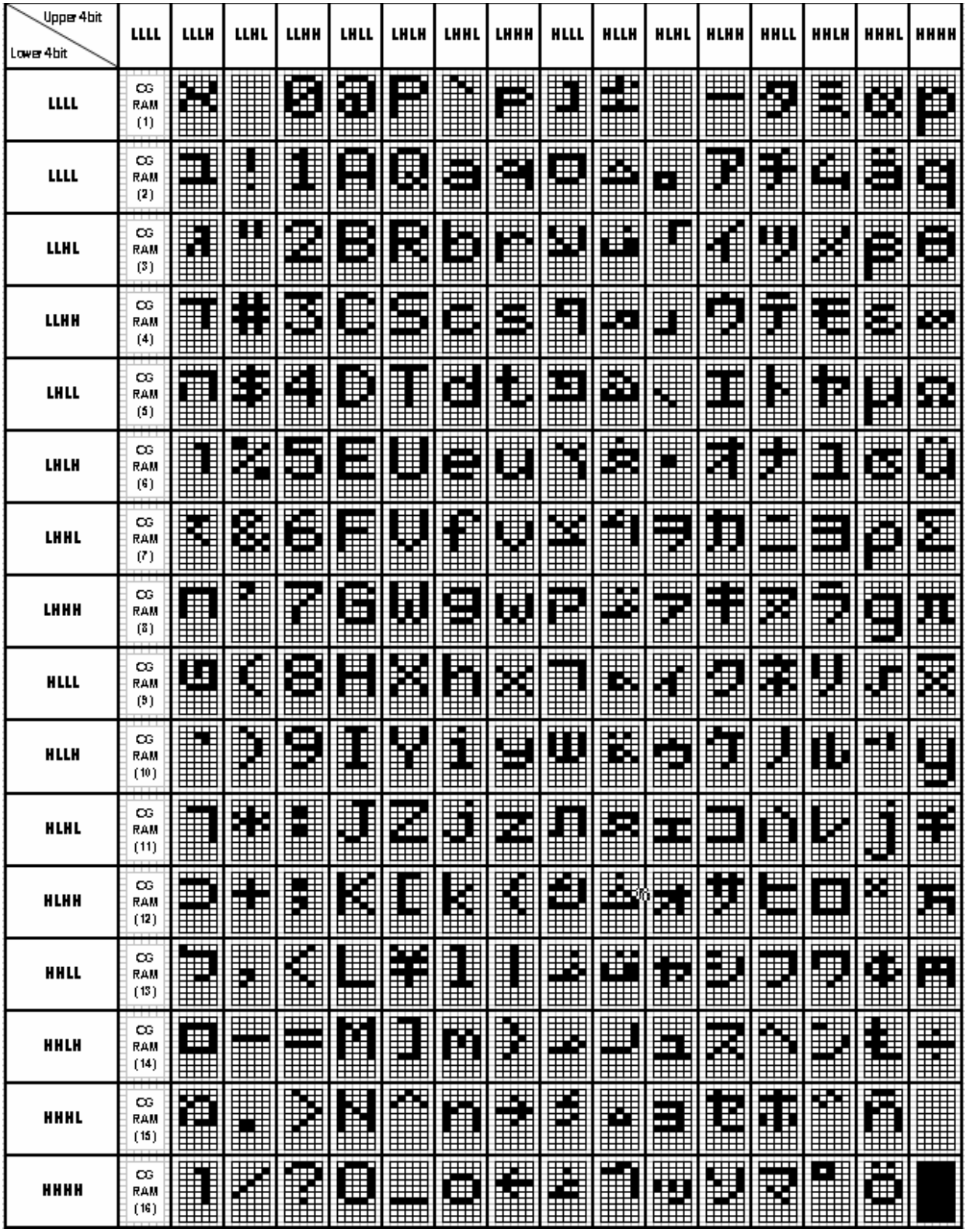



*Figure 15: English Japanese Character Set*

## 7.3 Block Diagram

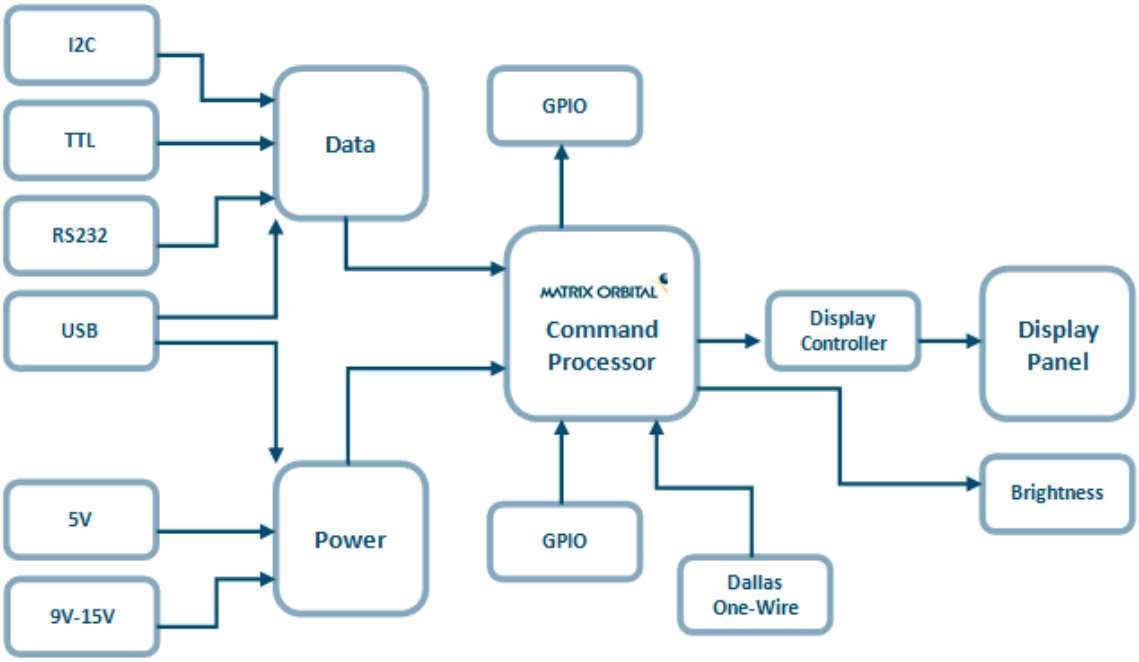



*Figure 16: Functional Diagram*

## 7.4 Environmental Specifications

*Table 31: Environmental Limits*

| | OLED Standard | OLED Extended (-E) |
|---|---|---|
| Operating Temperature | -20°C to +70°C | -40°C to +80°C |
| Storage Temperature | -40°C to +80°C | -40°C to +80°C |
| Operating Relative Humidity | Maximum 90% non-condensing | |

## 7.5 Electrical Tolerances

### Current Consumption

*Table 32: Current Consumption*

| Board | | OLED | | GPOs |
|---|---|---|---|---|
| 40mA | + | 23mA | + | 20mA each maximum |

*Table 33: OLED Current Draw*

| OLED |
|---|
| 23mA |

### Input Voltage Specifications

*Table 34: Voltage Specifications*

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

## 7.6 Dimensional Drawings

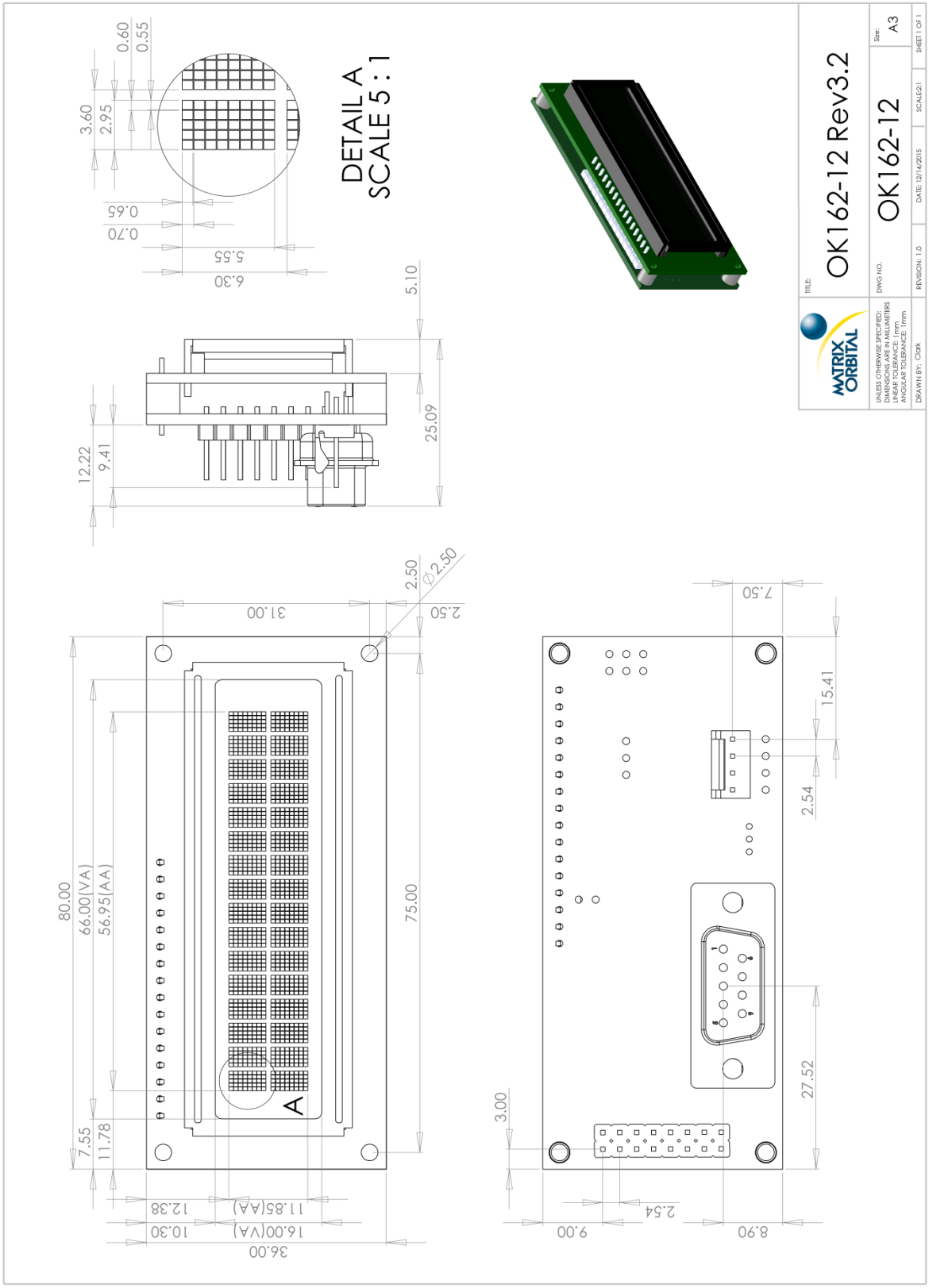



*Figure 17: OK162-12 Model Dimensional Drawing*

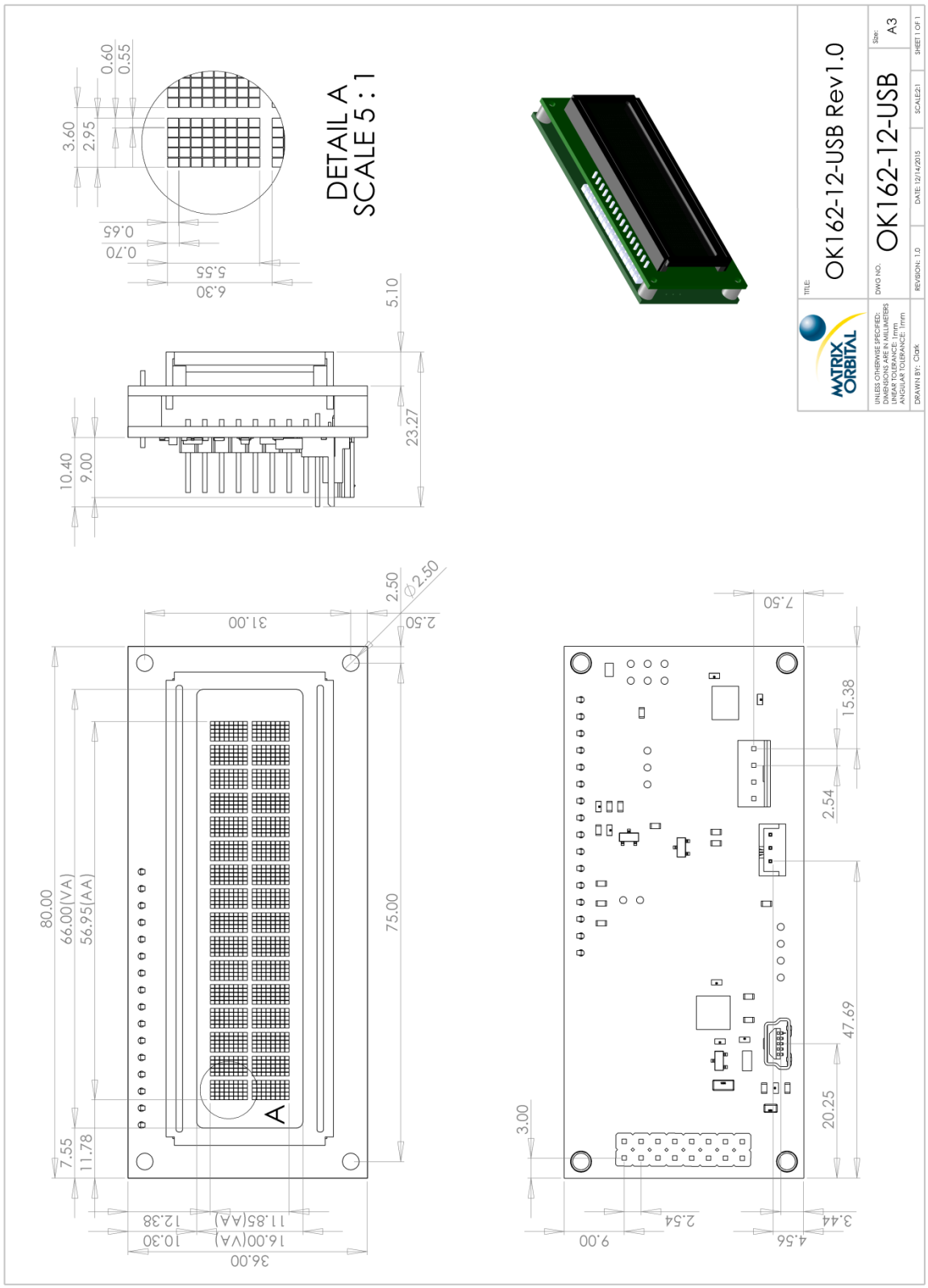



*Figure 18: OK162-12-USB Model Dimensional Drawing*

## 7.7 Optical Characteristics

*Table 35: Display Optics*

| | OLED Model | |
|---|---|---|
| **Module Size** | 80.00 x 36.00 x 25.09 | mm |
| **Viewing Area** | 66.00 x 16.00 | mm |
| **Active Area** | 56.95 x 11.85 | mm |
| **Character Size** | 2.95 x 5.55 | mm |
| **Character Pitch** | 3.60 x 6.30 | mm |
| **Pixel Size** | 0.55 x 0.65 | mm |
| **Pixel Pitch** | 0.60 x 0.70 | mm |
| **Viewing Angle** | -80 to +80 | ° |
| **Ratio** | 2000 | |
| **OLED Estimated Operating Life Time** | 100,000 | Hours |

# 8 Ordering

## 8.1 Part Numbering Scheme

*Table 36: Part Numbering Scheme*

| OK | -162 | -12 | -USB | -OR | -V | -E |
|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |

## 8.2 Options

*Table 37: Display Options*

| # | Designator | Options |
|---|---|---|
| 1 | Product Type | LK: Liquid Crystal Display with Keypad Input<br>OK: Organic Light Emitting Diode Display with Keypad Input<br>VK: Vacuum Florescent Display with Keypad Input |
| 2 | Display Size | -162: 16 columns by 2 rows |
| 3 | Keypad Size | -12: 12 key maximum |
| 4 | Protocol | *NP: Standard Model<br>-USB: USB Only Model |
| 5 | Colour | OB: OLED Blue on black<br>OG: OLED Green on black<br>OR: OLED Red on black<br>OW: OLED White on black<br>OL: OLED Yellow on black |
| 6 | Voltage | *NP: Standard Voltage<br>-V: Wide Voltage |
| 7 | Temperature | *NP: Standard<br>-E: Extended Temperature |

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

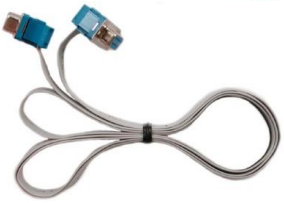
## 8.3 Accessories

### Power

| | | |
|---|---|---|
| **PCS** | Standard Power Cable | |

### Communication

*Table 38: Communication Accessories*

| | | |
|---|---|---|
| **CSS1FT** | 1 ft. Serial Cable | |
| **CSS4FT** | 4 ft. Serial Cable |  |
| **EXTMUSB3FT** | Mini-USB Cable | |
| **INTMUSB3FT** | Internal Mini-USB Cable | |
| **SCCPC5V** | Serial Communication/5V Power Cable | |
| **BBC** | Breadboard Cable | |

### Peripherals

*Table 39: Peripheral Accessories*

| | | |
|---|---|---|
| **KPP4x4** | 16 Button Keypad | |
| **Temperature Probe** | Dallas One-Wire Temperature Probe | |

# 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.

DOW: Dallas One-Wire protocol, similar to $I^2C$, provides reduced data rates at a greater distance. One wire carries data, while two others supply power and ground. Matrix Orbital tests non-parasitic devices only, those that do not draw power from the data line; however, some parasitic devices may work.

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.

$I^2C$: 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.

OLED: Organic light emitting diode display.

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 $I^2C$ 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 $I^2C$ protocol. This open drain line should be pulled high through a resistor. Nominal values are between 1K and 10K Ω.

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